

Análisis, especificación, implementación y
evaluación de una forma alternativa de
Interacción, Guiada por Objetivos,
para usuarios ocasionales, no expertos,
o con necesidades especiales



Antonio L. Carrillo León

Director Dr. Juan A. Falgueras Cano

Departamento de Lenguajes y Ciencias de la Computación

Universidad de Málaga

Tesis Doctoral

Programa de doctorado en Tecnologías Informáticas


E.T.S. Ingeniería Informática

Noviembre, 2019



UNIVERSIDAD
DE MÁLAGA

AUTOR: Antonio Luis Carrillo León

 <http://orcid.org/0000-0002-5710-4762>

EDITA: Publicaciones y Divulgación Científica. Universidad de Málaga



Esta obra está bajo una licencia de Creative Commons Reconocimiento-NoComercial-SinObraDerivada 4.0 Internacional:

<http://creativecommons.org/licenses/by-nc-nd/4.0/legalcode>

Cualquier parte de esta obra se puede reproducir sin autorización pero con el reconocimiento y atribución de los autores.

No se puede hacer uso comercial de la obra y no se puede alterar, transformar o hacer obras derivadas.

Esta Tesis Doctoral está depositada en el Repositorio Institucional de la Universidad de Málaga (RIUMA): riuma.uma.es



Carta de Aval



Departamento de Lenguajes y Ciencias de la Computación
Escuela Técnica Superior de Ingeniería Informática
Universidad de Málaga

El Dr. **Juan A. Falgueras Cano**, Titular de Universidad perteneciente al Departamento de Lenguajes y Ciencias de la Computación de la Universidad de Málaga,

certifica

que D. **Antonio L. Carrillo León**, Licenciado en Informática por la Universidad de Sevilla, y profesor Titular de Escuela Universitaria del departamento de Lenguajes y Ciencias de la Computación de la Universidad de Málaga, ha realizado en dicho departamento, y bajo su dirección, el trabajo de investigación correspondiente a su Tesis Doctoral titulada:

Análisis, especificación, implementación y evaluación de una forma alternativa de interacción, guiada por objetivos, para usuarios ocasionales, no expertos, o con necesidades especiales

Revisado el presente trabajo, estimo que puede ser presentado al tribunal que ha de juzgarlo. Y para que conste a efectos de lo establecido en la legislación vigente, autorizo la presentación de la Tesis Doctoral en la Universidad de Málaga.

En Málaga, noviembre de 2019

Firmado: Dr. Juan A. Falgueras Cano

DECLARACIÓN DE AUTORÍA Y ORIGINALIDAD DE LA TESIS PRESENTADA PARA OBTENER EL TÍTULO DE DOCTOR

D. ANTONIO LUIS CARRILLO LEÓN

Estudiante del programa de doctorado TECNOLOGÍAS INFORMÁTICAS de la Universidad de Málaga, autor/a de la tesis, presentada para la obtención del título de doctor por la Universidad de Málaga, titulada: ANÁLISIS, ESPECIFICACIÓN, IMPLEMENTACIÓN Y EVALUACIÓN DE UNA FORMA ALTERNATIVA DE INTERACCIÓN, GUIADA POR OBJETIVOS, PARA USUARIOS OCASIONALES, NO EXPERTOS, O CON NECESIDADES ESPECIALES

Realizada bajo la tutorización de LLANOS MORA LÓPEZ y dirección de JUAN A. FALGUERAS CANO (si tuviera varios directores deberá hacer constar el nombre de todos)

DECLARO QUE:

La tesis presentada es una obra original que no infringe los derechos de propiedad intelectual ni los derechos de propiedad industrial u otros, conforme al ordenamiento jurídico vigente (Real Decreto Legislativo 1/1996, de 12 de abril, por el que se aprueba el texto refundido de la Ley de Propiedad Intelectual, regularizando, aclarando y armonizando las disposiciones legales vigentes sobre la materia), modificado por la Ley 2/2019, de 1 de marzo.

Igualmente asumo, ante a la Universidad de Málaga y ante cualquier otra instancia, la responsabilidad que pudiera derivarse en caso de plagio de contenidos en la tesis presentada, conforme al ordenamiento jurídico vigente.

En Málaga, a 1 de FEBRERO de 2020

Fdo.: ANTONIO L. CARRILLO LEÓN

Agradecimientos

Hace ya tiempo que comenzó el trabajo que culmina ahora con la presentación de esta tesis. Ha supuesto un continuo esfuerzo y dedicación. Sin embargo, hasta estos últimos años no ha podido ser todo lo intenso y productivo que uno hubiese querido. A veces la vida te sorprende y se complica, y te hace tener que elegir y poner prioridades. Agradezco a Dios que me ayudase en mi elección porque realmente mereció la pena. Y también, que me haya allanado el tramo final del camino y facilitado el poder intensificar el trabajo en el momento decisivo.

También agradezco a lo mejor de mi vida, mi familia y amigos, su paciencia y comprensión por los momentos de ausencia. Y muy especialmente, a mis queridos y excepcionales padres y hermana, y a mis dos grandes tesoros, mi esposa Silvia y mi hijo Adrián. Son lo más. Los adoro. Espero poder compensarlos a todos.

Y, por supuesto, estaré siempre muy agradecido a mi tutora, Llanos, y a mi gran amigo y director de tesis, Juan. Han sido muchas las horas de trabajo y las vivencias compartidas, y muchos consejos, ánimos y empujones los que me ha dado. Han sido fundamentales para mí.

Finalmente, quiero agradecer el apoyo y ayuda de mis compañeros y de todos aquellos que han participado o colaborado de alguna manera en el trabajo realizado y presentado, siempre desinteresadamente, simplemente porque son buenas personas. Gracias a todos.



UNIVERSIDAD
DE MÁLAGA

Índice general

Capítulo 1. Introducción.....	1
1.1. Introducción.....	1
1.2. Motivación del Trabajo de Tesis.....	4
1.3. Estructura de la Tesis.....	6
Capítulo 2. Un usuario cada vez más extendido, el Usuario Ocasional... 9	
2.1. Introducción.....	9
2.2. Representación y clasificación del usuario.....	11
2.2.1. Representación del usuario: usuario promedio vs categoría de usuarios.....	12
2.2.2. Clasificación de usuarios: experiencia y aprendizaje.....	14
2.3. Las clasificaciones de usuario en contexto	15
2.3.1. Utilidad de las clasificaciones de usuario en el diseño de la interfaz.....	15
2.3.2. El reducido número de publicaciones recientes sobre clasificaciones de usuarios	16
2.4. Revisión de la literatura sobre clasificaciones de usuarios	17
2.4.1. Revisión de las clasificaciones de usuarios inexpertos e infrecuentes en la literatura académica y científica.....	18
2.4.2. Revisión de los usuarios inexpertos e infrecuentes en otras fuentes	26
2.4.3. Análisis de las variables representativas utilizadas para definir usuarios inexpertos e infrecuentes	27
2.5. Lecciones aprendidas de las clasificaciones de usuarios analizadas	28
2.5.1. Importancia de la inexperiencia.....	28
2.5.2. Importancia de la infrecuencia	30
2.6. Síntesis: necesidad de una nueva caracterización del Usuario Ocasional.....	32
2.7. Usuario Ocasional: características y parámetros para su definición.	34
2.7.1. Variables específicas para la definición del UO.....	34
2.7.2. Diferencias con otras categorías de usuarios.....	35
2.7.3. Herramienta para clasificar a un usuario como UO: el árbol de decisión	36
2.7.4. Ejemplos de Usuarios Ocasionales	37
2.8. Implicaciones y recomendaciones para el diseño de la IU	40
2.8.1. La incertidumbre sobre el uso futuro, y otros factores.....	40

2.8.2.	Recomendaciones y principios de diseño para un uso ocasional.....	41
2.8.3.	Resumen de estudios empíricos que respaldan los principios y recomendaciones sobre el diseño de la IU para el UO.....	42
2.8.4.	Estudio 1: uso ocasional de aplicaciones para la compra de productos, mediante tableta táctil.....	43
2.8.5.	Estudio 2: uso ocasional de aplicaciones para el diseño de cocinas.....	45
2.9.	Resumen y Conclusiones	47
Capítulo 3.	Interacción Guiada por Objetivos	49
3.1.	Introducción	49
3.2.	Los Estilos de interacción y el Usuario Ocasional	51
3.3.	Fundamentos de la Interacción Guiada por Objetivos	61
3.4.	La interfaz IGO	63
3.4.1.	La Ventana de Objetivos	64
3.4.1.1.	Métodos y Selecciones: Pasos y Alternativas.....	64
3.4.1.2.	Cancelación de Objetivos.	69
3.4.1.3.	Navegación hacia delante y hacia atrás.....	70
3.4.2.	La Ventana de Trabajo (VT).....	71
3.4.3.	La Jerarquía de Objetivos Activos (JOA)	71
3.5.	Metodología y notación para la especificación de <i>Interfaces IGO</i>	71
3.5.1.	Objetivos.....	73
3.5.2.	Acciones.....	73
3.5.3.	Métodos	74
3.6.	Ejemplo de especificación NGOMSL-IGO	82
3.7.	Resumen de las aportaciones de la notación NGOMSL-IGO	86
3.8.	Limitaciones de la IGO.....	87

**Capítulo 4. Interacción Guiada por Objetivos vs Manipulación Directa:
un estudio empírico..... 89**

4.1.	Introducción	89
4.2.	Participantes	90
4.3.	Material empleado y tareas a realizar	91
4.4.	Diseño del estudio	94



4.5. Medidas empíricas	96
4.5.1. Medidas objetivas: tiempos y errores.....	96
4.5.2. Medidas subjetivas; cuestionario <i>Post-interfaz</i> y cuestionario <i>Comparativo</i> final .	99
4.6. Procedimiento	102
4.7. Resultados	104
4.7.1. Resultados objetivos: tiempos y errores.....	105
4.7.2. Resultados subjetivos	107
4.7.3. Resultados comparativos directos; preferencias de los participantes	108
4.7.4. Resultados de un pequeño estudio realizado con Eye-Tracker	110
4.8. Discusión y conclusiones	112

Capítulo 5. AGILE: Interacción Guiada y Asistida para personas mayores sin experiencia con la tecnología (novice elderly users)115

5.1. Introducción.....	115
5.1.1. ¿Qué es AGILE?.....	115
5.1.2. Comprensión de los usuarios y sus necesidades.....	116
5.1.3. Nuestro usuario objetivo: personas mayores sin experiencia con la tecnología (novice elderly user).....	117
5.2. Revisión de la literatura.....	118
5.2.1. La innecesaria doble exclusión.....	118
5.2.2. La validez de la inclusión	119
5.2.3. Modelos Inclusivos, modelos de Tareas y modelos Cognitivos	120
5.2.4. ¿Por qué los modelos principales no cubren las necesidades de los <i>novice elderly users</i> ?	121
5.3. Fundamentos de la <i>Interacción AGILE</i>.....	123
5.3.1. Importancia del entorno	123
5.3.2. La interfaz como entorno del usuario.....	125
5.3.3. Adaptación de la interfaz para el <i>novice elderly user</i>	126
5.3.4. Variables que caracterizan al <i>novice elderly user</i>	127
5.3.5. Valores de las variables que definen al <i>novice elderly user</i>	129
5.3.6. Los objetivos como determinantes del comportamiento del usuario	130
5.3.7. Implicaciones y recomendaciones para el diseño	132
5.4. El modelo AGILE.....	133
5.4.1. Filosofía de interacción.....	133
5.4.2. Aprendiendo de otros estilos de interacción	134

5.4.3. Metodología.....	134
5.4.4. Descripción del prototipo AGILE inicial.....	136
5.5. Evaluación de la interfaz AGILE	137
5.5.1. Metodología de evaluación.....	137
5.5.2. Iteración 1: desarrollo y evaluación del prototipo inicial	138
5.5.3. Iteración 2: rediseño, implementación y evaluación	139
5.5.4. Resultados de la evaluación comparativa de las interfaces para la compra de billetes de tren, en iPad2	143
5.5.5. Resultados de la evaluación comparativa de las interfaces para la compra de libros, en iPad2.....	144
5.6. Discusión	144
5.7. Conclusiones.....	150
Capítulo 6. Conclusiones y trabajos futuros.....	153
Anexo I. Especific. NGOMSL-IGO de la interfaz IGO evaluada en Cap.4.	159
Anexo II. Publicaciones que avalan la Tesis	163
Listado de Figuras	165
Listado de Tablas	169
Bibliografía	171

Capítulo 1. Introducción

Esta tesis implementa una nueva forma de interacción, la *Interacción Guiada por Objetivos*, dirigida a usuarios que no puedan o no quieran abordar una curva de aprendizaje, resultando especialmente apropiada para un tipo de usuario, el *Usuario Ocasional*, que analizamos y definimos en los primeros capítulos. Como soporte de este estilo de interacción especificamos una *metodología para el desarrollo y modelado* de las correspondientes interfaces de usuario. Los resultados de un completo proceso de evaluación empírica con usuarios reales confirmarán la adecuación de este tipo de interacción. Finalmente, y bajo un enfoque inclusivo, desarrollamos una variante de estas interfaces guiadas, especialmente dirigida a personas mayores con poca experiencia tecnológica (*novice elderly users*), avalada también por un proceso transnacional de evaluación con usuarios y técnicas de eye-tracking.

1.1. Introducción

La investigación presentada en esta memoria queda encuadrada dentro del ámbito de las *Interfaces de usuario* y de la *Interacción Persona-Ordenador*, área multidisciplinar relacionada con el diseño, implementación y evaluación de sistemas informáticos interactivos para su uso por seres humanos, y con el estudio de los fenómenos más importantes con los que están relacionados [52]. En concreto, el objetivo fundamental de esta tesis ha sido ofrecer una nueva forma de interacción guiada, alternativa a las tan extendidas interfaces de *manipulación directa*, y caracterizada por no requerir al usuario ni conocimiento previo de la interfaz ni un periodo de aprendizaje, guiándolo en todo momento tanto en el “qué hacer” como en el “cómo hacerlo”. Como consecuencia, resultará especialmente adecuada para un tipo de usuario cada vez más frecuente y extendido, el *usuario ocasional*, no especificado formalmente en la literatura del área hasta ahora, y que será analizado en profundidad y caracterizado acorde a las nuevas tecnologías y contextos de uso. La definición de este estilo de interacción conlleva el desarrollo de una completa metodología, adecuada para realizar el diseño y la especificación de las correspondientes

interfaces de usuario. También se ha puesto a prueba y sometido a evaluación esta forma de interacción a través de un estudio empírico, tanto cuantitativo como cualitativo, con usuarios reales, que demostró lo acertado de la propuesta. La investigación presentada se completa con el análisis, desarrollo y una evaluación transnacional de una variante de esta forma de interacción, dirigida especialmente a personas mayores sin experiencia con la tecnología y con necesidades especiales (principalmente cognitivas) debidas a la edad.

Uno de los principios esenciales del diseño de todo sistema interactivo es el de conocer bien al usuario. Esto permite realizar sistemas más usables, interfaces más apropiadas, menos ensayo y error en el proceso de diseño y uso de la interfaz, menor entrenamiento del usuario, etc. [132]. En este sentido, las *clasificaciones de usuarios* proporcionan a los diseñadores de la *interfaz de usuario* (IU) un catálogo de necesidades y habilidades de sus usuarios que puede aportar una valiosa información para su trabajo.

En una *clasificación de usuarios*, éstos se agrupan en *categorías* concretas y previamente definidas y, en función de ciertos factores o características, como su experiencia general con la tecnología o con el sistema en particular, dichos usuarios se mantienen o se desvían de dichas categorías en función de su proceso de aprendizaje. Desde hace varias décadas, la que más se ha utilizado e influido en el trabajo de investigadores, educadores y diseñadores de la interfaz de usuario ha sido la ya tradicional clasificación de 3 categorías: *novatos* o *principiantes* (*novice*), *intermedios* (*intermediate*) y *expertos* (*expert users*). Shneiderman, desde la primera edición (1986) de su libro “Designing the User Interface”, un clásico en el área de la *Interacción Persona-Ordenador* (IPO), ya proponía dicha clasificación; 20 años después, en su sexta edición, aún se sigue manteniendo [157].

Estas categorías surgieron para conceptualizar las necesidades, estrategias y objetivos de los usuarios alrededor de los años 80, cuando el *estilo de interacción* predominante era el de las *Interfaces de Líneas de Comandos*, cuya sintaxis no era intuitiva ni fácil de aprender para los no expertos [174], al mismo tiempo que les requería un conocimiento sustancial de los conceptos relacionados con la computadora. Por tanto, para poder hacer un uso efectivo y eficiente del sistema, los usuarios solían verse obligados a tener que abordar una dura curva de aprendizaje, pasando por las 3 categorías antes comentadas.

Posteriormente se estructuraron los elementos gráficos de las interfaces de usuario, apareciendo *ventanas*, *iconos*, *menús* y el *puntero*, haciéndose cada vez más frecuentes en las diferentes plataformas, dando lugar a lo que hoy se conoce como *interfaces WIMP* (Windows, Icons, Menus and Pointers). Estos, junto a otros elementos, representan *metáforas* [26] de objetos del mundo real y, por primera vez, permitían al usuario interactuar con ellos en la pantalla a través de un dispositivo apuntador como el ratón, de una manera mucho más familiar e intuitiva, permitiendo la *manipulación directa* de los elementos de la interfaz [154, 157]. En comparación con la *interfaz de línea de comandos*, este tipo de interfaz gráfica de *manipulación directa* dio como resultado un *estilo de interacción* mucho más accesible, familiar e intuitivo para usuarios sin muchos conocimientos ni experiencia con el sistema, especialmente si se aplican bien los principios y guías de usabilidad y accesibilidad.

A pesar de todo, este tipo de interfaz necesita que el usuario reconozca los elementos o conceptos que orquestan la metáfora. Sin embargo, y aunque deberían resultarle familiares, esto no siempre ocurre, por lo que el usuario puede tener que seguir cierto proceso de aprendizaje y de familiarización con sus componentes y funcionalidades [161, 168, 157]. Generalmente, esto se consigue a través del uso repetido de la interfaz. Según Stasko [161]:

“[...] aunque las interfaces gráficas de usuario de manipulación directa tipo WIMP son un gran avance respecto a los terminales por líneas de comandos, todavía requieren cierta curva de aprendizaje y pueden resultar difíciles de usar”.

Sin embargo, ¿qué ocurre cuando el usuario necesita utilizar el sistema o realizar una tarea concreta de forma ocasional, pero no quiere o no puede abordar una curva de aprendizaje? ¿Qué pasa si el usuario necesita utilizar el sistema con una frecuencia irregular o no planificada, y lo que recuerda de sus usos anteriores, caso de haberlos, no le proporciona el conocimiento suficiente para el uso actual?

Existen, al menos, tres argumentos a favor, tanto de la consideración explícita de este uso ocasional, como a favor del diseño explícito de las interfaces de usuario correspondientes: avances en las interfaces de usuario, nuevos enfoques en el diseño y nuevas tecnologías y redes informáticas. Por un lado, la gran evolución de las *interfaces gráficas de usuario* [157, 105] con la aparición de los dispositivos *multitáctiles* [16, 147,

122] ha aumentando notablemente el número y la heterogeneidad de los usuarios. En segundo lugar, el aumento de *usuarios ocasionales* también se ha visto impulsado por la continua incorporación de principios de *Usabilidad* [118, 114, 53], *Accesibilidad* [170, 1, 104, 172], y *Diseño inclusivo* [29, 142]. Y, por último, y debido en gran medida al desarrollo y expansión de *Internet*, a la permanente interconexión on-line de los dispositivos, a las *tecnologías móviles* [27, 44, 152], e incluso de las *tecnologías Self-Service* [8, 96, 97], también han surgido nuevos y variados contextos de uso de la tecnología en espacios físicos que tradicionalmente estaban dedicados a fines no tecnológicos, como ciertos espacios exteriores, lugares de interés turístico, centros comerciales, aeropuertos, supermercados, etc.

Por tanto, todos estos avances y diversos contextos de uso de la tecnología requieren que el usuario realice actividades no rutinarias ni planificadas previamente, actividades que implican conseguir un objetivo de forma espontánea u ocasional, pero a la vez inmediata y efectiva, como ocurriría al comprar un billete de tren en una máquina de autoservicio cuando el tren está a punto de partir, o al usar un sistema de audio-guía en nuestro dispositivo móvil durante la visita a un museo o a un lugar de interés arqueológico, o al intentar realizar, de manera autónoma y en la propia tienda, la compra del mobiliario de la cocina de casa, usando un ordenador dispuesto a tal efecto por el propio comercio, etc. El objetivo prioritario en estos casos es conseguir realizar la tarea, correctamente y en un tiempo aceptable. En tales situaciones, dedicar tiempo y esfuerzo a aprender a usar el sistema o a aumentar la pericia en su manejo no es tan relevante como conseguir el objetivo en sí. Lograr la meta, es decir, conseguir el billete a tiempo se convierte en la prioridad, mientras que aprender o incrementar la “expertise” durante ese proceso de interacción se vuelve algo circunstancial, ya que el usuario podría no saber siquiera si volverá a usar el mismo sistema más adelante, como le ocurriría al *usuario ocasional*.

1.2. Motivación del Trabajo de Tesis

Las definiciones de las categorías de usuarios ya establecidas no cubren ni las características ni los escenarios de uso descritos en el apartado anterior. Esto, junto al desajuste detectado entre la tradicional clasificación de usuarios de tres categorías, y los “nuevos”, ubicuos y cada vez más frecuentes *usuarios ocasionales*, nos llevó a realizar una revisión y análisis de las clasificaciones de usuarios existentes en la literatura al respecto.

Este primer estudio tenía el objetivo de investigar las características y variables que se utilizan para definir a los usuarios que son poco frecuentes o son inexpertos en el uso de un sistema, por su relación más directa con el uso ocasional. El resultado fue la realización de una nueva y actualizada *definición y caracterización* del *usuario ocasional*, así como la especificación de una serie de *principios* generales a tener en cuenta en el diseño de interfaces para ellos.

Sin embargo, como ya se ha indicado, el gran objetivo de esta tesis era el de ofrecer una alternativa a las tradicionales y tan extendidas interfaces de *manipulación directa*, con el desarrollo y propuesta una nueva forma de interacción guiada que se caracterizase por no requerir al usuario conocimiento previo sobre el uso de la interfaz y por guiarlo en todo momento para la consecución de sus objetivos con el sistema. Por tanto, esta interacción resultará especialmente apropiada para aquellos usuarios que no puedan o no quieran abordar una curva de aprendizaje, y en particular, para nuestros *usuarios ocasionales*. La definición de este estilo de interacción, denominado *Interacción Guiada por Objetivos*, conllevaba la especificación de una completa metodología de diseño y una notación para el modelado de las correspondientes interfaces de usuario. Finalmente, también era necesario evaluar y validar esta forma de interacción a través de un estudio empírico con usuarios, que confirmase lo acertado de la propuesta.

Por otro lado, el constatar que la *Interacción Guiada por Objetivos* se podía adaptar bastante bien a las características y necesidades de las *personas mayores sin experiencia con la tecnología* (*novice elderly users*) motivó la propuesta, desarrollo y evaluación de una variante de esta forma de interacción, dirigida especialmente a estas personas mayores con capacidades cognitivas reducidas por la edad y que se enfrentan hoy día a un mundo cada vez más tecnológico. El proceso de evaluación de este tipo de interfaces se desarrolló desde dos universidades, una de Escocia y la nuestra de Málaga, con pruebas con usuarios y técnicas de *eye-tracking*, utilizando aplicaciones sobre dispositivos táctiles móviles. El resultado es la metodología AGILE (*Assistive Guided Interaction with no Learning nor Experience required*).

1.3. Estructura de la Tesis

A continuación, tras enumerar los capítulos que constituyen esta memoria, pasaremos a describir con cierto detalle la estructura y los principales aspectos tratados en cada uno de ellos:

- Capítulo 1. Introducción (capítulo actual).
- Capítulo 2. Un usuario cada vez más extendido, el *Usuario Ocasional*.
- Capítulo 3. La *Interacción Guiada por Objetivos (IGO)*.
- Capítulo 4. Interacción Guiada por Objetivos vs Manipulación Directa: un estudio empírico.
- Capítulo 5. *AGILE*: Interacción Guiada y Asistida para personas mayores sin experiencia con la tecnología (*novice elderly users*).
- Capítulo 6. Conclusiones.
- Anexos, Listados de figuras y de tablas, y Bibliografía

En el **Capítulo 2** comenzamos describiendo un tipo de usuario que los diseñadores de la interfaz suelen considerar en muchas ocasiones: el *usuario representativo* o *usuario promedio*. Se cuestiona si este tipo de usuario refleja realmente el amplio espectro de usuarios existente actualmente, y si, en última instancia, resulta útil para el diseño de sistemas interactivos con categorías heterogéneas de usuarios. Posteriormente, y tras constatar la falta de literatura reciente al respecto, realizamos una revisión y análisis de las distintas *clasificaciones de usuarios* existentes, prestando especial interés a aquellas que se centran en usuarios inexpertos e infrecuentes por compartir las mismas dimensiones de tiempo y conocimiento con nuestro usuario objetivo, el ocasional. Dicha revisión, además de proporcionar una perspectiva cronológica de las distintas *categorías*, tiene como objetivo determinar las variables que las definen, y si tales categorías cubren todo el espectro de usuarios actuales. Tras un proceso de síntesis, se pone en evidencia la necesidad de realizar una nueva caracterización del *usuario ocasional* acorde con las nuevas tendencias, tecnologías y escenarios de interacción. Esta tarea la concretamos especificando las variables que permiten definir a un *usuario* como *ocasional*, incluimos un árbol de decisión que puede ayudar a categorizar a estos usuarios como tales, y terminamos enumerando claros ejemplos de usuarios de este tipo. Después analizamos y concretamos las principales implicaciones y recomendaciones, en forma de *principios*

generales de diseño, que la nueva caracterización del *usuario ocasional* tiene de cara al desarrollo de las interfaces para estos usuarios.

En el **Capítulo 3**, profundizamos en el análisis anterior, para abordar el objetivo principal de esta tesis. Comenzamos revisando los principales *estilos de interacción* para concluir con la propuesta y descripción de una nueva forma de interacción, concebida especialmente para estos *usuarios ocasionales* y, en general, para aquellos usuarios que no quieran o no puedan abordar una curva de aprendizaje. La hemos denominado *Interacción Guiada por Objetivos* (IGO) y su filosofía de interacción consiste en guiar y conducir al usuario, en todo momento, y de manera jerárquica y progresiva, tanto en lo que respecta a los objetivos a abordar, como a los pasos o elecciones a realizar en cada momento para conseguirlos. Para ello, tomaremos y combinaremos aspectos y elementos de otros estilos de interacción, así como características e ideas de una de las principales técnicas del modelado cognitivo y del análisis jerárquico de tareas, la técnica NGOMSL de D. Kieras [72, 73]. Posteriormente, basándonos en ella, pero adaptándola y extendiéndola, propondremos una metodología y una notación especialmente adecuadas para realizar el proceso de modelado o especificación de este tipo de interfaces. Terminamos el capítulo concretando las limitaciones de esta forma de interacción.

En el **Capítulo 4**, y para respaldar las aportaciones anteriores, sometemos a evaluación a esta forma de interacción a través de un estudio empírico comparativo, basado en datos tanto cuantitativos como cualitativos. Para ello, se desarrollan y utilizan dos tipos de interfaces para una misma aplicación destinada al uso ocasional: una, la tradicional interfaz de *Manipulación Directa* (inspirada en la aplicación de la empresa IKEA), y la otra, siguiendo una *Interacción Guiada por Objetivos*. Nuestro propósito era, además de tomar medidas sobre tiempos y errores, que *usuarios ocasionales* reales pudiesen “experimentar” ambos tipos de interacción, y decidir cuál consideraban más apropiada para ellos. Comenzaremos este capítulo estableciendo 3 hipótesis de partida, describiremos las características de los participantes involucrados, el material empleado, y las tareas a realizar en los tests de usuario. Después, diseñaremos el tipo de estudio a llevar a cabo, concretaremos los tipos de medidas a tomar (tanto las objetivas como las subjetivas), y describiremos el procedimiento a seguir para realizar la evaluación. Finalmente, presentaremos los resultados, los discutiremos, y terminaremos con las conclusiones del estudio.

En el **Capítulo 5** presentamos la *Interacción AGILE*, una forma de interacción guiada y asistida, derivada de la *Interacción Guiada por Objetivos*, pero enfocada especialmente a *personas mayores sin experiencia con la tecnología*, con necesidades especiales (principalmente cognitivas) debidas a la edad (*novice elderly users*). El objetivo es, desde un *enfoque inclusivo*, mejorar la usabilidad y satisfacción con la interfaz de dichos usuarios en la realización de transacciones digitales. Comenzaremos realizando un análisis holístico de estos usuarios, de sus objetivos y necesidades, y realizando una revisión de la literatura al respecto. El apartado 5.3 se centra en los fundamentos de la *interacción AGILE*. Se analizarán y concretarán las variables (y sus valores) que permiten caracterizar y definir al *novice elderly user*, se destacará la importancia de los objetivos del usuario como determinantes de su comportamiento, y extraerán las implicaciones y recomendaciones para el diseño que se desprenden de este estudio. En la sección 5.4 se especifica ya la filosofía de esta forma de interacción, se describen los objetivos específicos de la metodología a seguir para el desarrollo de las correspondientes interfaces de usuario, y se describe la estructura propuesta inicialmente para las *interfaces AGILE*. En la sección 5.5, tras concretar la metodología de evaluación a seguir, se aborda, en una primera iteración del proceso, el desarrollo y evaluación del prototipo de interfaz AGILE inicial, y en una segunda iteración, su rediseño, implementación y evaluación (mediante tests de usuarios realizados en dos países y con uso de eye-tracker). Terminaremos el capítulo discutiendo los resultados y extrayendo las conclusiones.

En el **Capítulo 6** terminaremos extrayendo las principales *conclusiones* sobre todo lo expuesto en esta memoria, y sugiriendo posibles trabajos futuros.

Como complemento, incorporamos dos apéndices. En concreto, el **Anexo II** recoge las principales aportaciones, en forma de *publicaciones*, que se han realizado durante el desarrollo de esta tesis.

Capítulo 2. Un usuario cada vez más extendido, el Usuario Ocasional

Este capítulo se dedica al estudio y caracterización del *Usuario Ocasional*, un tipo de usuario de las nuevas tecnologías cada vez más frecuente.

Comenzaremos introduciendo los conceptos relacionados con las clasificaciones y las categorías de usuarios, destacando la importancia de las mismas.

Realizaremos una profunda revisión y análisis de la literatura existente al respecto, prestando especial interés a aquellas que se centran en usuarios inexpertos e infrecuentes por estar más relacionadas con nuestro usuario objetivo.

Las conclusiones de dicho trabajo nos permitirán realizar una nueva caracterización del *Usuario Ocasional*,

acorde con las nuevas tendencias, tecnologías y escenarios de interacción, y proporcionar una serie de recomendaciones o principios básicos de diseño de la interfaz para estos usuarios.

El siguiente capítulo se dedicará al elemento central de esta tesis, una nueva forma de interacción guiada que, como veremos, resultará especialmente adecuada a las características y necesidades de estos usuarios.

2.1. Introducción

Como ya se introdujo en el capítulo anterior, uno de los principales aspectos a tener en cuenta en el proceso de diseño de todo sistema interactivo, y más en concreto, de su interfaz de usuario (IU), es conocer bien las características de sus usuarios [46]. A su vez, la idoneidad de la representación del usuario es un factor importante que puede influir notablemente, no solo en el diseño del sistema y de su interfaz, sino también, en la forma y grado de éxito con que los usuarios lo usarán [132]. Una de las formas más utilizadas para representar a los usuarios de un sistema es a través de *clasificaciones de usuarios*.

Tres categorías de usuarios, *principiantes*, *intermedios* y *expertos*, han dominado el trabajo de los investigadores, educadores y diseñadores de la interfaz de usuario (UI) durante décadas [155, 157]. Estas categorías eran el denominador común de las

clasificaciones de usuario que aparecieron alrededor de los años 80 para ayudar a estos profesionales de la IU a conceptualizar las necesidades, estrategias y objetivos de los usuarios.

Desde entonces, se han producido al menos dos cambios de paradigma en la historia de las IU. El primero se correspondió con la evolución desde una interfaz basada en líneas de comandos y exclusiva para expertos, a una interfaz gráfica mucho más intuitiva basadas en *metáforas* [26], incluyendo ventanas, iconos, menús y punteros (WIMP), donde se representan tanto los objetos como las acciones, y se permite la *manipulación directa* de dichos objetos [154, 157]. El otro gran cambio lo ha provocado la aparición de las *interfaces táctiles* [16, 147, 122] en *dispositivos móviles* [27, 44, 152], consolidando una coreografía de gestos interactivos que hacen que los habituales periféricos, como el ratón y el teclado físico, no sean tan esenciales.

Afortunadamente, estos dos grandes hitos en el desarrollo de las IU, junto con la incorporación de principios de *Usabilidad* [118, 114, 53], *Accesibilidad* [170, 1, 104, 172], y *Diseño inclusivo* [29, 142], y otros avances, como la expansión de *Internet* y de las tecnologías móviles que permiten un permanente estado *on-line* de los dispositivos, han diversificado tanto los contextos y formas de uso como el espectro de usuarios previo, dando acceso a la tecnología a personas con pocos conocimientos informáticos y para realizar actividades, no solo cotidianas o planificadas de antemano, sino también espontáneas u ocasionales.

El diseño de sistemas para este uso ocasional requiere una buena definición o identificación de los usuarios que van a utilizar dichos sistemas, y de sus necesidades y objetivos. Este será, precisamente, nuestro propósito principal en este Capítulo, la definición y especificación de estos “nuevos” *Usuarios Ocasionales* (UO), y la propuesta de principios y recomendaciones generales de diseño de la interfaz que se desprendan de dicha caracterización, con la intención de ayudar a la comunidad IPO (*Interacción Persona-Ordenador*) a abordar el desafío de diseñar para estos usuarios, sin exigirles conocimiento previo de la interfaz ni participación en una curva de aprendizaje.

En el próximo apartado comenzaremos introduciendo el tema de la representación y clasificación del usuario. Posteriormente, en la sección 2.3, ponemos en contexto dichas clasificaciones y destacamos la falta de literatura reciente al respecto. En el apartado 2.4

profundizamos en el estado del arte sobre las diversas *clasificaciones de usuario* existentes en la literatura científica, subrayando las fortalezas y debilidades de cada una respecto al uso ocasional. En la sección 2.5 pasamos a describir las lecciones aprendidas en dicho proceso de revisión, especialmente las implicaciones que la inexperiencia y la infrecuencia tienen en el uso de la tecnología y en el diseño de la interfaz de usuario. Después, ya en el apartado 2.6, y tras un proceso de síntesis de todo lo anterior, se pone en evidencia la necesidad de realizar una nueva caracterización del *Usuario Ocasional*. Esta tarea se plasma en la sección 2.7, donde especificaremos tanto las variables como los valores concretos que deben tener dichos parámetros representativos en el caso de un UO, incluimos un árbol de decisión que puede ayudar a los diseñadores a categorizar a estos usuarios como tales, y terminamos enumerando claros ejemplos de usuarios de este tipo. En el apartado 2.8 analizamos y concretamos, en forma de principios generales, las principales implicaciones y recomendaciones que la nueva caracterización del UO tiene de cara al diseño de interfaces para estos usuarios, y se resumen dos estudios empíricos que respaldan la validez de dichos principios de diseño. Estos estudios serán tratados con más detalle en capítulos posteriores. Finalmente, se dedicará una pequeña sección a extraer las conclusiones generales del trabajo reflejado en este capítulo, y a la repercusión que tendrá de cara al que se realizará en los siguientes. En concreto, en el Capítulo 2 se presentará y describirá una nueva forma de interacción guiada (y la correspondiente metodología de desarrollo y modelado de las interfaces asociadas) que resultará especialmente apropiada para el UO considerado en este capítulo.

2.2. Representación y clasificación del usuario.

La representación del usuario en el proceso de diseño de un sistema interactivo es un poderoso instrumento para que los diseñadores puedan conocer las características y habilidades de sus potenciales usuarios y hacer que la interfaz de dicho sistema las pueda explotar. La idoneidad de la representación del usuario es un factor importante que puede influir no solo en el diseño del sistema, sino también en la forma en que los usuarios lo utilizarán. Una forma de representar a los usuarios es a través de una *clasificación de usuarios*. En una clasificación de usuarios, estos se agrupan en *categorías* previamente definidas. Según su experiencia con la tecnología en general, o con un sistema específico en particular, los usuarios se mantienen en la misma categoría, o progresan o se desvían de una a otra según su proceso de aprendizaje.

2.2.1. Representación del usuario: usuario promedio vs categoría de usuarios

Entre los diversos conceptos con los que trabajan los diseñadores de interfaces de usuario se encuentra el del *usuario representativo* [68, 123]. La intención es reunir en un mismo estereotipo un conjunto de características representativas de los potenciales usuarios del sistema específico a diseñar. Se supone que este usuario representa el promedio de los posibles valores que pueden tener dichas características. Sin embargo, la desventaja de disponer de un único representante o usuario promedio puede ser equivalente a la desventaja asociada con el uso de la media numérica. Es decir, los valores de los extremos pueden ser muy diferentes o estar muy lejos del valor medio.

Cuando el asunto a tratar es conceptual en lugar de numérico, como es el caso si el sistema a diseñar está destinado a diferentes tipos de usuarios, el considerar un usuario promedio hace que, de manera inherente, se difumine la diferencia entre dichos usuarios, impidiendo una distinción y caracterización efectiva entre ellos. El concepto del usuario promedio, en contraposición a una categorización de usuario más rica desde la perspectiva de las necesidades y objetivos del usuario, puede no representar equitativamente los diferentes tipos de usuarios del sistema. Una consecuencia es que el proceso de diseño basado en un usuario promedio o un usuario representativo único (en oposición a un conjunto de usuarios representativos caracterizados, por ejemplo, mediante la técnica *Personas* [32, 135]), incorpora la perspectiva de una categoría de usuario homogénea, con las mismas necesidades y estrategias. En cambio, la continua incorporación de nuevas tecnologías altera y extiende los escenarios de uso prevalentes, aumentando el número de usuarios y, lo que es más importante, diversificando los estereotipos de usuarios.

Si bien es cierto que la incorporación de principios de *Accesibilidad* y de *Usabilidad* han aumentado la heterogeneidad en el diseño para los usuarios más comunes, todavía existe un número muy reducido de aplicaciones desarrolladas de manera efectiva para usuarios específicos, como ancianos, niños, discapacitados o cualquier otro colectivo con necesidades especiales [81, 94, 89, 86].

Langdon y Thimbleby ([77], p. 439):

“Es probable que gran parte de la investigación aceptada [relativa a la usabilidad] resulte poco adecuada en el futuro a la hora de proporcionar información sobre el diseño de las interfaces de usuario, y ciertamente inadecuada para informar sobre el diseño inclusivo”.

Por otro lado, campos como el Diseño Universal [43] y el Diseño Inclusivo [29, 142] intentan también equilibrar la escena de la investigación en interfaces de usuario, aumentando el número y la calidad de los diseños para estos usuarios, dejando de lado el enfoque marginal tradicional de la supuesta uniformidad del usuario. Sin embargo, el problema aún persiste, mostrando que diferentes categorías de software, hardware y contextos de uso pueden dar lugar a un usuario representativo (o usuario promedio) diferente para cada una de ellas, ya que lo que significa la definición de usuario promedio en un contexto puede diferir de lo que significa en otro. Por ejemplo, el usuario promedio de los que usan máquinas de escribir con interfaz analógica y mecánica no encaja en los mismos parámetros que el usuario promedio de los que realizan el self-service checkout mediante pantalla táctil en un supermercado. La participación del usuario en el primer escenario puede o no trasladarse al contexto del segundo, pero ambos usuarios podrían ser la misma persona. Además, no está claro si el modelo basado en el usuario promedio es transferible a otros dispositivos, a otros tipos de usuarios o a diferentes contextos de uso. El estereotipo del usuario promedio no siempre incorpora una descripción de su contexto de uso, y no siempre cubre los aspectos de usabilidad, accesibilidad, y técnicas de interacción requeridas por la evolución de la tecnología. Por lo tanto, cuando se considera un conjunto realista de usuarios, parece prudente considerar también una amplia variabilidad en su espectro.

En resumen, los argumentos presentados anteriormente generan dudas sobre la utilidad del concepto del usuario promedio de cara al diseño de las IU. Esto significa que, en la actualidad, el promedio de todos los perfiles de usuario no siempre sería un reflejo completo de un espectro de usuarios que crece en variabilidad, lo que invita a realizar un análisis más razonable sobre las necesidades del usuario y el contexto de uso. Estos problemas sugieren que se debe crear una relación inequívoca entre, por un lado, las aptitudes humanas cognitivas y físicas, y por otro, los nuevos tipos de dispositivos y sus escenarios de uso.

2.2.2. Clasificación de usuarios: experiencia y aprendizaje

Para el diseño de la IU, es importante comprender la relación entre el usuario, su experiencia y su aprendizaje. La Figura 2.1 ilustra uno de los conceptos subyacentes de las clasificaciones de usuarios, la *curva de aprendizaje* [115]. Dado un usuario y una interfaz, dicha curva traza el conocimiento que un usuario adquiere a través de los usos repetidos (también llamados “sesiones”) con la misma interfaz. El gráfico representa el uso (eje x) y el conocimiento que el usuario adquiere sobre la interfaz y la funcionalidad del sistema (eje y).

El usuario promedio o representativo, discutido en la sección anterior, se ubicaría hipotéticamente en el centro de la curva (región B), representando al grupo de usuarios con una experiencia media del sistema. Esta región central de la distribución delimita dos conjuntos diferentes de usuarios, unos con menos y otros con más conocimientos sobre la interfaz (región A y región C) respectivamente. Los usuarios novatos y otros con necesidades especiales no encajan en ninguna región con cierta cantidad de conocimiento (B o C), por lo tanto, pertenecen a la región con menos conocimiento sobre la interfaz (A).

La Figura 2.1 muestra cómo las clasificaciones de usuarios tradicionales conciben la adquisición de la *expertise* o conocimiento del usuario sobre la interfaz, y ofrece algunas pistas sobre por qué los usuarios inexpertos e infrecuentes pueden quedar fuera del grupo principal o mayoritario. El concepto subyacente al conocimiento de la interfaz que se obtiene a través de la experiencia previa y de las repetidas sesiones de uso de la IU explica la falta de éxito de los usuarios que no están familiarizados con ella.

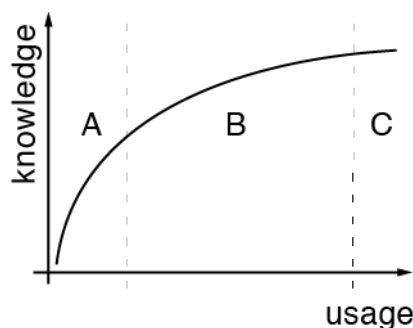


Figura 2.1 Evolución del conocimiento de la interfaz, adquirido por un usuario a lo largo de las repetidas sesiones de uso de la misma interfaz.

En la mayoría de los casos, el diseño de la IU se basa en el supuesto de que el usuario realizará más de una sesión con la misma interfaz. En teoría, a través de este mecanismo de repetición del uso se debería adquirir el conocimiento suficiente para saber cómo usarla. Esto deja a los usuarios que empiezan a usar el sistema sin otra alternativa que aplicar estrategias más simplistas, como podría ser la de *prueba y error*, y que pueden generarle cierta frustración, especialmente durante el primer o incluso único uso (si fuese el caso). El riesgo del enfoque del usuario promedio es que los usuarios más alejados de dicho promedio pueden interactuar inadecuadamente con la interfaz, convirtiéndose en usuarios “fallidos” que pueden desarrollar cierto temor hacia la tecnología o verse a sí mismos como usuarios incompetentes [130, 180]. Esta es la razón por la cual existe un claro desajuste entre los diseños de la interfaz basados en el usuario promedio, y aquellos que abordan las necesidades del usuario integrando otros valores del potencial espectro de usuarios de una interfaz, y que normalmente son excluidos por el concepto del promedio, como las personas mayores o aquellas con necesidades especiales.

2.3. Las clasificaciones de usuario en contexto

Como ya se indicó anteriormente, conocer al usuario es uno de los principios esenciales en el diseño de la IU [46]. Razones de tal importancia son que permite realizar sistemas más usables, interfaces más apropiadas, menos *ensayo y error* en el proceso de diseño, menor entrenamiento del usuario, etc. [132]. En este sentido, las *clasificaciones de usuarios* proporcionan a los diseñadores de la IU un catálogo de necesidades y habilidades de sus usuarios que puede aportar una valiosa información para su trabajo.

2.3.1. Utilidad de las clasificaciones de usuario en el diseño de la interfaz

Históricamente, los primeros acercamientos a las clasificaciones de usuarios se desarrollaron en el contexto de la investigación sobre bases de datos [169] para evaluar el mejor enfoque para los lenguajes de consulta y gestión de datos. Los problemas asociados con el uso de lenguajes de comandos para comunicarse con la máquina motivaron dichos estudios con el fin de encontrar las mejores técnicas [174]. En ese momento, la preocupación principal era determinar cómo los usuarios podían manejar satisfactoriamente la información al realizar consultas de datos, reduciendo el número de posibles errores y aumentando el grado de satisfacción, para lo que se llevaron a cabo análisis exhaustivos de dichas consultas.

Una clasificación de usuarios ayuda a conocer la experiencia de los potenciales usuarios de una aplicación. Por ejemplo, Schneider [144] creó una clasificación de cinco categorías abarcando, desde las personas que usan el sistema sin comprender lo que están haciendo (*parrots*), hasta los denominados *maestros*, e incluyendo también a *novatos*, *intermedios*, y *expertos*. Este modelo de cinco etapas se denominó “modelo prescriptivo”, porque proporcionaba a los diseñadores información valiosa sobre el nivel de experiencia que los usuarios podían presentar al usar un sistema.

Hay más razones que explican por qué las clasificaciones de usuarios son útiles. Por un lado, contribuyen a una mejor comprensión del usuario final. Una clasificación confiable debe incluir las características más representativas y relevantes del usuario. El rango y los valores asociados a estas características contribuyen a dibujar un mapa apropiado de las necesidades, virtudes y posibles deficiencias del usuario que deberían ser los pilares de todas las etapas del proceso de diseño (ISO 9241-210 [61]). Por lo tanto, es importante disponer de un conjunto bien definido de variables, de manera que se pueda especificar lo que un usuario puede hacer potencialmente al usar el sistema, qué se puede esperar de él, cuáles son sus necesidades, y cuál es la mejor manera de prevenir y tratar los posibles errores. Además, las clasificaciones de usuarios también facilitan el estudio y el trabajo del diseño centrado en el usuario.

Por último, hay un factor temporal asociado a cada clasificación realizada, y que permite reflexionar sobre cómo la tecnología ha estado cambiando los hábitos de la población de usuarios, mostrando avances colectivos, por un lado, y problemas, por el otro, y que son contemporáneos al momento en que se realizó la clasificación. Y a la inversa, también pueden reflejar cómo la evolución de los hábitos del usuario y de la sociedad ha influido en la dirección en que ha evolucionado la tecnología.

2.3.2. El reducido número de publicaciones recientes sobre clasificaciones de usuarios

A pesar de su utilidad, las clasificaciones de los usuarios no siempre han sido de gran interés para la investigación y el trabajo en el campo de la Interacción Persona-Ordenador. Desde su aparición a mediados de los años setenta y principios de los ochenta, y en paralelo con la emergencia de las computadoras personales, el número de trabajos explícitos sobre clasificaciones de usuario en revistas científicas ha ido disminuyendo ligeramente, con solo

unos pocos números en los años noventa y muy pocos en los años 2000. Todo este tiempo, aunque ha habido un aumento en el número y tipo de usuarios, debido especialmente a los factores y cambios de paradigma descritos en la introducción (sección 2.1), las publicaciones científicas relacionadas con la tecnología y las computadoras “miraron” en otra dirección.

Sin embargo, el hecho de tener un reducido número de publicaciones recientes sobre clasificaciones de usuarios nos ha animado a revisar más cuidadosamente la información disponible y refinar nuestra búsqueda. De hecho, uno de los principales hallazgos es que muchos de los conceptos utilizados en las primeras clasificaciones siguen siendo válidos, y varios de ellos incluso seguirían activos si se ajusta su terminología a la tecnología y a los escenarios de uso actuales. Por ejemplo, la categoría de los *usuarios intermedios* definidos por Martin [91] hacía referencia a aquellos usuarios que encontraban demasiado difícil de usar la interfaz por línea de comandos y delegaban su manejo en otro usuario más experimentado, el cual podría realizar la tarea en su nombre y proporcionarle la información necesaria. Esta y otras primeras descripciones de usuarios todavía se pueden encontrar y adaptar a diferentes contextos, como aquel en el que un niño descarga una aplicación para la *tablet* de sus padres, o un viajero experimentado que ayuda a una pareja de ancianos a imprimir sus tarjetas de embarque desde la máquina de auto-checking en el aeropuerto.

Finalmente, este reducido número de recientes publicaciones sobre clasificaciones de usuarios ha reforzado el valor de las primeras, cuyo fundamento teórico aún puede ser válido si se adapta a los nuevos escenarios de uso y a la evolución de la tecnología.

2.4. Revisión de la literatura sobre clasificaciones de usuarios

La revisión de la bibliografía que se realizará a continuación permitirá proporcionar una perspectiva cronológica de las distintas categorías de usuarios existentes, pero prestando especial interés a aquellas que consideren a usuarios sin experiencia, y que puedan resultar valiosas tanto para nuestro estudio y caracterización del UO, como para el de futuras clasificaciones. Dicha revisión se divide en dos subsecciones. La primera presenta y describe las variables empleadas en la definición de las distintas categorías, especialmente en las que se centran en usuarios infrecuentes e inexpertos, por compartir las dimensiones del tiempo y del conocimiento con el UO. El segundo subapartado describe otras

descripciones informales que se encuentran en fuentes de información no académicas ni científicas, como ciertas páginas web de Internet.

2.4.1. Revisión de las clasificaciones de usuarios inexpertos e infrecuentes en la literatura académica y científica

Las clasificaciones de usuarios se han basado tradicionalmente en variables específicas para agrupar a los usuarios por características diferenciables. En la bibliografía, es común encontrar similitudes entre los nombres utilizados para clasificar grupos de usuarios en categorías concretas e independientes. Sin embargo, es menos común encontrar una descripción formal de las variables utilizadas para realizar dicha división.

A continuación, vamos a explorar las diversas categorías de usuarios existentes, con el objetivo de diferenciar al UO de otras definiciones de usuarios con las que comparte una denominación parecida o, conceptualmente, las mismas dimensiones de tiempo y de experiencia (o conocimiento). Esta búsqueda y análisis han tenido en cuenta las siguientes definiciones:

- *usuario inexperto* es aquel usuario de un sistema que, o no tiene el suficiente conocimiento sobre la computadora en general y/o sobre el sistema específico que va a usar;
- *usuario infrecuente* es aquel usuario que no cumple con una frecuencia de uso previamente establecida, es decir, que no usa un sistema específico cierto número de veces en un período de tiempo determinado.

Se presentará, cronológicamente, un enfoque uniforme entre los diferentes autores intentando subrayar las coincidencias y divergencias entre las variables en las que basaron sus categorías en general, y de los usuarios sin experiencia e infrecuentes en particular. Se puede consultar un resumen en la Tabla 2.1 Variables utilizadas por los distintos autores para establecer sus categorías de usuarios. y en la Tabla 2.2 Revisión del concepto de usuario inexperto e infrecuente en las distintas clasificaciones de usuario.. Las primeras ocho descripciones, excepto Eason [37], se extraen de Cuff [34]. La investigación de Cuff se considera un trabajo seminal que ha influido fuertemente en desarrollos posteriores sobre usuarios inexpertos o casuales. Él investigó y analizó las definiciones del término *usuario*

casual realizadas por otros autores, introduciendo nuevas características y pautas para el diseño.

En la bibliografía analizada hay varias clasificaciones de usuario ampliamente aceptadas. Hemos seleccionado las siguientes porque reflejan bastante bien “la escena” existente en la bibliografía y dibujan una imagen más real de los valores y conceptos utilizados para establecer sus categorizaciones de usuario. Siempre que sea posible, se han establecido las variables en las que tales clasificaciones basan sus criterios.

En la primera clasificación analizada, Martin [91], se utilizaba la frecuencia de uso del sistema como la variable que permitía realizar la clasificación de los usuarios. Describía el uso de la aplicación de manera *intermitente*, porque los usuarios, en aquella época, tenían más probabilidades de estar realizando otras tareas que las basadas en el uso del ordenador. Esta clasificación se creó en los años en que las computadoras no estaban tan extendidas como hoy día, y la mayor parte del trabajo consistía en tareas no automatizadas y tenían un carácter más bien electromecánico, manual o verbal. La capacitación en el uso de sistemas informáticos específicos era escasa o inexistente, y se recomendaba diseñar la interfaz para que resultara lo más natural e intuitiva posible para evitar la confusión del usuario y el riesgo de rechazo al sistema.

Codd [30] definió un *usuario casual* basado en la desigualdad existente en la frecuencia de las interacciones con el sistema. Las razones laborales o sociales fueron excluidas de las motivaciones para tal uso. Este usuario no estaba versado en computadoras, programación, ni en ningún aspecto técnico del procedimiento.

Mann [88], contrariamente a la práctica común en ese momento, argumentó que el lenguaje de comandos debería dirigirse solo a profesionales o usuarios intensivos, que tenían experiencia en su uso. Por lo tanto, no se recomendaba el lenguaje de comandos para usuarios sin experiencia en informática, porque no solventarían los obstáculos que encontrarían al usar el ordenador.

Shapiro and Kwasny [150] definieron al *usuario casual* en base a la falta de familiaridad con una parte o el sistema completo. Se definió como un usuario poco frecuente al que no le gustaban las entradas y salidas del sistema breves y poco explicadas, como las indicaciones de tipo sí/no ni los menús imprecisos. Shapiro y Kwasny proponían aplicaciones que se pudieran entender en el lenguaje natural para explicar la parte

desconocida del sistema a dichos usuarios casuales, pero también a usuarios frecuentes que desearan adquirir el conocimiento necesario para poder usarlas de manera rápida y eficiente.

Zloof [184, 185] describió, en primer lugar, al *usuario no profesional* como aquel que no tenía ni ordenador ni conocimientos matemáticos. Tres años más tarde, redefinió el concepto para aquellas personas sin experiencia en programación pero que podían ser profesionales en otro campo no relacionado con la informática. A diferencia de Codd [30], el trabajo y la familiaridad con la aplicación eran las motivaciones para el uso de la tecnología. Ese usuario tenía que estar preparado para aprender lenguajes formales y modelos relacionales. También enumeró ejemplos, basados en la profesión, a los que normalmente pertenecerían los usuarios casuales: secretario, empleado (clerk), ingeniero y analista.

Kennedy [71] hizo uso del concepto de *computer naïve* para referirse al usuario que tenía un conocimiento limitado del sistema, y que se basaba en registros, listas o archivos. Cuff [34] describió las implicaciones de tal definición añadiendo que el modelo mental de dichos usuarios se basaba en conceptos previos a la computadora, identificando así un aspecto clave de estos usuarios casuales. La familiaridad con el funcionamiento del sistema y el entrenamiento permitiría evolucionar el modelo mental original.

Eason [37] definió al *naïve computer user* como aquel que necesitaba ser ayudado por la computadora al realizar una tarea. Este tipo de usuario no tenía un *conocimiento* profundo ni de la tecnología ni del sistema en cuestión y probablemente no pretendía adquirir dicho conocimiento. Además, buscaba minimizar el tiempo de aprendizaje y el esfuerzo para usar el sistema. Los *intereses* y aspiraciones de la mayoría de estos *naïve computer users* radicaban en el trabajo que tenían que realizar, y no en la herramienta que estaban obligados a emplear. Eason afirmó que estos *naïve users* podían estar representados por personas de diferentes trabajos, como gerentes, empleados, ingenieros, miembros del público, científicos, controladores de procesos, y argumentó que poco se podía hacer para guiar al diseñador para satisfacer las necesidades de grupos de usuarios concretos. Eason usó el concepto de *frecuencia de uso* y el término *intermitente*, para definir como *naïve intermittent* a alguien que necesitaba que le recordaran los detalles del uso.

Lough and Burns [84] estaban en línea con la segunda definición de Zloof [184] donde los *usuarios casuales* eran profesionales en un campo distinto de la informática, como

gerentes, abogados o gestores. Sin embargo, establecieron una diferencia importante: dichos usuarios no querían *conocer* las complejidades del sistema y tampoco se les debía exigir que aprendieran modelos de datos, ni métodos, ni aspectos de la programación. Incluían también a los usuarios que utilizaban el sistema de manera aleatoria, por ejemplo, los cajeros de banca o los empleados de compañías de seguros, que seguían las mismas rutinas y tenían un conjunto bien estructurado de necesidades que les permitían realizar sus tareas y consultas, dado el carácter repetitivo de las mismas. Conviene destacar la notable influencia que la *frecuencia de uso* en los procedimientos de aprendizaje del uso del sistema tiene en esta categoría.

Bjerre [9] definió al *usuario casual* como aquel que usaba el sistema con poca *frecuencia*, sólo para extraer algunos datos, y que no necesitaba tener habilidades en programación.

Cuff [34] evitó catalogar explícitamente al *usuario casual* –“no se obtendrá ninguna definición de este estudio” (p. 164). En cambio, Cuff analizó las interpretaciones del *usuario casual* de otros autores, a través de las cuales propuso una lista de características que caracterizaban a este tipo de usuarios: la *frecuencia de uso*, el nivel de conocimientos del ordenador (*computer knowledge*) y *familiaridad* con el sistema (*interface knowledge*). De esta manera modeló una clase convenientemente etiquetada como *usuario casual*. A pesar de la variedad interna existente dentro de esta categoría de usuarios, también compartían importantes características que se concretaban en varios requisitos para el diseño de sistemas “pensados” para este tipo de usuarios.

Moran [102] presentó dos categorías principales de usuarios: *expertos* y *novatos*. La clasificación fue una división bidimensional, basada en las variables conocimiento del usuario y estructura de tareas. El conocimiento del usuario estaba relacionado con la frecuencia de uso del sistema y el nivel de habilidad del usuario. Por estructura de tareas, Moran se refería al rango de acciones que un usuario puede o no puede tomar, cuyo componente más representativo es la interfaz. Moran argumentó que el novato es vulnerable a muchas variaciones en la estructura de tareas, en contraste con el experto que apenas lo sería. Los novatos se centraban en cómo superar la tarea y cómo aprender el uso de la interfaz. Los expertos eran hábiles y eficientes en el uso de la aplicación y, en comparación con los novatos, apenas requerían carga cognitiva para hacerlo. Ambos tipos de usuarios tendrían que volver a usar la aplicación en el futuro. Su clasificación se basó

implícitamente en la *frecuencia de uso* y, explícitamente, en el *conocimiento del ordenador y de la interfaz* (o *estructura de tareas*).

Vassiliou and Jarke [169] basaron su clasificación en cuatro variables diferentes, agrupadas de dos en dos.

- Para definir el *conocimiento sintáctico*, concepto usado también por Shneiderman, emplearon las dos variables siguientes:
 - *la familiaridad con los conceptos de programación* (cuyos términos equivalentes hoy día podrían ser la familiaridad con los conceptos y patrones de la interfaz gráfica) de aquellos usuarios que no tenían miedo a las computadoras y habían adquirido habilidades lógicas y algorítmicas para la resolución de problemas,
 - *y la frecuencia de uso*, la cual ligaban directamente a la cantidad de entrenamiento o formación recibida (o que sería necesario recibir).
- Para tratar el *conocimiento semántico*, las variables empleadas eran:
 - el *conocimiento de la aplicación*, usado para medir la precisión del modelo conceptual que el usuario tenía sobre la estructura y el contenido de la base de datos,
 - *y la gama de operaciones*, o cantidad de tipos de consultas diferentes que quería hacer el usuario empleando el lenguaje de la base de datos.

El *usuario casual* era uno con un valor bajo en las cuatro variables descritas.

Rutkowski [140] distinguía entre el usuario *profesional* y el *novato*, en el contexto de la ingeniería y del mercado de productos. Aconsejó que las funciones complejas se asignasen a los usuarios con más experiencia: “*Las funciones más complejas pueden manejarse de forma más compleja, porque normalmente serán utilizadas por usuarios más experimentados*”. Además, también enumeró el tipo de usuario asociado a cada etapa del lanzamiento de un producto al mercado: el especialista técnico, el entusiasta, y el consumidor.

El trabajo de Carroll and Thomas [26] fue en la dirección de definir consistentemente la metáfora como un componente útil de la interfaz para todo tipo de usuarios, y especialmente

para aquellos con poca experiencia, como el usuario *ingenuo* (*naïve user*). Ellos destacaron la metáfora del escritorio, considerando que comparaba de manera efectiva las características del sistema con los objetos del lugar de trabajo físico habitual, como los archivos y las carpetas. Así, los *naïve o optional users*, entre los que consideraban al director de una oficina, se enfrentan a un nuevo sistema con modelos preexistentes de cosas o conceptos que ya conocían de su trabajo, de sus tareas y de las estrategias que seguían para solucionar los problemas cotidianos del trabajo.

Nielsen [115] propone un análisis tridimensional de los usuarios que hace distinciones en términos del *conocimiento del dominio*, de la *experiencia con ordenadores* y de la *experiencia con el sistema* o IU de la aplicación. Sin embargo, aclaró que esta última, la experiencia del usuario con respecto a la IU específica es la dimensión a la que normalmente se hace referencia cuando se habla de la “expertise” o experiencia del usuario. Para Nielsen, la del *usuario casual* es una de las tres categorías de usuarios más importantes, además de *novatos* y *expertos*:

“(...) [los usuarios casuales] son personas que utilizan un sistema de forma intermitente, en lugar del uso tan frecuente que se asume para los usuarios expertos. Sin embargo, a diferencia de lo que ocurre con los novatos, los usuarios casuales ya han usado antes el sistema, por lo que no necesitan aprenderlo desde cero, sino sólo recordar cómo usarlo, en función de su aprendizaje anterior”.

Nielsen también hizo referencia al *completo novato*, como aquel que no tenía ninguna experiencia previa en informática. Sin embargo, argumentó que, en ese momento, eran ya menos frecuentes que en años anteriores, debido a que la mayoría de usuarios tenían o habían manejado antes ordenadores, por lo que ya sabían cómo usarlos.

Marsden and Hollnagel [90], y más tarde Hollnagel and Woods [54], definieron al usuario accidental como “una persona que se ve obligada a usar un sistema o un dispositivo específico para lograr un fin, pero que preferiría hacerlo de una manera diferente, si hubiese una alternativa”. El usuario accidental ve la tecnología como una barrera que dificulta la consecución de su objetivo. Estos autores no consideran que dicho tipo de usuario sea necesariamente inexperto, ni infrecuente u ocasional.

Turoff [166] afirmó que una clasificación de usuarios juega un papel funcional en el diseño de sistemas, distinguiendo una gran variedad de usuarios: *novatos*, *casuales*, *experimentados*, *intermediarios*, *frecuentes*, *operadores*, *rutinarios*, *avanzados*, *solucionadores de problemas*, y *usuarios de tiempo real*. De toda esta amplia gama de categorías, las más relacionadas con el uso irregular y/o la falta de conocimientos o experiencia previa con las TIC son los novatos y los casuales. Su *novato* es aquel que está intentando aprender desde su primer uso del sistema, siendo dicho aprendizaje su objetivo. Turoff también consideró la *motivación* como un factor clave que decide si se realizará o no el esfuerzo de aprender, y que dependerá de cómo se presente el sistema al usuario. Para él, el *usuario casual* “*usará el sistema solo unas pocas veces a la semana, o menos*”. Estos usuarios son “*novatos continuos, en el sentido de que no retendrán mucho de lo que aprenden sobre el sistema durante estas interacciones*”. Su usuario casual no sólo es infrecuente, sino que, además, y lo que es más importante, “*no tiene ninguna ambición por dominar el sistema y a menudo prefiere que se le lleve de la mano para lograr lo que necesita hacer*”.

Shneiderman [153, 155], inicialmente, diferenciaba entre dos tipos de conocimiento con respecto a la interfaz de usuario: el *sintáctico*, que describe un conocimiento, dependiente del dispositivo, sobre cómo usar un sistema particular; y el *semántico*, independiente del dispositivo y relacionado con los conceptos del ordenador y los conceptos de las tareas. Más tarde, desde 2005, Shneiderman et al. [156, pp. 67-68, 157, pp.89-90] actualizaron esos dos tipos de conocimiento, que ahora pasaban a ser: el *conocimiento de la interfaz*, y el *conocimiento de la tarea*. En base a ello, dividen el espectro de usuarios en tres categorías distintivas: el *novato* y el *primerizo*, el *intermitente con conocimientos* (knowledgeable intermittent user), y el *experto frecuente* (expert frequent user):

- La primera categoría (*novice* o *first-time user*) incluye, tanto al *novato*, un usuario que conoce poco sobre los conceptos relacionados con la tarea o sobre los relativos a la interfaz, como al *primerizo* (*first-time user*), que sería un profesional en el dominio de la tarea, pero que tiene un conocimiento muy superficial de la interfaz. En ambos casos estos usuarios “*pueden llegar con cierta ansiedad que les impediría o dificultaría el aprendizaje sobre el uso del ordenador*”.
- La segunda categoría (*knowledgeable intermittent user*) agrupa a los usuarios que tienen conocimientos, pero que son intermitentes de una variedad de sistemas.

Tienen conceptos estables sobre las tareas y un conocimiento amplio sobre los conceptos de la interfaz. Por el contrario, “*pueden tener dificultades para retener la estructura de los menús o la ubicación de las funciones*”.

- La última categoría (*expert frequent user*) sería la del usuario experto, que usa frecuentemente el sistema, y que está familiarizado tanto con los conceptos de las tareas como con los de la interfaz, y que busca realizar su trabajo rápidamente.

Cooper [33] diferencia también tres tipos de usuarios: *principiantes*, *intermedios* (que son perpetuos) y *expertos*. La clasificación se basa en el conocimiento que tiene el usuario sobre el producto y su dominio de aplicación, en virtud de la frecuencia de uso. Sin embargo, también considera que la mayoría de los usuarios no son ni principiantes ni expertos ya que, con el tiempo, tienden a convertirse en usuarios intermedios, dependiendo de la frecuencia con la que usan la aplicación. Los *principiantes* quieren aprender y mejorar, por lo que pueden convertirse en *intermedios* muy rápidamente. A veces, los *intermedios* pueden usar el producto intensivamente, aumentando su conocimiento, alcanzando el nivel de experto. Por el contrario, si los *expertos* no usan la aplicación durante un largo período de tiempo, pueden olvidar porciones significativas de lo que sabían, convirtiéndose así en *intermedios*.

Para Gillingham [42], el primero en utilizar explícitamente el término ocasional, este tipo de usuario se corresponde con “*alguien dentro de una organización de servicios humanos que sólo necesitaría usar un sistema de información particular con una base ocasional más que de forma regular*”. Afirma que la idea específica que le ayudó a conceptualizar su usuario ocasional fue la definición del *usuario accidental* de Marsden and Hollnagel [90]. También especifica que “*donde los usuarios accidentales y ocasionales difieren es en que el usuario accidental no es necesariamente un usuario infrecuente. Un usuario accidental puede comenzar como novato, pero el uso frecuente de un artefacto o sistema lo llevará eventualmente a un nivel aceptable de competencia*”. Finalmente, Gillingham agrega que diseñar para usuarios ocasionales es similar a diseñar para usuarios accidentales, excepto que no se puede esperar de los primeros que, con el tiempo, lleguen a estar familiarizados con el sistema.

Tabla 2.1 Variables utilizadas por los distintos autores para establecer sus categorías de usuarios.

Author	Year	Parameters used to establish the classification of users					
		Frequency of use	Computer knowledge	Interface knowledge	Task domain knowledge	Motivation	Other
Martin	1973	x					
Codd	1974	x	x			x	Technical knowledge
Mann	1975						Programming experience
Shapiro & Kwasny	1975	x		x			
Zloof	1975, 1978		x	x		x	
Kennedy	1975			x			
Eason	1976		x		x	x	
Lough & Burns	1977	x	x	x		x	
Bjerre	1977	x					Programming skills
Cuff	1980	x	x	x			
Moran	1981	x	x	x			Task structure
Rutkowski	1982						Experience
Vassiliou & Jarke	1982	x	x	x			Range of operations
Carroll & Thomas	1982						Experience
Nielsen	1993	x	x	x	x		
Turoff	1997	x		x		x	
Shneiderman	1980	x	x		x		
Shneiderman & Plaisant	2005, 2010	x		x	x		
Marsden & Hollnagel	1996					x	Forced to use the system
Hollnagel & Woods	2005						
Cooper	2007	x		x	x	x	
Gillingham	2014	x					Organization's member
Carrillo et al.	2016			x			Prospective use

Tabla 2.2 Revisión del concepto de usuario inexperto e infrecuente en las distintas clasificaciones de usuario.

Author	Year	Inexperienced and/or infrequent user
Martin	1973	Infrequent use of computer
Codd	1974	Irregular interactions, such as occasional extracting of data, and not motivated, not versed in computers and technical aspects
Mann	1975	Naive user (vs. computer professionals and heavy users)
Shapiro & Kwasny	1975	Face unfamiliar new system and dislike prompts and imprecise menus
Zloof	1975	Non-programmer, motivated by job
	1978	
Kennedy	1975	Computer naive
Eason	1975	Naive user, without computer technology knowledge. Motivated only by job, seeks to minimize learning, time and effort.
Lough & Burns	1977	Professional in a field rather than computer, without need to learn data model or access methods
Bjerre	1977	Occasional extracting of data
Cuff	1980	No computer experience
Shneiderman	1980	Novice: no syntactic knowledge, little knowledge about computer semantics, professional on task domain and deduced prospective use of same application
Moran	1981	Novice with ensured prospective use of same application
Rutkowski	1982	Novice without complex functionalities
Vassiliou & Jarke	1982	Non-extensive familiarity and narrow range of operations intended, with low grade on Shneiderman's syntactic and semantic knowledge
Carroll & Thomas	1982	Naive: no domain experience and no training on data processing
Nielsen	1993	Novice: computer experience without application experience, need to learn interface use from the beginning
		Complete novice: novice without computer experience
Turoff	1997	Novice: learning for the first time a new system or a part of it
		Casual user: infrequent, without any ambition to master the system
Shneiderman & Plaisant	2005, 2010	Novice: no task knowledge or no interface knowledge and deduced prospective use of same application
		First-time: task knowledge + no interface knowledge and deduced prospective use of same application
Marsden & Hollnagel	1996	Accidental: forced to use a specific system or artefact (not necessarily inexperienced nor infrequent)
Hollnagel & Woods	2005	
Cooper	2007	Beginners without interest to learn or improve
Gillingham	2014	Occasional: infrequent user within a human service organization that use a particular information system

2.4.2. Revisión de los usuarios inexpertos e infrecuentes en otras fuentes

Los expertos en usabilidad también han utilizado, de manera más informal y en otras fuentes distintas a la bibliografía científica, términos relacionados con usuarios inexpertos e infrecuentes. Por ejemplo, [5] destacó las diferencias entre un *principiante* y un usuario casual: “un novato es alguien que acaba de embarcarse en un curso de estudio y cuya intención es convertirse en un maestro de ese tema. Un usuario casual es alguien que solo quiere hacer un trabajo y no tiene interés en dominar el sistema. Sus necesidades de información son muy diferentes”.

Reed [137] señaló que los usuarios frecuentes estarán más preparados o predispuestos a aceptar realizar un proceso de aprendizaje (seguir una curva de aprendizaje) que los usuarios poco frecuentes, y la importancia de la usabilidad para este último también es más significativa: “Una dinámica particularmente importante en situaciones como esta, donde cada usuario individual puede iniciar sesión sólo ocasionalmente, es asegurarse de que el sistema esté diseñado para atender a usuarios poco frecuentes. Puede que vayas a iniciar sesión en tu sitio web de banca por Internet todas las semanas; si es así, es probable que estés preparado para aceptar abordar una ligera curva de aprendizaje. Sin embargo, es probable que tengas menos paciencia para los sistemas a los que accedes sólo ocasionalmente. Sé que sería poco probable iniciar sesión sólo una vez al año para ver la declaración anual de pensiones, si el sistema fuera extremadamente difícil de usar.”

2.4.3. Análisis de las variables representativas utilizadas para definir usuarios inexpertos e infrecuentes

Del conjunto de clasificaciones realizadas por los autores considerados, descritas anteriormente (un resumen de las cuales se puede consultar en la **Tabla 2.1** y en la **Tabla 2.2**), se han extraído y enumerado una selección de las variables más comunes entre las consideradas por dichos autores. Dichas variables se describen a continuación para examinar posteriormente su idoneidad para su inclusión en la nueva caracterización del usuario ocasional (que realizaremos más adelante). Definimos pues un conjunto formado por las cinco *variables* más relevantes del trabajo revisado:

- *Frecuencia de uso (Frequency of use)*: la tasa con la que se utilizó un sistema concreto, durante un período de tiempo determinado, en el pasado.
- *Conocimiento del ordenador (Computer knowledge)*: el nivel de habilidad o la capacidad que tiene un usuario con respecto al uso de la tecnología en general, o de los sistemas informáticos, en particular.
- *Conocimiento de la interfaz (Interface knowledge)*: la familiaridad o conocimiento que tiene el usuario sobre la interfaz del sistema y de sistemas análogos.
- *Motivación (Motivation)*: la razón que provoca el uso del sistema.

- Otras: *conocimiento del dominio de la tarea, experiencia en programación, conocimiento técnico, ambición por dominar el sistema, gama de operaciones, y estructuras de tareas.*

2.5. Lecciones aprendidas de las clasificaciones de usuarios analizadas

A continuación, describimos varias lecciones aprendidas del proceso de revisión de las *clasificaciones de usuarios* existentes realizado en la sección anterior. Estas conclusiones están relacionadas con la importancia que tienen la *inexperiencia* y la *infrecuencia* en el uso de la tecnología en general, y de los sistemas interactivos en particular, así como sus implicaciones en el diseño de la Interfaz de usuario. Tanto dicha revisión como las lecciones aprendidas nos llevarán a hacer una reflexión crítica y a mostrar la necesidad de definir una nueva categoría de usuario en base a esos dos factores clave sobre el uso de un sistema.

2.5.1. Importancia de la inexperiencia

No existe un valor específico para la *inexperiencia* o la *incompetencia* en el dominio tecnológico correspondiente a las clasificaciones analizadas. Por ejemplo, originalmente era necesario un nivel mínimo de conocimiento de la sintaxis del lenguaje natural para trabajar eficazmente con los sistemas que lo usaban, como el lenguaje de comandos. A partir de ahí, resultó razonable asociar a los *usuarios novatos* cierta experiencia en lenguajes de programación, porque incluso el usuario más inexperto tenía que lidiar con comandos para extraer la información necesaria y ser capaz de trabajar con el sistema. Una de las excepciones fue Martin [91], quien consideró a los usuarios sin experiencia en programación. Los llamó *usuarios intermediarios (intermediary users)*, es decir, aquellos que tenían que delegar la tarea a otros usuarios con suficiente conocimiento. Sin embargo, los usuarios actuales no tienen que lidiar comúnmente con extracción de información compleja de bases de datos. Todavía existe la posibilidad de manejar bases de datos a través de lenguajes de comandos, pero ya han surgido y siguen apareciendo otras formas de acceso más fáciles, basadas en otros tipos de interacción, como mediante formularios o manipulación directa, que rompen la barrera del requisito de la experiencia en programación.

En contraste con las descripciones detalladas de los *usuarios intermedios* [141] y de los *expertos* descritos en la literatura revisada, las definiciones correspondientes a la categoría del *usuario novato*, si se encontraron, se caracterizaron por su incompletitud o su carácter poco formal (véase Eason [37] como notable excepción). Detrás de los términos *novato* (*novice*) e *ingenuo* (*naïve*) hay ligeras pero importantes connotaciones semánticas y funcionales que deben explicarse para entender las razones para clasificar a un usuario como *novato*. Los novatos generalmente se definen como aquellos usuarios sin conocimiento sobre el sistema. En su mayoría, están asociados con usuarios que están comenzando a usar un sistema con frecuencia, y se espera que estén dispuestos a aprender durante ese uso continuo. En los casos en que el usuario novato no tiene ninguna experiencia en TIC, sus características y potenciales requisitos no se recogen generalmente en las clasificaciones, analizadas, especialmente en aquellas diseñadas en la era de las bases de datos, y que se centraban en usuarios con conocimiento sobre la tarea o sobre el lenguaje de programación necesario para poder utilizar el sistema. Wilson [180] definió al usuario que no tenía ninguna experiencia tecnológica, yendo un paso más allá, refiriéndose a él como el que: “(...) puede ser no sólo tecnológicamente ingenuo, sino también temeroso de la tecnología”.

Según Coe [31], existen grandes diferencias entre cómo los *novatos* y los *expertos* perciben y usan las aplicaciones software. Por un lado, el modelo mental de los *novatos* se mantiene incompleto e inestable porque aún se encuentran en fase de aprendizaje. Generalmente están más centrados en cómo lidiar con la interfaz (en línea con Moran [102]). Además, su comprensión sobre las funcionalidades de la aplicación es aún incompleta, lo que sugiere que una guía explícita podría serle valiosa para poder construir un modelo mental más adecuado, así como proporcionándole ayuda y soporte en caso de errores.

Por otro lado, el *experto* tiene un modelo mental muy refinado, basado en su experiencia, y que le proporciona un buen mecanismo para detectar y tratar los problemas que le surjan durante sus interacciones, por lo que, a diferencia de los novatos, requieren menos cantidad de guía y ayuda.

Desde la perspectiva de la IPO, la usabilidad de la interfaz es importante para ambos tipos de usuarios. Para los novatos, la facilidad de uso es un paso indispensable para avanzar en la interacción. Para los expertos, la usabilidad viene representada principalmente por la

velocidad, la flexibilidad, y la posibilidad de acceso a toda la funcionalidad con poco esfuerzo. Citando a Hartson [49]:

“El dicho común ‘lead, follow, or get out of the way’ se puede aplicar al diseño de la interfaz para todo tipo de usuarios: ayudar a los Novatos a realizar las tareas; proporcionar buena retroalimentación (feedback) a los Intermedios; y no molestar ni frenar a los usuarios Expertos”.

2.5.2. Importancia de la infrecuencia

Otro aspecto que se puede destacar de las clasificaciones contempladas es el conjunto de términos diferentes que hacen referencia a la frecuencia del uso de un sistema (o a la ausencia de ella): *ingenuo* (naïve), *primerizo* (first-time), *novato* (novice), *frecuente* (frequent), *casual* (casual), *intermitente* (intermittent), *discrecional* (discretionary), *irregular* (irregular), *infrecuente* (infrequent), etc. Esto refleja la amalgama de conceptos encerrados en las diferentes terminologías utilizadas, así como la necesidad de realizar una clara organización de dichas categorías y de las variables que las definen, especialmente con respecto a los usuarios sin experiencia.

Entre las variables que pueden ayudar a definir el uso ocasional de la tecnología se ha observado la ausencia de cierta especificidad. La mayoría de los autores interpretaron la *frecuencia de uso* suponiendo que habría un uso repetitivo del sistema. La posibilidad de que un usuario no repita el uso de la misma interfaz en el futuro no queda definida formalmente por los valores de la frecuencia de uso, y esto puede tener un grave impacto en cualquier clasificación. En este documento, el término *frecuencia de uso* se refiere a hechos que ya ocurrieron en el pasado, es decir, una frecuencia de uso probada, que ya ha sucedido y es verificable. Por contra, el término *uso futuro* o *prospectivo*, se refiere al futuro, expresando su significado o valor en términos de probabilidad. El término *uso prospectivo* tiene implicaciones directas en los objetivos asociados a las diferentes frecuencias de uso. Lo más probable es que un *usuario frecuente*, que va a volver a usar el sistema en el corto plazo, esté más interesado en adquirir el nivel adecuado de pericia y habilidad que le permita disminuir los tiempos de interacción y encontrar la forma más efectiva y eficiente de lograr sus objetivos. Por el contrario, para el *usuario infrecuente* que no tenga garantizado el volver a usar el sistema en el corto plazo, la consecución de sus objetivos y el tiempo necesario para conseguirlos durante el uso actual se volverán su prioridad.

Moran [102] argumentó que, para los *novatos*, aprender el uso adecuado de la interfaz era más importante incluso que poder hacer la tarea. El tiempo empleado para hacer la tarea e incluso el poder realizarla quedaba, en su opinión, relegado a una prioridad inferior:

“El aprendizaje es, por supuesto, primordial para el novato, mientras que el tiempo necesario para terminar la tarea en ese momento, es secundario completar la tarea es su gran preocupación”.

Sin embargo, parece que este no es el caso en contextos donde el factor “aprendizaje” no tiene tanta importancia como el conseguir realizar la tarea en el tiempo necesario. Así, muchos usuarios sólo se preocuparán por conseguir terminar la tarea, especialmente en los casos en que el tiempo sea crítico, como ocurriría, por ejemplo, al comprar un billete de tren en una máquina de autoservicio justo cuando el tren que está a punto de partir. Así, puede surgir un conflicto al considerar este tipo de contextos. Por un lado, está el objetivo de realizar la transacción lo más rápido posible. Por otro lado, puede haber conciencia de que las interacciones futuras podrían ser más rápidas y eficientes si el usuario, durante la transacción actual, dedicase algún tiempo a aprender a realizar esa tarea adecuadamente. En el último caso, las diferencias individuales podrían explicar los distintos grados de disposición de estos usuarios a dedicar un tiempo extra a aprender; pero también podría influir en dicha disposición la incertidumbre sobre el número de interacciones futuras. Surgen pues algunas preguntas: ¿Qué sucede si ese uso prospectivo no va a suceder, o no con una probabilidad definida? ¿Qué pasa si ese uso es el primero y el último y, por lo tanto, único? En tales casos, aprender a usar convenientemente la aplicación no es tan importante como conseguir usarla en ese momento. Por lo tanto, la prioridad es lograr el objetivo puntual. Así, en el ejemplo anterior de la compra del billete de un tren que está a punto de partir, el tiempo que requiere la interacción es crítico. Lograr el objetivo, es decir, obtener el billete, se convierte en lo más importante, mientras “pararse a” aprender durante la interacción, que podría no repetirse en el futuro, se convierte en secundario (*“no tiene ninguna ambición por dominar el sistema y puede preferir ser llevado de la mano para lograrlo ...”*, Turoff [166]). Algo parecido podría suceder en otros escenarios como, por ejemplo, durante unas vacaciones “únicas” en la vida, en un lugar al que no está previsto volver, desaparecería la expectativa por aprender a reutilizar la infraestructura del transporte local, y aumentaría la necesidad de minimizar el tiempo de la transacción. La formación de impresiones seguiría siendo crítica en este contexto, por lo que una mala

experiencia para el usuario resultante de la pérdida del transporte podría tener consecuencias duraderas.

2.6. Síntesis: necesidad de una nueva caracterización del Usuario Ocasional

La revisión de la literatura realizada en este trabajo nos ha proporcionado una evolución cronológica de los usuarios inexpertos e infrecuentes, valiosa, tanto para la definición del Usuario Ocasional que pretendemos realizar, como para futuras investigaciones que utilicen clasificaciones de usuarios. Entre los diversos factores analizados, los que se han enfatizado por encima del resto han sido la *inexperiencia* y la *infrecuencia*. Por ejemplo, cuando los usuarios inexpertos se enfrentan a un nuevo sistema, es clave asociar las funcionalidades del sistema y elementos de la interfaz a los modelos preexistentes de las cosas y conceptos con los que el usuario está familiarizado, y esto se puede lograr mediante el uso de metáforas realmente representativas [26].

Una de las lecciones aprendidas es que el modelo mental del usuario inexperto [120] puede evolucionar, de uno basado en conceptos no relacionados con los ordenadores, a otro, explicativo y funcionalmente predictivo, a través del entrenamiento y la familiaridad con el sistema [34].

La otra lección es que, en el caso del uso ocasional, el hecho de no tener garantizada la existencia de futuras sesiones elimina la posibilidad de evolucionar el modelo mental a través de métodos de aprendizaje tradicionales, como podría ser el de *ensayo y error*. Por tanto, el diseño de la IU no debería realizarse en base a la existencia de otras sesiones, ni anteriores ni futuras, sino dirigiendo al usuario hacia el logro del objetivo que desea conseguir en la sesión actual.

También se han observado interconexiones entre los factores analizados y la ampliamente aceptada *clasificación de usuarios de 3 categorías (novatos, intermedios y expertos)*. Para Cooper [33], la frecuencia de uso determina hacia qué categoría gravita el usuario, tendiendo con el tiempo hacia la del *usuario intermedio*. La frecuencia de uso también fue señalada o destacada por Nielsen, quien declaró que el uso *intermitente* se basa en el aprendizaje de las sesiones anteriores, colocando a este usuario (y que él llamaba *casual*) entre el *novato* y el *experto*, fuera de la línea de pensamiento mostrada por los otros

autores. Más importante aún, la usabilidad puede tener un impacto directo sobre la frecuencia de uso, pudiendo convertir a un usuario ocasional en usuario “de una sola vez” (*one-time user*) si la interfaz está mal diseñada, o a un usuario *infrecuente* en usuario *frecuente*, si el diseño es el apropiado [137]. Además, la poca frecuencia de uso también influye en la ambición por “dominar el sistema”, según lo descrito por Turoff [166]: “[un usuario que] no tiene ninguna ambición por dominar el sistema y puede preferir ser llevado la mano para lograr lo que necesita hacer”.

Todas estas definiciones se basan en diferentes grados de valores para los términos relacionados con el usuario y el uso del sistema: *frecuencia*, *experiencia* y *ambición por dominarlo*. Sin embargo, demuestra la falta de consenso que existe cuando se utilizan informalmente estos términos para definir lo que es *infrecuente*, *irregular* u *ocasional*. Lo que realmente define al *usuario ocasional* es la ausencia de *conocimiento previo* y la incertidumbre sobre la *perspectiva de uso futuro*. Por lo tanto, esta categoría de usuario hay que colocarla fuera de la tradicional curva de aprendizaje (Figura 2.2). El UO se corresponderá con un punto *fuera de esa curva*, disociando los usos actuales de los que potencialmente puedan suceder en el futuro. Esto resalta el problema de su inclusión en la tradicional *clasificación de usuarios de 3 categorías*.

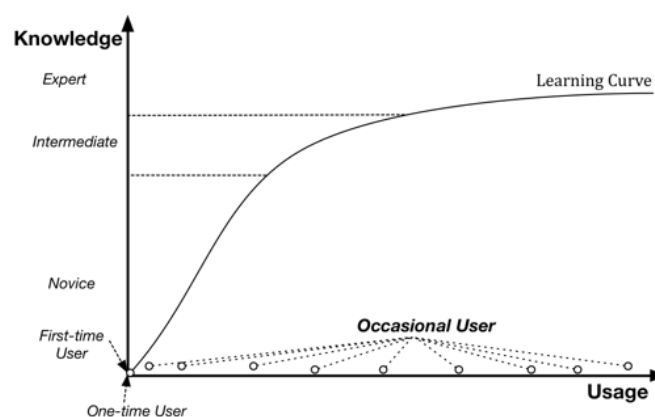


Figura 2.2 El Usuario Ocasional se sitúa fuera de la tradicional curva de aprendizaje asociada a la típica clasificación de usuarios de 3 categorías (novatos, intermedios y expertos). El Usuario Ocasional queda representado por un punto fuera de dicha curva.

2.7. El Usuario Ocasional: características y parámetros para su definición.

El UO de un sistema es un tipo de usuario sin el suficiente *conocimiento sobre la interfaz* de dicho sistema, y cuya prioridad es poder utilizarlo y conseguir sus objetivos sin apenas costo, en términos de tiempo y/o esfuerzo. Además, la *perspectiva de uso futuro* de la misma interfaz por parte del mismo usuario es desconocida y, generalmente, no se podrá garantizar dicho uso futuro. Por eso, requerir tiempo para aprender a usar la interfaz podría no ser apropiado, ya que al ignorar el usuario si volverá a usarla de nuevo, es muy probable que carezca de la voluntad para querer conocer bien el sistema y dominar la interfaz. Además, en ciertos casos, aprender de antemano a usarla podría no ser posible, debido al contexto de uso, como ocurriría en un sistema de aeropuerto para autenticación de pasaportes, donde el usuario sólo podrá usarlo en el mismo aeropuerto y en el momento que lo necesita usar. Los puntos claves del proceso de interacción para un UO serían la orientación y guía durante todo ese proceso, y la asistencia o ayuda en caso de error, sin requerirle, prácticamente, ningún conocimiento previo para poder utilizar la interfaz.

2.7.1. Variables específicas para la definición del UO

Como se indicó en los apartados anteriores, hay dos variables, una asociada con el conocimiento y otra con el tiempo, que resultan críticas para definir al UO [21]:

- 1) *conocimiento de la interfaz*
- 2) *perspectiva de uso futuro*

El primero de estos dos parámetros, el *conocimiento de la interfaz*, identificará la experiencia previa que el usuario tiene con la interfaz. En el caso del uso ocasional, su valor sería insuficiente para una conseguir una interacción óptima. Esto significa que, si el usuario ha tenido encuentros con la misma tecnología e interfaz, o con una parecida, el tiempo transcurrido desde la última interacción, las dificultades que experimenta mientras aprende y, en muchos casos, su falta de motivación, hacen que no sea prudente confiar en la memoria del usuario ni en el reconocimiento visual implícito como los únicos mecanismos para conseguir usar de forma satisfactoria el sistema. Por tanto, se recomienda considerar que el usuario se enfrenta a una interfaz desconocida.

La otra variable, la *perspectiva de uso futuro*, es una referencia explícita a la probabilidad de uso del mismo sistema, por el mismo usuario, en un futuro próximo. Debido a que, para el UO, la probabilidad de usar la misma interfaz en el futuro no se puede inferir con un nivel de probabilidad razonable, esto restringe la probabilidad a que sea siempre menor que 1. Dicho de otra forma, el UO no estará seguro de si volverá usar el sistema en un futuro cercano.

2.7.2. Diferencias con otras categorías de usuarios

Si intentamos incluir a este tipo de usuario en la típica *clasificación de 3 categorías*, no tendríamos éxito. Este tipo de usuario no encajaría ni en la categoría con la que podría estar más relacionado, la del *novato* o *principiante*, ya que su uso futuro de la interfaz es incierto, no sabe si volverá a usarla en el futuro. Esto coloca a la UO fuera de la tradicional curva de aprendizaje (Figura 2.2) asociada a *novatos*, *intermedios* y *expertos*.

Algo análogo ocurre cuando se compara con otras categorías menos extendidas, como la del *one-time user*, o la del usuario primerizo (*first-time user*). Existe una diferencia sustancial entre ellos y el OU.

En el caso del *one-time user*, existe la certeza de que ese uso de la interfaz será tanto el primero como el último (de ahí la etiqueta que se le asigna). Por lo tanto, la probabilidad de uso futuro es conocida, e igual a ‘0’.

Por el contrario, para el *first-time user*, la probabilidad de uso futuro también se conoce, pero sería igual a ‘1’, ya que es seguro que el usuario volverá a usar la interfaz en un futuro próximo (y por eso que se le etiqueta como usuario *primerizo* o *first-time*).

En el caso del UO, esas “certezas” no existen. Para ellos, y a priori, la probabilidad de uso futuro es desconocida y siempre menor que ‘1’. Esto significa que el UO:

- se convertiría en *one-time user* si sabe, a posteriori, que no volverá a utilizar la interfaz;
- se convertiría en *first-time user* si sabe, a posteriori, que sí repetirá el uso de la interfaz en un futuro próximo, y en cuyo caso, debería ser etiquetado como *novato* o *primerizo* (*novice*);

- o seguirá siendo UO, si la probabilidad de uso futuro continúa siendo desconocida, y siempre menor que '1' (y siempre que siga teniendo un conocimiento insuficiente de la interfaz).

Por lo tanto, en términos de la frecuencia de uso, es la certeza de la *probabilidad de uso futuro* de la interfaz lo que define y distingue al UO tanto del *one-time* como del *first-time user*. Así, la *incertidumbre* sobre el *uso futuro* hace evidente el problema de la correcta inclusión de este tipo de usuario en cualquiera de las categorías de usuarios tradicionales.

Las implicaciones de esta nueva categorización del UO deben incluirse en todas las etapas del diseño de un sistema cuyo potencial espectro de usuarios incluya a aquellos que no dependen ni del *conocimiento previo* de la interfaz ni de su *uso futuro* y, además, determinará la selección del *estilo de interacción* más apropiado para ellos (lo que tendremos en cuenta en el siguiente Capítulo).

2.7.3. Herramienta para clasificar a un usuario como UO: el árbol de decisión

El árbol de decisión del UO es una herramienta realizada con la intención de ayudar a los diseñadores de la IU en el proceso de clasificación de los potenciales usuarios de sus aplicaciones como OU. Dicho proceso de categorización se lleva a cabo verificando los valores correspondientes a las variables que definen al UO (Figura 2.3): el *conocimiento de la interfaz* y la *probabilidad de uso futuro* de la misma.

Estos valores deberían inicializarse y chequearse a priori, o lo que es lo mismo, antes de usar la interfaz del sistema.

La primera verificación a realizar es sobre el nivel de *conocimiento que tiene el usuario sobre la interfaz*:

- Si dicho *conocimiento de la interfaz* es suficiente, si el usuario ya conoce la interfaz y ya sabe cómo usarla convenientemente, ya no se debería clasificar como UO.
- Si es dicho *conocimiento de la interfaz* es insuficiente, se deberá realizar una segunda verificación, relativa a la *probabilidad de uso futuro* del sistema:

- En el caso de que dicha probabilidad sea conocida e igual a '0', el usuario se debería clasificar como *one-time user*. En principio, las mismas pautas de diseño de la interfaz que se apliquen para el UO podrían aplicarse para el *one-time user*, aunque para poder afirmarlo sería necesaria más investigación al respecto.
- En el caso de que la probabilidad de uso futuro sea conocida e igual a '1', el usuario se debería catalogar como *first-time user*.
- Solo cuando la probabilidad es menor que '1', y su valor explícito desconocido, el usuario se clasificará como OU.

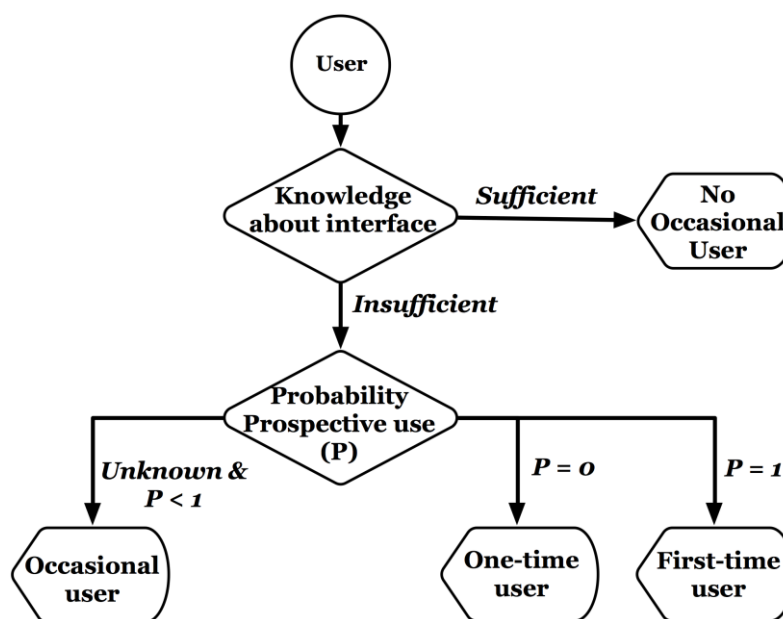


Figura 2.3 Árbol de decisión para clasificar a los potenciales usuarios como *Usuarios Ocasionales*.

2.7.4. Ejemplos de Usuarios Ocasionales

Para resaltar la importancia de este tipo de usuarios, los seis ejemplos descritos a continuación muestran escenarios en los que un gran porcentaje de usuarios podrían clasificarse como *ocasionales*.

- En la mayoría de aeropuertos se están instalando escáneres de pasaportes como una forma alternativa de identificación del pasajero que llega del extranjero, a través de un proceso de autenticación automática del pasaporte. La máquina requiere que el

pasajero abra el pasaporte por la página donde está presente su información personal y su fotografía, con una orientación específica necesaria para que el sistema funcione. Sin embargo, hay una doble dificultad implícita en el proceso. En primer lugar, no es obvio qué página contiene la información específica entre las diversas páginas que tiene un pasaporte. En segundo lugar, la orientación correcta del pasaporte a escanear no es intuitiva porque en la mayoría de los casos la página de la fotografía debe estar boca abajo, pero orientada inversamente desde la perspectiva del usuario. Este es un ejemplo típico de *Self Service Technology* [8, 96, 97] donde el usuario inexperto debe saber de antemano cómo operar dicha singular interfaz. Un UO debería que ser capaz de realizar la tarea sin tener que conocer los mecanismos de autenticación y escáneres de aeropuerto.

- La empresa de muebles IKEA ofrece a sus clientes una aplicación web para diseñar virtualmente una cocina paso a paso: la herramienta *3D IKEA Home Planner* [57]. Los usuarios (clientes) especifican las dimensiones físicas, indican la forma de la cocina, y concretan el diseño colocando también las puertas y ventanas en las paredes que correspondan. Los usuarios tienen la oportunidad de elegir entre diversos artículos (productos) y decorar u ocupar el espacio con ellos. Una vez terminado, pueden imprimir el diseño de la cocina y la lista de productos elegidos, y/o guardarlos en el servidor de IKEA. También pueden solicitar en la tienda el asesoramiento de un comercial o experto sobre el diseño realizado de la cocina. Según la compañía, esta aplicación “*tiene una interfaz fácil de usar, pensada para diseñadores de cocina (usuarios) sin experiencia*”. EL UO debería poder ser capaz de diseñar virtualmente su cocina, por ejemplo, con la ayuda de una guía efectiva, y siguiendo los pasos que ésta le indicase, sin necesidad de tener que ser un experto ni en el diseño de cocinas ni en la venta del mobiliario correspondiente, y sin tener la necesidad de tener que repetir el uso del sistema para poder manejarlo adecuadamente en esa ocasión puntual.
- Muchos parques de atracciones reciben una gran cantidad de visitantes cada año. Disneyland Paris es un parque de atracciones francés que ofrece una aplicación móvil gratuita [35] que proporciona información sobre el acceso al parque, disponibilidad de los hoteles, y horarios de los espectáculos. Una vez en el parque, la aplicación permite ubicarse a los usuarios (visitantes), encontrar su automóvil en

el parking, conocer las ubicaciones y distancias exactas de atracciones, espectáculos, y restaurantes, los tiempos de espera en cola para cada uno, y personalizar alertas e itinerarios propios en el parque. Muchos de estos visitantes podrían ser UO, y deberían poder usar dicha app para moverse por el parque e interactuar con su atracciones y lugares de interés sin necesidad de tener que estar familiarizados ni con el funcionamiento de la misma, ni con la forma en que dichos parques de trabajan y se organizan.

- Cada año, la Agencia Tributaria española ofrece una aplicación informática gratuita con la que los contribuyentes puedan realizar y enviar su propia declaración anual de la renta. Este programa, conocido como PADRE (Programa de Ayuda a la Declaración de la REnta)[134], teóricamente permite al contribuyente hacer su declaración anual de impuestos directamente desde su hogar, sin tener que hacer cola en las oficinas, y poder enviar la información necesaria usando Internet. En este contexto, un UO sería un usuario que quiere hacer su declaración de la renta sin tener que ser un experto en el dominio de la tarea (por ejemplo, sin tener que conocer toda la terminología ni los tecnicismos asociados con dicha declaración) y sin tener conocimientos ni experiencia sobre cómo usar dicha interfaz (por ejemplo, dónde están los campos que hay que completar del formulario, dónde está la ayuda, etc.). Por lo tanto, en la mayoría de las situaciones se podría completar el programa con una guía eficaz, y ser capaz de comprender y comunicar el resultado a quien sea pertinente: al contable familiar, a la Agencia Tributaria, a sus propios familiares, etc. En algunos casos, sin embargo, y como veremos en el capítulo siguiente, será necesario ser un experto para poder abordar ciertos casos más específicos o complejos.
- El museo de arte del Louvre de París [85] al igual que la mayoría de los principales museos del mundo (el Museo Británico, el Museo Americano de Historia Natural, el Hermitage ruso, etc.), importantes sitios arqueológicos (como Chichén Itzá, Machu Picchu, etc.) y otros lugares turísticos ofrecen aplicaciones móviles a modo de audioguía, que proporcionan información útil para sus visitantes. Probablemente, la mayoría de ellos sean visitantes ocasionales o incluso puede que esa sea la “única vez en la vida” que usen dicha aplicación. Ambos se beneficiarían de un diseño pensando explícitamente para el UO.

- Otros contextos y ejemplos de UO se corresponderían con aquellos usuarios que tienen que usar una aplicación o página web para comprar billetes para eventos que ocurren cada varios años, como los Juegos Olímpicos o el Mundial de fútbol.

2.8. Implicaciones y recomendaciones para el diseño de la IU

La caracterización del UO y, en particular, los valores de las dos variables que lo definen tienen implicaciones directas sobre el diseño de la IU y de la interacción recomendadas para ellos. Hay varios factores de la definición del UO que deben tenerse en cuenta en el diseño de la interfaz, como son la incertidumbre sobre el uso futuro, el proceso de olvido, o el diseño para uso puntual. A continuación, se concretan o destacan varios principios y recomendaciones, deducidos del proceso de revisión y análisis realizado, que consideramos fundamentales de cara al diseño de la interfaz. A ello le seguirá un resumen de dos estudios empíricos propios que respaldan dichas recomendaciones, y que serán presentados con mayor detalle en capítulos posteriores.

2.8.1. La incertidumbre sobre el uso futuro, y otros factores.

La incertidumbre sobre el uso se refiere a la dificultad que puede tener el diseñador para tener certeza sobre la perspectiva de uso futuro de su sistema. Por lo tanto, en estos casos, la posibilidad de uso ocasional del mismo se debería incluir, idealmente, durante el proceso de diseño, para evitar usos ocasionales desatendidos. Otros factores que pueden influir en el uso futuro de un sistema serían, entre otros, el *olvido*, la *motivación* y el *contexto* de uso.

Por *olvido* (o *forgetfulness*) nos referimos a que, incluso si el diseñador está seguro de que el usuario utilizará su sistema en el futuro, todavía queda la cuestión de si dichas sesiones estarán lo suficientemente cerca o condicionadas como para evitar que los usuarios olviden lo aprendido. Esto significa que el aprendizaje del sistema por parte del usuario a lo largo del tiempo no necesariamente tiene que ser siempre incremental, sino que puede disminuir.

Sobre el aprendizaje del sistema y de la interfaz también puede influir la *motivación* que tenga el usuario para aprender a usarlo. Incluso en el caso de un uso futuro asegurado, su influencia o repercusión en el aprendizaje no siempre estará clara si el usuario no está motivado por aprender.

Finalmente, el *contexto de uso* denota las condiciones del entorno que pueden impedir o dificultar el aprendizaje del sistema, incluso cuando el uso futuro está asegurado.

2.8.2. Recomendaciones y principios de diseño para un uso ocasional.

Las siguientes recomendaciones y principios [21] se basan en dos estudios empíricos propios que se resumen en la siguiente sección y se detallarán en capítulos posteriores (capítulos 4 y 5, respectivamente), así como en otras recomendaciones recopiladas de la revisión que sobre las clasificaciones de usuarios existentes se ha llevado a cabo en apartados anteriores (por ejemplo, [37, 166]):

- **Facilidad de aprendizaje** (*Learnability*): los típicos mecanismos de aprendizaje de las funcionalidades y del manejo de la interfaz por *retención* (memorización), *repetición*, o *ensayo y error*, son extremadamente limitados, ya que, para el UO no deben asumirse ni ser tenidas en cuenta posibles interacciones futuras (por lo que no se debe presuponer ningún periodo de aprendizaje o familiarización con la interfaz). En lugar de confiar en que el usuario aprenda a usar el sistema, se recomienda mostrarle cómo lograr sus objetivos en cada nuevo uso (consulte *Goalability* a continuación).
- **Consecución de objetivos** (*Goalability*): se refiere a la importancia de que el usuario consiga lograr sus objetivos en el mismo momento en que necesite usar el sistema y su IU, que es la razón principal que justifica por qué el UO lo utilizará. Dirigir o guiar al usuario hacia el logro de su objetivo es una prioridad, minimizando la necesidad de conocimiento previo, la ambigüedad, y la probabilidad de error.
- **Tiempo transcurrido** (*Elapsed Time*): tiempo que el usuario dedica al uso de la interfaz para lograr sus objetivos o, en otro caso, para recibir un resultado útil del sistema. Los posibles incrementos en el tiempo de uso sólo deberían permitirse para facilitar la interacción, la consecución de objetivos o la asistencia y guía.
- **Guía y asistencia** (*Guidance and Assistance*): siempre que sea posible se deben proporcionar mecanismos eficientes de guía y orientación durante el mismo proceso de interacción. Este aspecto se aborda para compensar la deficiencia o no existencia de varios de los habituales mecanismos de aprendizaje (descritos en el primer punto).

- **Recuperación y manejo de errores** (*Recoverability and Error handling*): se debería proporcionar un sistema de ayuda efectivo para los casos de error o imposibilidad de lograr un objetivo. Este aspecto está relacionado con el *feedback* que la interfaz proporcione al usuario, y tendrá influencia directa sobre la noción y opinión que éste se forme sobre el proceso de interacción.

Estas pautas recogen el hecho de que el diseño de la IU para el UO no debe esperar ni basarse en siguientes usos del sistema, ni requerir conocimiento previo de la interfaz. Requisitos relacionados con la “expertise” del usuario deben excluirse de entre las condiciones previas al desarrollo de interfaces para el UO, que generalmente desconocerá también los detalles de bajo nivel del sistema (versión de software, posibilidad de configuración personalizable, etc.). Además, las características de memoria y aprendizaje inherentes al UO requieren una interacción diseñada con mecanismos ágiles que hagan que el uso de la IU sea cognitivamente simple.

El potencial beneficio del diseño para estos UO es que el resto de usuarios con una amplia gama de experiencia y pericia puedan usar también dicha IU sin disminuir su nivel de efectividad, eficiencia y satisfacción, como se comprobará en los estudios introducidos y resumidos en el siguiente apartado.

2.8.3. Resumen de estudios empíricos que respaldan los principios y recomendaciones sobre el diseño de la IU para el UO.

A continuación, presentamos un resumen de dos estudios empíricos propios, los cuales se describirán, desarrollarán y analizarán con mucho mayor detalle en los capítulos 4 y 5, dentro ya de un contexto más concreto. Ambos están enfocados al uso ocasional de diferentes aplicaciones: el primero relacionados con aplicaciones para la compra de productos, y el segundo, una aplicación para el diseño de cocinas. A su vez, ambos trabajos incluyen un análisis comparativo entre dos IU diferentes, una desarrollada por nosotros en base a los principios que acabamos de presentar y recomendar para el UO, y la otra consistente en una versión comercial desarrollada por un tercero o basada en ella.

2.8.4. Estudio 1: uso ocasional de aplicaciones para la compra de productos, mediante tableta táctil

Este estudio [21, 94] implicó el desarrollo de dos prototipos de aplicaciones (una para la compra de un billete de tren, y otra para la compra de productos, según detallaremos más adelante), cuyas interfaces seguían los *principios* y recomendaciones dadas para el diseño de interfaces para el UO, con el objetivo de simplificar la toma de decisiones del usuario (principio 1: *Learnability*), y guiarlo y ayudarlo con el uso de la interfaz (principio 4: *Guidance and Assistance*). Dicha IU se implementó ofreciendo un mecanismo de recuperación de errores que incluye pasos específicos para permitir a los usuarios modificar sus decisiones (principio 5: *Recoverability and Error handling*). Una ruta de toma de decisiones sencillas conduce al usuario hacia su objetivo (principio 2: *Goalability*), que además se puede conseguir en un tiempo razonable e inferior al tradicional (principio 3: *Elapsed time*).

Estas interfaces se probaron para transacciones digitales que los usuarios realizaban de manera *ocasional*, midiendo el tiempo y el grado de consecución de objetivos (con el propósito de probar la verificabilidad de los principios 2 y 3). El dispositivo elegido fue uno portátil (tipo *tableta*), con entrada *táctil*. Los usuarios objetivo eran personas mayores, con poca o ninguna experiencia ni con ese tipo de dispositivos táctiles, ni con ordenadores o dispositivos tecnológicos similares. Cada participante tenía que realizar dos veces la misma tarea o transacción, consistente en la compra de un artículo concreto, empleando la misma tableta, pero usando cada vez una de las dos versiones de interfaces disponibles la prueba de evaluación. Una consistía en la propia aplicación web ofrecida por la empresa propietaria del servicio, y la otra en el prototipo desarrollado por nosotros aplicando los principios dados. Este proceso se realizó de forma equilibrada (*counterbalanced design*), es decir, alternando el orden de uso de ambas interfaces de un participante a otro.

El experimento se realizó en dos países distintos y en contextos diferentes. Una parte de la evaluación tuvo lugar en Dundee, Reino Unido, en la propia casa de cada participante; y la otra en Málaga, en un centro de salud y en una escuela para adultos. Participaron en total 11 personas mayores (con un promedio de 71 años de edad). Las interacciones de los participantes con el dispositivo se grabaron con una cámara de video. También se registraron todas las operaciones, y las preguntas y respuestas, durante y después de la interacción. Tras finalizar su prueba, cada participante rellenó un cuestionario sobre su

experiencia general, problemas particulares, y opiniones y recomendaciones sobre los dos tipos de interfaces utilizados.

Los resultados de la evaluación de las pruebas transnacionales abordaron la idoneidad de la interfaz para transacciones digitales ocasionales, como comprar un billete de tren (en el Reino Unido) o comprar un libro (en España). Las versiones específicas implementadas para el uso ocasional se inspiraron en las aplicaciones web ofrecidas por los propietarios (Figura 2.4 para el caso del billete de tren), pero diseñados teniendo en cuenta (como se detalló al principio) los principios recomendados para el UO (Figura 2.4 para el caso del billete de tren; la otra era la web de Amazon). Nuestras versiones aumentaban el número de pasos a realizar, en comparación con el sitio web de referencia (por ejemplo, de 7 pasos obligatorios a 14, en el caso de la compra del billete de tren). A pesar de este hecho, el tiempo total empleado en completar las tareas con ellas fue de más del triple de rápido que al emplear sus contrapartidas, los sitio webs de las compañías, a pesar de utilizar la misma tableta en ambos casos (por ejemplo, 3 minutos de media y $SD=0.5$, frente a un promedio de 10.1 minutos y $SD=0.68$, en el caso de la compra del libro, testada en España). Las respuestas al *cuestionario post-test* también evidenciaban que las interfaces desarrolladas para el UO era las preferidas para realizar las compras, aumentando la satisfacción del usuario.



Figura 2.4 Capturas de pantalla correspondientes a las dos interfaces comparadas para la adquisición de un billete de tren por *Usuarios Ocasionales*. (a) A la izquierda, la tradicional interfaz web. (b) A la derecha, la interfaz de usuario equivalente, construida aplicando explícitamente los principios de diseño dados para este tipo de usuario.

En el Capítulo 5 se retomará este experimento, en un contexto más específico y con mucho mayor detalle.

2.8.5. Estudio 2: uso ocasional de aplicaciones para el diseño de cocinas

El segundo estudio [21, 20] considera dos versiones de una misma aplicación de escritorio, cada una con una IU distinta (Figura 2.5). Ambas ofrecen la misma funcionalidad y permiten realizar las mismas tareas. Una versión incluye la típica *interfaz de Manipulación Directa* (MD) y está inspirada en el [57]. La otra ofrece una *interfaz guiada* (IG), basada explícitamente en los *principios* de diseño dados para el UO. Esta última versión ofrece y permite al UO abordar de manera jerárquica los distintos objetivos y subobjetivos que se puede plantear con el sistema (principio 2: *Goalability*), guiándolo paso a paso durante todo el proceso de interacción (principio 4: *Guidance and Assistance*), e indicándole tanto el “qué hacer” como el “cómo hacerlo”, sin requerirle prácticamente ningún conocimiento previo sobre el uso de la interfaz (principio 1: *Learnability*). Aunque, implícitamente, dicho proceso de guía ayudará a reducir notablemente tanto las dudas como la posibilidad de cometer errores, también incluye mecanismos explícitos para su corrección, como dar la posibilidad de retroceder en la jerarquía de pasos y objetivos activos, mediante la cancelación directa del objetivo en curso (principio 5: *Recoverability and Error handling*). Y todo ello en un tiempo razonable y, en la mayoría de casos, inferior al que se puede conseguir mediante la tradicional *interfaz de MD* (principio 3: *Elapsed time*).

Veinte participantes, ninguno de ellos con grandes conocimientos informáticos, tomaron parte en este estudio. De ellos, 18 fueron catalogados como “ocasionales”; sólo 2 se etiquetaron como “expertos”, por ser profesionales en el dominio de la tarea y trabajar diariamente con un software similar a nuestra versión de MD, aunque técnicamente más potente y complejo. Todos los participantes tenían que realizar los mismos 3 grupos de tareas, todas relacionadas con el diseño y amueblado de una cocina: 1) diseñar la cocina, especificando su forma, medidas, y ubicación de puertas y ventanas; 2) elegir el mobiliario a comprar y colocarlo en las posiciones dadas; y 3) realizar ciertas modificaciones (como mover y eliminar muebles) y operaciones finales (consultar presupuesto, ver en 3D, y encargar o guardar la cocina). Para mitigar el posible efecto de transferencia de aprendizaje por el uso de una interfaz antes que la otra, se fue alternando el orden de uso de las mismas entre los participantes (*counterbalanced design*).

Los *datos empíricos* recopilados fueron, por un lado, el tiempo (en segundos) empleados en completar cada uno de los 3 grupos de tareas considerados, y por otro, el número de

incidencias o errores detectados, tanto el total como diferenciando entre leves, moderados, y severos. Los tests estadísticos realizados, mostraron diferencias significativas, tanto en tiempos como en número de errores, a favor de la IG. Es decir, los participantes completaban las tareas en menor tiempo y con menos dudas y errores al utilizar la IG que con la alternativa, la tradicional interfaz de MD. El estudio también incluía *datos subjetivos*, recopiladas mediante dos tipos de cuestionarios. El primero, con preguntas concretas sobre la usabilidad de la interfaz, debía ser rellenado dos veces por cada participante, justo al terminar de usar cada interfaz. Finalmente, y una vez ‘experimentadas’ ya las dos versiones, los participantes rellenaban el segundo tipo de cuestionario, que requería la comparación directa entre ellas. Los resultados obtenidos de ambos cuestionarios eran evidentes y estadísticamente significativos, ya que todos los participantes (incluidos los expertos) mostraron una clara preferencia por la IG, tanto en la evaluación independientemente como en la comparativa directa. Lo que resultó más inesperado fue encontrar que todos los participantes (incluidos los profesionales en el tema), consideraban apropiada a la IG, no solo para el uso ocasional, sino también para el frecuente. En cambio, el 80% de los participantes consideraban a la *interfaz DM* apropiada para uso frecuente, pero no así para un solo uso, ni para el ocasional.

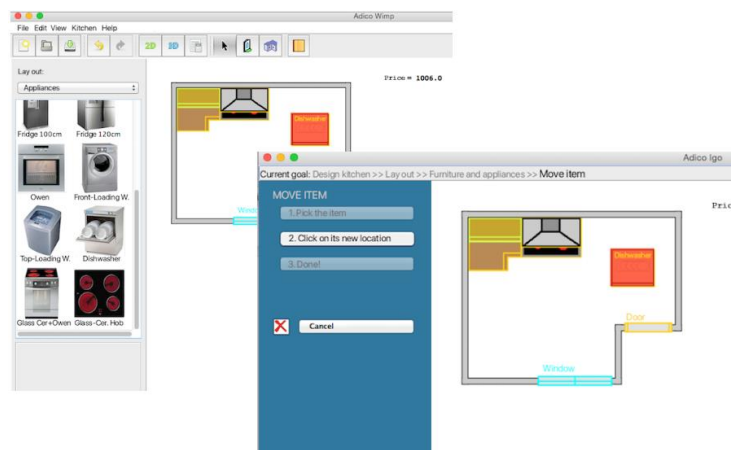


Figura 2.5 Capturas de pantalla de las dos interfaces testeadas para el diseño y amueblado de cocinas (durante el proceso de mover o cambiar la posición un mueble): a la izquierda, la interfaz de manipulación directa (MD); a la derecha, la interfaz guiada (IGO).

Como ya se ha indicado al comienzo de esta sección, los detalles sobre estos dos estudios que acabamos de introducir y resumir se tratarán con mucha mayor profundidad en los siguientes capítulos.

2.9. Resumen y Conclusiones

A pesar de que a lo largo de los años se han utilizado diferentes términos para etiquetar a los usuarios que utilizan un sistema con poca frecuencia y/o conocimiento de su interfaz, por ejemplo, *intermitente*, *causal*, *ingenuo*, *infrecuente*, etc., poco se ha hecho para categorizarlos y definirlos formalmente.

Uno de los objetivos perseguidos con esta tesis, y abordado explícitamente en este capítulo, era el de caracterizar nuevamente al *Usuario Ocasional*. Para ello, hemos realizado previamente una profunda revisión y análisis de las distintas *clasificaciones de usuarios* existentes en la literatura al respecto, prestando especial atención a aquellas categorías que consideraban a usuarios infrecuentes y/o inexpertos, por compartir con el UO las dimensiones del tiempo y/o del conocimiento, respectivamente. El resultado y conclusiones de dicho trabajo ha dado lugar a una propuesta de actualización de dichas clasificaciones de usuarios, mediante la inclusión de una nueva caracterización del UO, acorde con las nuevas tecnologías y escenarios de interacción. Se ha definido como un usuario que va a usar un sistema concreto sin el conocimiento suficiente de su interfaz (por lo que apenas se le debe exigir conocimiento previo sobre la misma), y que no sabe si volverá a usarla más adelante (por lo que no se debe presuponer la existencia de ningún periodo de aprendizaje ni de familiarización con la interfaz).

Los valores de las dos variables que definen al UO, un *conocimiento insuficiente de la interfaz*, y una *perspectiva de uso futuro* de la misma desconocida, lo distinguen de otros usuarios “non-knowledgeable”, como el *novato* y el *first-time user*, cuya adquisición (del conocimiento suficiente de la interfaz) se espera conseguir mediante el uso repetido de la misma. Se trata pues de a una categoría de usuario ortogonal con las ya establecidas, al deberse colocar fuera de la tradicional curva de aprendizaje correspondiente a los *novatos*, *intermedios* y *expertos*.

Finalmente, en este capítulo se han analizado las necesidades y características de este UO, así como las principales implicaciones de cara al diseño de la IU, y se han proporcionado unos principios y recomendaciones básicas al respecto, tanto para diseñadores de la IU como para la comunidad HCI en general, respaldados por dos estudios, tanto empíricos como subjetivos, en cuyos detalles profundizaremos en los capítulos 4 y 5.

En concreto, el siguiente capítulo se dedicará al elemento principal de esta tesis, un nuevo estilo de interacción guiada, y una metodología para su desarrollo y modelado que, como veremos, resultará especialmente adecuada para las características y necesidades de estos *usuarios ocasionales*.

Capítulo 3. Interacción Guiada por Objetivos

En el capítulo anterior se realizó una nueva caracterización del *Usuario Ocasional*, se analizaron sus principales necesidades y requisitos, y se concretaron unos principios básicos a tener en cuenta para el diseño de las interfaces dirigidas a estos usuarios. En este capítulo abordamos el objetivo principal de esta tesis, la propuesta de una nueva forma de interacción guiada, caracterizada por no requerir conocimiento previo ni curva de aprendizaje, por lo que resultará especialmente adecuado para el uso ocasional. Como soporte de este estilo de interacción, se especifica una metodología para el desarrollo y modelado de las correspondientes interfaces de usuario. Terminaremos analizando las limitaciones de esta interacción y situaciones para las que no es adecuada.

3.1. Introducción

Uno de los temas fundamentales dentro el ámbito de la *Interacción Persona-Ordenador* (IPO) y, más en concreto, de las *Interfaces de Usuario* (IU) es el de los *Estilos de Interacción* [157, 151]. Según Lorés et al. [83], mediante este término genérico se agrupan las “diferentes maneras en que los usuarios se comunican o interaccionan con el ordenador [133]”, entendiéndolo por *interacción* al conjunto de “todos los intercambios que suceden entre la persona y el ordenador [4]”. Aunque cada estilo de interacción ofrece su propia forma de organizar la funcionalidad del sistema, administrar las entradas del usuario y mostrar la información, se pueden considerar dos grandes enfoques a la hora de clasificar la forma de interactuar con los ordenadores [50]: el mundo *conversacional*, de naturaleza básicamente textual y secuencial, y el mundo de los *modelos*, donde priman los gráficos, las metáforas, y los diálogos asíncronos.

El *enfoque conversacional* se basa en el intercambio de diálogos textuales entre el usuario y el sistema: el usuario emite comandos textuales y el sistema responde también textualmente. Entre los principales *estilos de interacción* que siguen este enfoque se

encuentra la interacción basada en *Menús*, las *Interfaces de Línea de Comandos*, los diálogos de *Preguntas y Respuestas*, y la interacción basada en *Lenguaje Natural*.

En el otro gran enfoque, el del *mundo de los modelos*, se hace uso de gráficos y metáforas [26], como las ventanas, los iconos, los menús y los dispositivos apuntadores (WIMP, de sus siglas en inglés: *Windows, Icons, Menus and Pointers*) para ayudar al usuario con una gestión de objetos de la pantalla asíncrona y libre. Por último, la *Manipulación Directa* (DM) [154, 55, 157] de dichos elementos interactivos hizo este enfoque aún más familiar e intuitivo. Idealmente, el usuario ve objetos conocidos y predecibles, y sigue sus instintos para manipularlos, recibiendo retroalimentación o feedback inmediato mientras los utiliza: “No notas la computadora porque sólo piensas que estás haciendo la tarea, no que estás usando el ordenador” [121]. De hecho, el éxito de este enfoque basado en la MD ha eclipsado a los demás estilos de interacción.

Sin embargo, estas interfaces también suelen requerir cierto periodo de familiarización y que se siga una curva de aprendizaje que generalmente se recorre a través del uso repetido de la aplicación y de su interfaz. Moran [102] argumentó que, para los *novatos* o *principiantes*, aprender a usar la interfaz es más importante que poder realizar la tarea en sí. Sin embargo, como ya se argumentó en el capítulo anterior, este no es el caso del *Usuario Ocasional* (UO), que necesita alcanzar sus objetivos en el uso inmediato del sistema, sin conocimiento previo de la interfaz, ni periodo de aprendizaje ni familiarización con la interfaz.

En este capítulo abordamos el principal objetivo que nos planteamos con esta tesis, profundizar en el diseño explícito de interfaces dirigidas al uso ocasional. En base al trabajo realizado en el capítulo anterior, desarrollamos y presentamos una nueva forma de interacción guiada, dirigida a usuarios que no conozcan el sistema y no quieran o no puedan abordar una curva de aprendizaje, resultando especialmente adecuada para los *usuarios ocasionales*.

Esto requiere del análisis sobre qué enfoque o *estilo de interacción* resulta realmente apropiado para estos usuarios ocasionales. Como se verá en la siguiente sección, cada estilo afecta a la eficiencia y usabilidad del sistema y de la interfaz, pero eso no quiere decir que haya *estilos de interacción* buenos o malos. Todos tienen ventajas e inconvenientes,

dependiendo del tipo de usuario al que van destinados y, a menudo, los diseñadores de interfaces suelen usar combinaciones de ellos en vez de emplear sólo uno.

A continuación, en la sección 3.2, realizamos una revisión de los principales *estilos de interacción*. El propósito es encontrar los que mejor se adecúan a las necesidades del usuario ocasional y extraer las características más apropiadas de cada uno, de cara a la implementación de la forma de interacción que hemos desarrollado especialmente para estos usuarios y, en general, para aquellos que no quieran o no puedan abordar una curva de aprendizaje. La hemos denominado *Interacción Guiada por Objetivos, IGO* (o *Goal Guided Interaction, GGI*, en inglés), y la presentamos y describimos en las secciones 3.3 y 3.4. Finalmente, en el apartado 3.5, especificamos una metodología de desarrollo y una notación especialmente adecuada para realizar el modelado de este tipo de interfaces. En el capítulo siguiente, se pondrá a prueba esta forma de interacción a través de un estudio empírico con usuarios reales, obteniendo resultados favorables a dicha forma de interacción, tanto en efectividad (menos incidencias y errores) como en eficiencia (menor tiempo para completar las tareas) y satisfacción del usuario.

3.2. Los Estilos de interacción y el Usuario Ocasional

Un *Estilo de Interacción* es un término genérico empleado para agrupar las “diferentes formas en que los usuarios se comunican o interaccionan con el ordenador” [83, 4]. Los cinco siguientes fueron los diferenciados originalmente por Ben Shneiderman [155]: la *Interfaz por Línea de Comandos*, los *Menús*, los *Formularios*, la *Manipulación directa*, y el *Lenguaje Natural*. Hoy día siguen estando en vigor y son aceptados como los más representativos, y a ellos se les añade la *Interacción Asistida*, el *Point and Click*, y los diálogos de *Preguntas y Respuestas* [157, 151, 36, 183, 133]. A continuación, destacaremos las principales características de cada uno (resumidas en la Tabla 3.1) y, especialmente, las que mejor se adecúan a las necesidades y requisitos del usuario ocasional y de la forma de interacción que propondremos para ellos.

Interacción mediante Línea de Comandos. A pesar del tiempo transcurrido desde su aparición, este estilo de interacción sigue estando vigente hoy día [157, 151], siendo, en muchas situaciones, el preferido por usuarios expertos. Requiere que el usuario indique, explícitamente, los comandos que desea ejecutar, los cuales pueden consistir en palabras completas, nemotécnicos, o incluso en pulsaciones de teclas de función y/o caracteres.

Según [36] y [83], “la interfaz por línea de órdenes es potente, porque ofrece acceso directo a la funcionalidad del sistema. También es muy flexible: el comando normalmente tiene una serie de opciones o parámetros para modificar su comportamiento y puede ser aplicado a muchos objetos a la vez, haciéndolo útil para tareas repetitivas”. Sin embargo, esta flexibilidad y esta potencia también dan lugar a su gran inconveniente: su dificultad de aprendizaje [174]. “Para poder introducir las órdenes en el sistema hace falta tenerlas memorizadas, porque no hay una indicación visual del comando que se necesita” [83]. Por ello, para ayudar al proceso de recuerdo, los desarrolladores de estos sistemas deben cuidar especialmente los nombres que asignan a los comandos e intentar que haya consistencia entre comandos similares, al menos dentro del mismo sistema. En cualquier caso, este estilo de interacción es muy susceptible a errores, y ofrece una pobre gestión de los mismos, resultando mucho más apropiado para usuarios expertos que para novatos debiendo ser descartado para un uso ocasional.

Selección mediante Menús. De acuerdo con Lorés et al. [83], un *menú* consiste en un “conjunto de opciones mostradas en pantalla, que se pueden seleccionar, y la selección de una de ellas supone la ejecución de una orden subyacente y, normalmente, un cambio en el estado de la interfaz. A diferencia de la *interfaz por línea de comandos*, los usuarios tienen la ventaja de no tener que recordar ni palabras, ni sintaxis, sino reconocerlas en la lista de opciones disponibles”, basándose pues en uno de los principios básicos de usabilidad: reconocer mejor que recordar [116]. Además, dichas opciones se pueden mostrar de manera textual y/o gráfica, y deben ser auto-explicativas [124]. Las opciones del menú se pueden elegir con el ratón u otro dispositivo apuntador, mediante la voz, o mediante un atajo de teclado, que suele ser la opción elegida por los usuarios más experimentados. La estructura de menús de una aplicación se suele organizar de manera jerárquica, existiendo guías y directrices que ayudan y explican cómo hacer la estructura básica de un menú [172, 125]. Es fundamental que los diseñadores presten atención a las etiquetas que asignan a las distintas opciones [100] así como al número de opciones que incluyen [125]. Otro aspecto importante es determinar el orden en que se agrupan o muestran las opciones. La solución es trivial si existe una secuencia natural, como ocurre con los tamaños de una pizza o con los días de la semana. En caso contrario, se deberá elegir entre un orden alfabético, grupos de opciones relacionadas, colocar las más frecuentes primero, o incluso una combinación de estas posibilidades [157]. Este tipo de interacción sí es más adecuado para nuestros usuarios, ya que estructura mejor la toma de decisiones y apenas requiere experiencia, lo

cual acorta el aprendizaje. Sin embargo, por sí solo, tiene limitaciones en cuanto a su funcionalidad, siendo necesario combinarlo con otros estilos. De hecho, en cierta forma, también se ofrecerán opciones de menú en nuestra *Interacción Guiada por Objetivos*, en concreto, para lo que llamaremos *selecciones* del usuario (sección 3.5.4).

Formularios. Los *Formularios* de pantalla son similares a los formularios en papel, pero adaptados y mostrados en la pantalla, la cual se formatea o estructura en campos, en los que el usuario puede introducir o especificar información. Las interfaces basadas en formularios se utilizan principalmente para la entrada de datos, pero también pueden ser útiles en aplicaciones de recuperación de datos. Cada campo, salvo excepciones, debe tener una etiqueta o descripción que indique el tipo de dato a introducir, y que, tanto individualmente como en conjunto, suelen proporcionar un contexto al usuario, junto con la especificación de los posibles requisitos de entrada. Por tanto, los formularios suelen ser auto-explicativos, requieren poco entrenamiento, y facilitan el aprendizaje. Además, existen muchos estudios que proporcionan guías y directrices para mejorar su usabilidad y accesibilidad (por ejemplo, [146, 6]). Sin embargo, y especialmente si no se aplican dichas guías, pueden presentar algunos inconvenientes para usuarios inexpertos y ocasionales, ya que éstos podrían desconocer los posibles valores de las entradas, se les requiere cierta habilidad al teclear, saber cómo moverse por los campos, etc. En cualquier caso, si son necesarios para introducir datos, también se pueden usar en la interacción guiada que proponemos, aunque, generalmente, con un carácter más secuencial que el habitual (asemejándose, más que al típico *formulario*, a una *interacción basada en Preguntas y Respuestas*). Esto ocurre, por ejemplo, en la *interfaz IGO* elegida y desarrollada para realizar el proceso de evaluación de este tipo de interfaces, descrito en el siguiente capítulo.

Diálogos basados en Preguntas y Respuestas. Utilizan un mecanismo bastante simple para proporcionar información a una aplicación, en un dominio específico, combinando características de los *menús* y de los *formularios*. Al usuario se le hace una serie de preguntas, de una en una, y principalmente con respuestas de tipo sí/no, de opción múltiple, o textual. El usuario debe ir respondiendo o introduciendo información, mientras se va avanzando paso a paso. En este caso, es el sistema el que dirige la interacción. Al ser auto-explicativos, y fáciles de usar y de aprender, estas interfaces resultan más adecuadas para usuarios novatos y ocasionales, aunque, por otro lado, están limitadas en funcionalidad y potencia, por lo que sólo suelen ser apropiados para dominios restringidos [36].

Manipulación Directa (MD). La aparición y evolución de las interfaces gráficas de usuario, estrechamente ligadas al uso de la metáfora [26], han facilitado la creación de entornos interactivos en los que se ofrece al usuario en pantalla una continua representación visual, tanto de los objetos de interés como de las acciones que se pueden realizar con ellos, permitiendo, junto al uso del ratón u otro dispositivo apuntador, lo que se puede considerar una *manipulación directa* de dichos objetos. El término *Manipulación Directa* [157, 55] fue introducido por Ben Shneiderman en su discurso de apertura en el Simposio de Interfaces de Usuario de la NY University, y más explícitamente en [154], para describir un *estilo de interacción* que, según el propio Shneiderman, destaca por las tres siguientes características: “representación continua de los objetos y de las acciones de interés; cambio de una sintaxis de comandos compleja por la manipulación de objetos y acciones; acciones rápidas, incrementales y reversibles que provocan un efecto visible inmediatamente en el objeto seleccionado” [83]. En este caso, se sigue un paradigma *objeto-acción*: se selecciona primero un objeto (icono, ventana, texto, etc.) y después la acción a aplicarle.

Estas ideas fueron desarrolladas en el *Xerox Parc*, en torno a 1973 y aplicadas a los ordenadores *Xerox Alto* y *Xerox Star* y, posteriormente, ya en 1984, al *Apple Macintosh*, que fue el primer ordenador de MD que tuvo éxito comercial [67, 159].

La MD tipo WIMP es, en la actualidad, y especialmente para el ordenador de sobremesa, el entorno más común de *manipulación directa*. WIMP es el acrónimo de *ventanas, iconos, menús y apuntadores* (Windows, Icons, Menu and Pointers) [74, 36, 83]. Las *ventanas* son objetos gráficos, generalmente rectangulares, que se pueden mostrar incluso simultáneamente en la pantalla del ordenador, y en cada una de las cuales se puede realizar una tarea distinta. Se utilizan para mostrar documentos, aplicaciones, herramientas, diálogos, etc. Podemos ocultarlas, moverlas, colocar unas sobre otras, cambiar su tamaño, cerrarlas, o incluso transformarlas en *iconos*. Los *iconos* son pequeñas imágenes gráficas que, apoyándose en metáforas, representan de forma figurativa y más o menos realista, un significado o un objeto (por ejemplo, una ventana, un archivo, una carpeta, una papelera, etc.). El puntero es un pequeño símbolo gráfico o cursor que se muestra y se desplaza por la pantalla en correspondencia con el movimiento de otro dispositivo físico que maneja el usuario (como un ratón, un joystick, un trackball, o incluso su propio dedo), y mediante el cual se puede apuntar, seleccionar y manipular objetos mostrados en pantalla. Finalmente, y como ya se ha indicado antes, los *menús* presentan listas gráficas y/o textuales de alternativas, operaciones, o funciones a seleccionar, y que deberían tener un claro

significado y ser auto-explicativos. Aunque estas son las cuatro características principales y que dan nombre a la interfaz WIMP, también se utilizan muchos otros objetos y técnicas de interacción adicionales, como los botones, las barras de herramientas, las paletas, los cuadros de diálogos, etc.; son los denominados *widgets*.

Como ya se ha indicado, la MD es, con diferencia, el tipo de interacción más común, especialmente en las aplicaciones de escritorio y móviles, en realidad, ha eclipsado a los demás. Sin embargo, a pesar de su facilidad de aprendizaje y de fomentar la “exploración”, no está exenta de problemas [41, 55, 76]. Según el propio Shneiderman [157], uno de ellos es que el usuario necesitará conocer o aprender el significado de los iconos y componentes de la representación visual. La descripción que aparece sobre los iconos al poner cursor sobre ellos (*flyover help*) ofrecen sólo una solución parcial. Además, estas representaciones pueden ser engañosas: los usuarios pueden captarlas rápidamente, pero sacar conclusiones incorrectas sobre las acciones permitidas, e incluso sobre el momento en que deben ser realizadas. En algunos casos, especialmente el usuario ocasional, podrá necesitar la ayuda de otro usuario más experimentado, la lectura de algún manual, o un tiempo de entrenamiento con la interfaz para poder cumplir sus objetivos.

Point and Click. Este estilo está estrechamente relacionado con la MD-WIMP, pero con una filosofía de interacción mucho más simple. Consiste en apuntar a un elemento de la interfaz y, a continuación, pulsar o hacer un simple click con el ratón u otro dispositivo apuntador, siendo especialmente útil en sistemas con pantalla táctil. Está directamente relacionado y prácticamente limitado al *hipertexto*, la navegación web, y a las aplicaciones multimedia. Evidentemente, a pesar de su simplicidad, ideal para nuestros usuarios, presenta una funcionalidad y potencia muy limitadas si no se combina con otros estilos.

Lenguaje Natural. Una de las formas más atractivas de comunicarse con un ordenador, al menos, a primera vista, es mediante el *Lenguaje Natural*. El usuario desea que el dispositivo comprenda sus instrucciones, expresadas mediante palabras, expresiones o gestos cotidianos. Desafortunadamente, la ambigüedad del lenguaje natural hace que sea muy difícil de entender para una máquina. El lenguaje es ambiguo en varios niveles: en la sintaxis o estructura de una frase, en el significado de las palabras utilizadas, etc. A menudo, confiamos en el contexto y en nuestro conocimiento general para resolver estas ambigüedades. Sin embargo, esta información es difícil de proporcionar a la máquina. Además, para el usuario puede ser difícil determinar qué objeto y qué acciones son

apropiadas, y puede tener que aprender o suponer qué frases entiende el sistema, problemas que se acrecientan en el caso del uso ocasional. Sin embargo, la comprensión del lenguaje natural es tema de gran interés e investigación, en el que se están haciendo importantes progresos, y se está usando con cierto éxito en dominios restringidos, tales como información de aerolíneas, pérdida de equipajes, entrada de datos de registros médicos, y asistentes personales inteligentes [11, 107, 70, 129].

Si bien aún no existe la posibilidad de interactuar verbalmente con computadoras omnipresentes al estilo de “Star Trek”, los *asistentes inteligentes*, como *Siri*, *Alexa* o el *Asistente de Google*, se están volviendo cada vez más comunes. Sin embargo, el estudio realizado por Laubheimer and Budiu [79]) concluye que “*las interacciones con estos asistentes inteligentes están plagadas de problemas, que van desde la comprensión deficiente de los comandos hasta las limitaciones inherentes en la producción verbal*”. Poco después, Whinton and Budiu [172] descubrían que incluso los usuarios frecuentes los usan solo para una pequeña cantidad de tareas simples, lo que refleja la poca usabilidad de estos asistentes, lejos aún de satisfacer las necesidades reales de los usuarios, especialmente de los novatos y ocasionales.

Interacción Asistida. Este estilo de interacción, según Lorés et al. [83] “utiliza la metáfora del *agente* o del *asistente* personal que colabora con el usuario en el mismo ambiente de trabajo, y el usuario en vez de dirigir la interacción, trabaja en un entorno cooperativo en que el usuario y los *agentes* o *asistentes* se comunican, controlan eventos y realizan tareas. Este tipo de interacciones deberían permitir reducir el esfuerzo necesario para realizar ciertas tareas, debido a que, en el caso de la MD, para realizar una tarea hay que seleccionar objetos y elegir acciones”.

Dentro de este tipo de interacción, se suelen diferenciar entre *agentes* y *asistentes*. Los **Agentes** de la interfaz [99, 101, 143] son, según Lieberman [80], un tipo de software que el usuario suele percibir como un programa que pretende ayudarlo, y no tanto como una herramienta que complementa a la interfaz de MD. Citando a Lorés et al [83], “tienen que presentar algunas, quizás no todas, las características que asociamos con la inteligencia humana: capacidad de aprender, inferencia, adaptabilidad, independencia, creatividad, etc [80]. Según Pattie Maes [87], el usuario más bien colabora con el *agente* o le delega una tarea, en vez de ordenarle que la realice”. Además, “el *agente* puede leer la entrada que el usuario presenta en la interfaz, y hacer cambios en los objetos que se muestran en pantalla,

aunque no necesariamente uno a uno con las acciones del usuario. Puede observar muchas interacciones del usuario antes de hacer una acción, o con una sola interacción lanzar una serie de acciones y actuar en determinados períodos de tiempo que se le hayan fijado [80]. Con el uso de los *agentes* los usuarios se liberan de muchas tareas rutinarias. Son más discretos que los *asistentes*, trabajan en segundo plano y actúan por propia iniciativa cuando encuentran información que puede ser relevante para el usuario”. En particular, Shiaffino and Amandi [143] afirman, que los principales problemas que presentan los agentes son la necesidad de personalización (ya que cada usuario desea interactuar con un tipo diferente de asistente) y la reacción de los usuarios hacia interrupciones, errores y solicitudes explícitas de comentarios. Por otro lado, para poder interactuar con *agentes*, las aplicaciones deben ser *programables*, *controlables* y *examinables*. Según [83], “serán *programables* si permiten que el agente externo pueda ejecutar órdenes de la aplicación (ya sea mediante un lenguaje script o una API); serán *controlables* si pueden informar al agente de que el usuario es capaz de realizar una función por menú, por icono o por teclado; y serán *examinables* si el agente puede inferir las acciones que se están realizando con la interfaz, revisando periódicamente y comparando estados de las estructuras de datos”.

Por otro lado, en [83] se define a los *asistentes* como “entidades computacionales que nos asisten en el uso de aplicaciones existentes, exponiendo de una manera relativamente fácil, qué es lo que se ha de hacer, pudiendo entender palabras escritas o habladas, o acciones gráficas, e interpretarlas. Interpretar quiere decir que el asistente puede hacer acciones complejas u órdenes cortas. Es importante que los asistentes sean muy flexibles en la forma en que reciben las instrucciones. Pueden ser más flexibles que los menús y las macros porque el usuario, generalmente, sólo tiene que decir lo que quiere hacer. El usuario activa un asistente seleccionando órdenes y accionando un botón de asistencia, o haciendo una acción gráfica encima de él, siendo habituales en las actuales aplicaciones de MD”.

Wizards. A pesar de su nombre, los *Wizards* [98, 10, 15, 165, 176] no son realmente una forma especial de interfaz de usuario, sino que fueron concebidos más bien como herramientas de apoyo desde el punto de vista de una interfaz de MD, y solo tienen un rango particular de utilidad. Básicamente consisten en una serie de pantallas o cuadros de diálogo (generalmente modales) que van guiando a los usuarios, paso a paso, para la realización de una tarea concreta. En general, en cada pantalla se solicita información a los usuarios, ya sea mediante selecciones o rellenando campos de formulario. Después de

introducir los datos requeridos, los usuarios avanzan o retroceden por las pantallas haciendo clic en las opciones de navegación, como “Atrás” y “Siguiente”, y donde los pasos posteriores pueden depender de la información introducida en los anteriores. Aunque un *wizard* consta de varios pasos, todos ellos deben corresponder a una sola tarea. Este es el principio fundamental para su diseño: "un asistente, una tarea". Un dilema clásico en el diseño de la interfaz de usuario es si asignar el control al usuario o la computadora. Por lo general, las personas se sienten mejor cuando tienen el control, pero tener demasiada libertad puede causarles problemas y, a veces, hacerles perder el tiempo. Los *wizards* permiten que la computadora controle el flujo del diálogo, pero esta limitación suele ser liberadora para usuarios novatos y, especialmente, para los ocasionales. De hecho, los *Wizards* quizás sean la forma de interacción que más se acerca a las necesidades de estos usuarios. Como veremos en breve, comparten características y propósito con la *Interacción Guiada por Objetivos* que aquí presentamos, el de guiar o ayudar al usuario, paso a paso, a conseguir su objetivo. Sin embargo, como ya se ha indicado, los *wizards* están concebidos más bien como herramientas de apoyo de una interfaz de MD, y son invocadas por el usuario para realizar tareas concretas y bien delimitadas, y no tanto como una forma de interacción a aplicar a un sistema o aplicación completa.

Terminamos esta sección indicando que todos estos *estilos de interacción* quedan encuadrados dentro de lo que se conoce como *paradigma del ordenador de escritorio* (o *de sobremesa*). Según [83], los ***paradigmas de interacción***, “representan los ejemplos o modelos de los que se derivan los *estilos de interacción*. Son una abstracción de todos los posibles modelos de interacción, organizados en grupos con características similares.” Los principales *paradigmas de interacción* actuales (Figura 3.1) son, además del *ordenador de sobremesa* (el paradigma dominante), la *computación ubicua*, la *realidad aumentada*, y la *realidad virtual*.

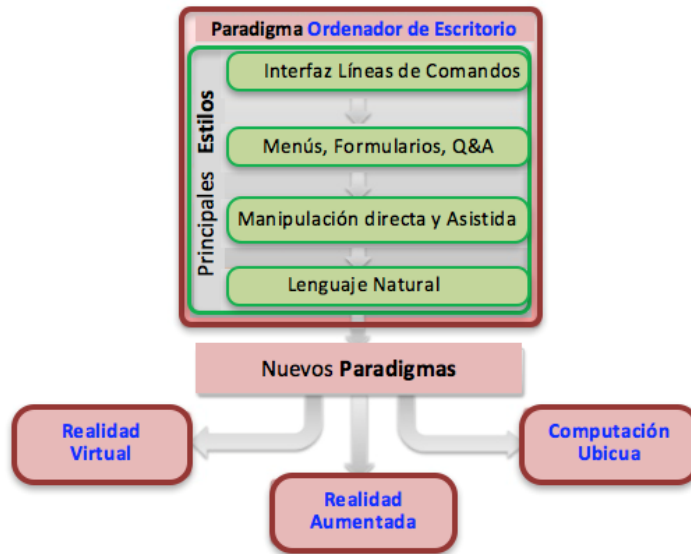


Figura 3.1 Principales Estilos y Paradigmas de interacción

Tabla 3.1 Descripción, características, y ventajas e inconvenientes de los principales Estilos de Interacción de cara a un uso ocasional.

Estilo de Interacción	Descripción, características, ventajas e inconvenientes
<i>Lenguaje de Comandos</i>	<ul style="list-style-type: none"> • Comunicar instrucciones al ordenador directa y rápidamente • Uso de teclas de función y/o caracteres, abreviaturas, o comandos completos. • Potente, rápido y eficiente, pero difícil de aprender y de recordar; requiere entrenamiento y aprendizaje.
<i>Menús</i>	<ul style="list-style-type: none"> • Solo hay que reconocer las opciones o comandos en una lista, donde se mostrarán textual y/o gráficamente, debiendo ser auto-explicativos. • Estructuran la toma de decisiones, y apenas requieren experiencia ni aprendizaje. Pero tienen funcionalidad limitada.
<i>Formularios</i>	<ul style="list-style-type: none"> • Estructura con formato, presentada en pantalla, con campos o huecos en los que el usuario introduce datos, debiendo ser auto-explicativos. • Poco uso de la memoria, proporcionan contexto; aprendizaje mínimo. • Se deben conocer los posibles valores de entrada, tener cierta habilidad en el tecleo, y saber moverse por los campos.
<i>Preguntas y Respuestas</i>	<ul style="list-style-type: none"> • Combina característica de los menús y de los formularios. • Se plantea, de una en una, una serie de preguntas, principalmente con respuestas sí/no, múltiples opciones, o textuales. • Fácil para usuarios novatos y ocasionales.
<i>Manipulación Directa</i>	<ul style="list-style-type: none"> • Se muestran gráficamente en pantalla los objetos que interesan al usuario, el cual siente que puede controlarlos directamente utilizando el ratón u otro dispositivo apuntador, realizando acciones rápidas, incrementales, y que generalmente, se pueden deshacer. • Se sigue un paradigma objeto-acción: primero se selecciona un objeto (icono, ventana, texto, etc.) y después la acción a aplicarle. • Es el estilo más extendido actualmente, da el control al usuario, ofrece acciones reversibles, favorece la exploración y el “ensayo y error”, y suele ser fácil de usar. • Sin embargo, es necesario conocer o aprender el significado de los iconos y componentes de la representación visual; además, se puede captar dicha representación, pero sacar conclusiones incorrectas sobre las acciones permitidas. Suele requerir algún entrenamiento y aprendizaje.
<i>Apuntar y Click</i>	<ul style="list-style-type: none"> • En los navegadores web y mayoría de los sistemas multimedia, prácticamente todas las acciones consisten en apuntar a un elemento de la interfaz y después hacer un clic de ratón • Muy simple y fácil, pero muy limitado en funcionalidad y potencia.
<i>Lenguaje Natural</i>	<ul style="list-style-type: none"> • En un contexto cultural o para un usuario con conocimiento del dominio de la tarea, es un sistema flexible y familiar. • Puede ser necesario aprender o suponer qué frases entiende el sistema, problemas que se acrecientan en el caso del uso ocasional.
<i>Interacción asistida</i>	<ul style="list-style-type: none"> • Utiliza la metáfora del <i>agente</i> o del <i>asistente</i> personal que colabora con el usuario. • Los <i>asistentes</i> son activados por el usuario para que le ayude en el uso de aplicaciones existentes; deben ser flexibles en la forma en que reciben las instrucciones. • Los <i>agentes</i> observan y aprenden del usuario, y actúan en segundo plano y por iniciativa propia, pero sin insistir en una solución si el usuario decide otra.
<i>Wizards</i>	<ul style="list-style-type: none"> • No son realmente un estilo de interacción, sino más bien herramientas de apoyo a la MD, y solo tienen un rango particular de utilidad. • Consisten en una serie de pantallas o cuadros de diálogo que van guiando a los usuarios, paso a paso, para la realización de una tarea concreta. • Ceden a la computadora el control del flujo del diálogo, pero esta limitación suele ser “liberadora” para usuarios novatos y, especialmente, para los ocasionales.

3.3. Fundamentos de la Interacción Guiada por Objetivos

Como ya se ha introducido anteriormente, la *Interacción Guiada por Objetivos* (IGO, o GDI, en inglés) [19, 20], es una forma de interacción cuya filosofía y propósito principal es el de guiar y conducir al usuario durante todo el proceso de interacción, tanto en lo que respecta a las tareas u objetivos a abordar, como a los pasos o elecciones a realizar en cada momento. Ha sido especialmente concebida para aquellos usuarios que no puedan o no quieran abordar una curva de aprendizaje, encajando perfectamente con las necesidades del UO.

Esto significa que, incluso si estos usuarios ya trataron previamente con la misma interfaz o con una interfaz análoga, hay una serie de factores que hacen que sea imprudente confiar tanto en el recuerdo de la memoria del usuario como en el reconocimiento visual implícito como principales mecanismos para conseguir usar la interfaz del sistema de forma satisfactoria. Algunos de esos factores podrían ser el tiempo transcurrido desde la última interacción, las dificultades que experimentan los usuarios en el aprendizaje y, en muchos casos, su falta de interés [54]. Se recomienda asumir que el usuario se enfrenta de nuevo a una interfaz desconocida, y que ésta no debería exigirle ningún conocimiento previo para usarla.

La IGO se fundamenta, en primer lugar, en los trabajos de Newell and Simon [111] sobre el *mecanismo de razonamiento humano para la resolución de problemas*. Su visión de la resolución de problemas se basaba en la descomposición del objetivo principal o general, en un árbol jerárquico de subobjetivos, cuyas ramas tendrían longitudes dependientes del grado de descomposición en subobjetivos. En las hojas del árbol nos encontraríamos con subobjetivos elementales, alcanzables mediante procesados de información u operaciones básicas.

Basados a su vez en estos trabajos, Card, Moran and Newell [17, 18] desarrollaron uno de los principales modelos cognitivos existentes, el *Modelo de Procesador Humano*, cuyo paradigma de partida consistía en concebir el proceso de interacción como una tarea de resolución de problemas, y describía también un modelo psicológico de los humanos constituido por tres sistemas interactivos: el *perceptivo*, el *motor* y el *cognitivo*, y cada uno de los cuales tendría su propia *memoria* y su propio *procesador*. El sistema *perceptivo* es

el que maneja los estímulos sensoriales externos, el sistema *motor* controla las acciones, y el sistema *cognitivo* suministra el conocimiento suficiente para conectar ambos.

Esta visión del usuario como un sistema procesador de información permite la formalización de las actividades, tanto físicas como mentales, que intervienen en esa labor de procesamiento, y dio origen a unos de los métodos de modelado y especificación de la Interfaz de usuario más extendidos, los modelos GOMS (Goals, Operators, Methods, and Selection rules) [18, 64, 65, 66], los cuales permiten describir las secuencias de comportamiento y el conocimiento que necesita el usuario para realizar las tareas y alcanzar su objetivos con un sistema.

Nosotros también consideraremos la interacción como una tarea de resolución de problemas y al usuario como un sistema procesador de información. Esto nos permitirá describir, tanto la jerarquía de objetivos que se pueden abordar (el “qué hacer”), como las actividades y operaciones concretas a realizar para conseguir dichos objetivos (el “cómo hacerlo”), de forma similar a como lo hacen las técnicas GOMS, en base a *objetivos*, *métodos*, *operaciones* y reglas de *selección*.

Sin embargo, la idea central y la novedad que aporta la *Interacción Guiada por Objetivos* consiste en que el usuario no tenga que dedicar tiempo a adquirir dicho conocimiento, ni buscarlo en un manual de usuario, ni en ningún otro sistema de ayuda, sino en que esté integrado en la propia interfaz, en una zona que denominaremos *Ventana de Objetivos* (VO), y que será la encargada de, en todo momento, ir proporcionando al usuario dicho conocimiento a medida que éste va interactuando con el sistema y avanzando por la jerarquía de posibles objetivos. Por tanto, esta VO será la que actúe como mecanismo de guía, indicando, continuamente y de manera secuencial, la forma de alcanzar el *objetivo* actual (y que siempre será mediante un *método* o secuencia de pasos, o mediante una *selección* entre un grupo de alternativas), de acuerdo con el árbol jerárquico de objetivos y tareas resultante de un previo y casi obligado proceso de análisis y especificación de la interfaz y del proceso de interacción.

Para realizar esta importante tarea del proceso de desarrollo de estas interfaces, podemos anticipar que nos basaremos en una de las variantes que existen dentro de la familia de técnicas GOMS: NGOMSL [72, 73]. Sin embargo, como veremos más adelante, dicha técnica, o no permite o no resulta del todo apropiada para modelar algunas situaciones y

aspectos propios de la IGO. Por ello, ha sido necesario realizar una adaptación y extensión de la misma, la cual describiremos en el apartado 3.5. De esta forma, a partir de la especificación de una interfaz realizada con nuestra extensión NGOMSL-IGO, será casi directo, tras un fácil proceso de “compilación”, obtener la guía, la VO, de la IU final [25].

Como ya se ha indicado, la IGO es de naturaleza conversacional y secuencial. No permitirá realizar tareas en paralelo. Por tanto, restringe la libertad que la MD ofrece al usuario, de manera parecida a como ocurre en los procesos realizados mediante *Wizards* [98, 10, 165, 176]. Sin embargo, la IGO puede verse como un superconjunto de los *Wizards*, ya que, entre otros aspectos, la guía y la interacción va más allá de unos pocos pasos a través de unos pocos diálogos. Se podría considerar como una “capa” que se añade sobre la interfaz de MD y que la convierte en un tipo de *interfaz asistida o guiada*: el usuario puede tocar, mover o interactuar con los elementos y objetos gráficos presentes en ella, pero solo cuando la VO así lo indique y permita. Se podría decir que la MD transfiere el control al usuario, mientras que la IGO lo guía. Además, el usuario también tendrá la posibilidad de deshacer operaciones para retomar objetivos y pasos previos, mediante ciertos procesos de cancelación que describiremos más adelante.

A continuación, pasaremos a detallar la estructura y funcionamiento de estas interfaces, y posteriormente, la metodología y notación que proponemos para su especificación. Las pruebas que justifican su validez, se realizarán en el capítulo siguiente.

3.4. La interfaz IGO

Con la finalidad de ofrecer una IU simple y consistente, y con una estructura bien definida y organizada, proponemos estructurarla en las tres partes o zonas siguientes (Figura 3.2): la *Ventana de Objetivos* (VO) o *Área de Guía*, la *Ventana de Trabajo* (VT), y la *Barra de Objetivos Activa* (BOA).

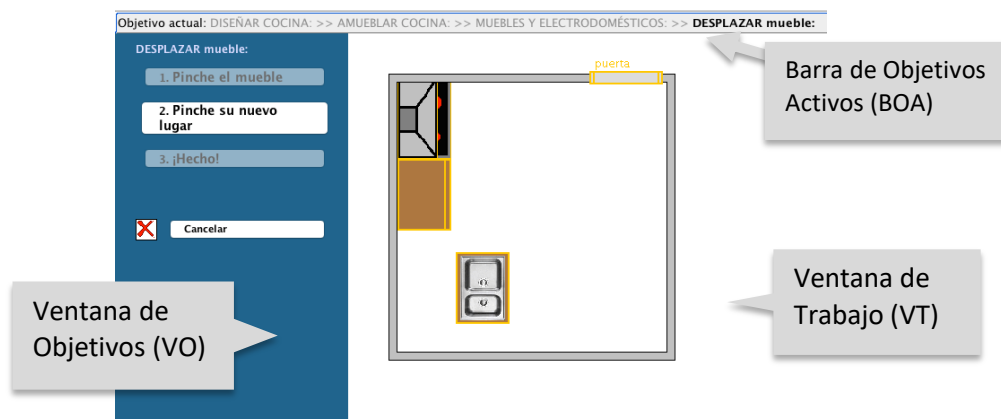


Figura 3.2 Estructura básica una *interfaz IGO*, en la que se puede apreciar sus tres componentes principales: la *Ventana de Trabajo (VT)*, la *Barra con la jerarquía de Objetivos Activos (BOA)*, y la *Ventana de Guía (VG)*, ofreciendo esta última el *método* que permite alcanzar el *objetivo* actual.

3.4.1. La Ventana de Objetivos

La VO es la zona de la interfaz (que nosotros proponemos situar en su parte izquierda) donde tendrá lugar el principal proceso de guía y de interacción. Es la zona de la interfaz donde se guía y se conduce al usuario por la jerarquía de *objetivos* a alcanzar y *acciones* a realizar, y donde, de manera secuencial, progresiva, y en cierta forma, “enmascarada”, se podrá acceder a las distintas funcionalidades del sistema, convirtiéndose en sustituta (en el caso más estricto de aplicación de la IGO, como se puede ver, por ejemplo, en la Figura 3.3) o en alternativa (en casos más flexibles de IGO) de la típica barra de menú de aplicación, de las barras de herramientas, de los menús contextuales, y de otros elementos que permitan acceder a las distintas funciones o comandos del sistema en las típicas interfaces MD-WIMP, y que ya no serían necesarios en una estricta *Interacción Guiada por Objetivos* (que será el caso testado y sometido a evaluación en el capítulo siguiente).

3.4.1.1. Métodos y Selecciones: Pasos y Alternativas.

La VO, en función del árbol jerárquico de objetivos contemplado para la aplicación, siempre presentará, bien el *método* (secuencia de *pasos*) o bien la *selección* (entre un grupo de *alternativas*) que permita al usuario alcanzar el *objetivo* que haya iniciado en cada momento. Por tanto, se podrán presentar los dos casos siguientes:

- a) **El objetivo actual se alcanza siguiendo un método.** En este caso, y como se puede ver en la Figura 3.3, la VO deberá mostrar el conjunto de *pasos* que componen dicho *método*. Se deberá indicar claramente cuál es el *paso* a realizar (el resto aparecerán deshabilitados o incluso no aparecerán, según se considere más apropiado), y el sistema sólo permitirá la realización de dicho *paso*. Se consideran 4 tipos de *pasos*: *Iniciación de objetivo*, *Realización de acción*, *Salto condicional*, y *Finalización de método*.
- ***Iniciación de objetivo.*** Se utilizará para indicar al usuario que debe iniciar e intentar alcanzar un nuevo *objetivo* (o *subobjetivo* derivado del actual), que de nuevo tendrá asociado otro *método* u otra *selección*, y cuya descripción deberá ser mostrada en la VO en sustitución de su contenido actual. Se trata pues de un paso puramente informativo y de preparación del usuario, de manera que cuando esté listo para iniciar el *objetivo* indicado, el usuario deberá explicitarlo pulsando sobre dicho paso (con el ratón u otro dispositivo apuntador, o utilizando un atajo de teclado, o cualquier otro mecanismo que se establezca para ello). Como resultado, la VO deberá cambiar su contenido, y pasar a mostrar ahora el *método* o la *selección* que permite alcanzar el *objetivo* recién iniciado (Figura 3.3). Esta activación explícita por parte del usuario, pero que podría realizarla de forma automática el sistema, además de servir de confirmación sobre el objetivo a iniciar, evita que se realice un cambio automático de contexto en la VO que pueda causar confusión al usuario, mejorando también este aspecto de la accesibilidad (WCAG [171] guideline 3.2).

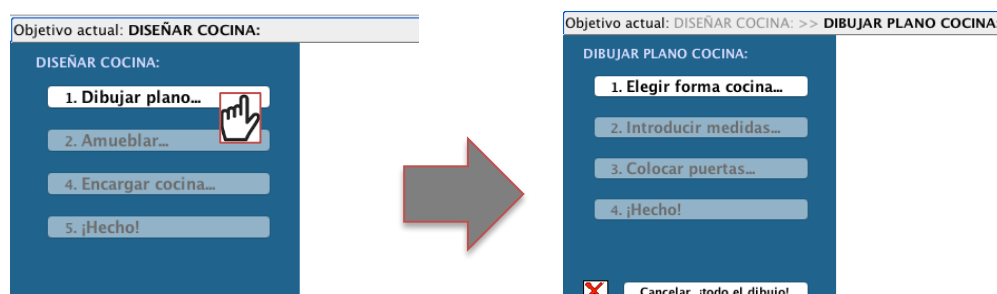


Figura 3.3 Ejemplo en el que el *paso* actual consiste en una *Iniciación de objetivo*. Al pulsar el usuario sobre dicho *paso*, la VO deberá actualizar su contenido, y pasar a mostrar el *método* (o la *selección*) que permita alcanzar el *objetivo* recién iniciado.

- **Realización de acción.** Este tipo de *paso* (Figura 3.4) requiere que el usuario realice una *acción* u operación concreta, ya sea perceptiva, cognitiva, motora, o una combinación de estas, pero que ya no será especificada con más detalle, bien porque ya no sea posible por tratarse de una *acción* primitiva, o bien porque se compruebe que el usuario sabrá realizarla sin descomponerla en más pasos.

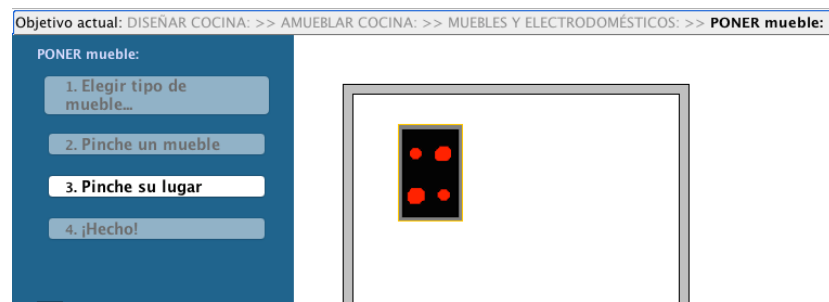


Figura 3.4 Ejemplo en el que el *paso* actual consiste en la *Realización de una acción*. En este caso, se trata de una acción a realizar en la VT (sin interacción física directa con la VO) cuya realización puede detectar el sistema, momento en el que éste pasará a habilitar el paso siguiente en la VO.

En aquellos casos excepcionales en los que el sistema no pueda detectar automáticamente la realización de este tipo de paso (por tratarse de una acción exclusivamente mental o cognitiva, ser externa al sistema, etc.), el usuario debería indicar explícitamente que ya lo ha realizado, pulsando sobre dicho paso en la VO, una vez lo haya realizado. Otras veces, la acción podría consistir en tener que pulsar, “simplemente”, sobre el paso que la describe (en la VO), por ejemplo, para dar un comando al sistema (como ocurre en la Figura 3.5), o como acción de confirmación, o como paso de seguridad, etc., sin tener que hacerlo recurriendo a elementos de la VT (que sería otra opción).

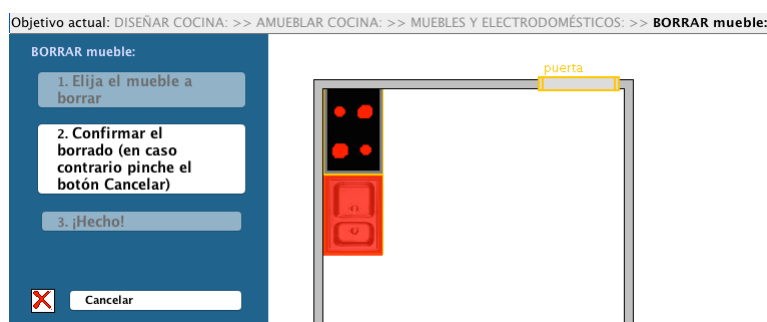


Figura 3.5 Ejemplo en el que el *paso* actual consiste en la *Realización de una acción*, en este caso, a realizar directamente sobre la propia VO, con la intención de activar un comando del sistema (eliminar un mueble). Para avanzar, el usuario deberá pulsar sobre dicho *paso* en la VO. En este caso, si se realiza el borrado del mueble seleccionado, todavía existiría la opción, ya en el siguiente paso, de cancelar dicha operación (el borrado del mueble).

- **Salto condicional.** Este tipo de *paso* implica una toma de decisión binaria por parte del usuario, al que se le dará a elegir entre saltar a otro *paso* concreto del *método* actual, o continuar con el siguiente (activando ambos pasos). De ambas opciones, y en función de la información que este paso suministre, el usuario deberá *activar* el *paso* deseado (como en la Figura 3.6). Se utilizará principalmente para crear bucles (dentro del método), ya sea para la repetición de secuencias de acciones o para dar la posibilidad de corregir errores dentro del método mostrado en la VO.

Existen, y se han probado, varias formas de implementar este tipo de paso. Una de ellas consiste en ofrecer explícitamente un *paso* previo que informe al usuario sobre la decisión a tomar, y que una vez tomada (y pulsado este *paso*), se activen ya los dos *pasos* correspondientes. Otra posibilidad (la finalmente implementada en la interfaz evaluada en el experimento del siguiente capítulo) consiste en acortar el proceso, mostrando, sólo a modo informativo, la descripción del paso condicional, a la vez que se ofrecen los dos pasos posibles para que el usuario elija el que desea llevar a cabo (Figura 3.6).

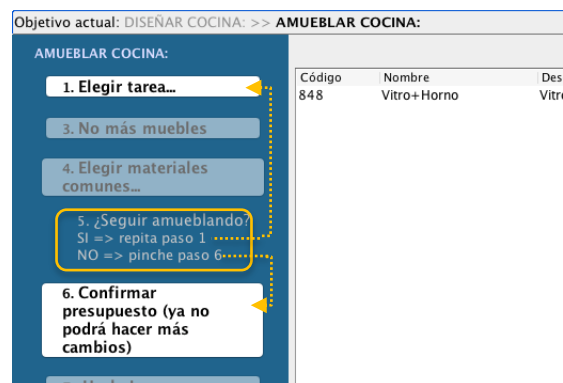


Figura 3.6 Ejemplo en el que el *paso* actual consiste en un *Salto condicional*. El usuario deberá decidir qué *paso* desea realizar, de los dos que se le ofrecen (continuar por el siguiente, o saltar al paso indicado).

- **Finalización de método.** Finalmente, y aunque se puede considerar opcional, se aconseja terminar todos los *métodos* con un *paso* que indique que ya se ha conseguido el *objetivo* actual, y tras el cual, se volverá al *paso* siguiente del *método* previo más cercano en la jerarquía de objetivos activa. De esta forma, si

el *paso i* de un *método A* deriva la interacción hacia otro *método B*, al terminar este último se retornaría de nuevo al *método A*, pero siendo ahora el paso actual a realizar el $i+1$ (Figura 3.7). De nuevo, la pulsación explícita de este paso por parte del usuario evitará que se realice un cambio automático de contexto (al actualizar el contenido de la VO) que pueda causarle confusión.

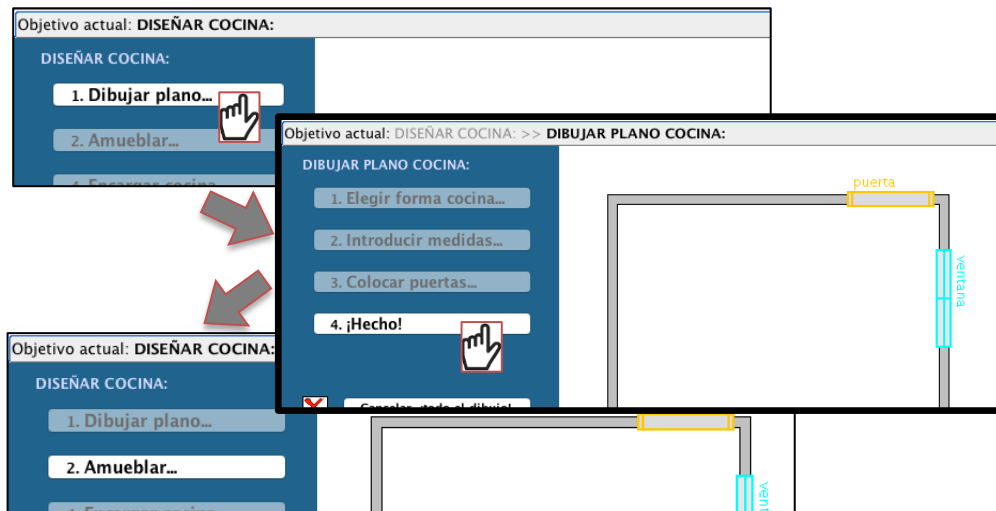


Figura 3.7 Ejemplo en el que se muestran 3 estados distintos de una interfaz IGO. En la imagen central, el *paso* a realizar implicará la *Finalización del método* actual. También se puede observar que se retornaría al siguiente paso del *método* padre (el siguiente *paso* al que desvió la interacción hacia el *método* que acaba de terminar).

- b) El **objetivo actual** se alcanza realizando una **selección** entre un grupo de *alternativas* u opciones. En este caso, la VO deberá mostrar dichas *alternativas*, de manera que el usuario pueda elegir una de ellas (Figura 3.8).

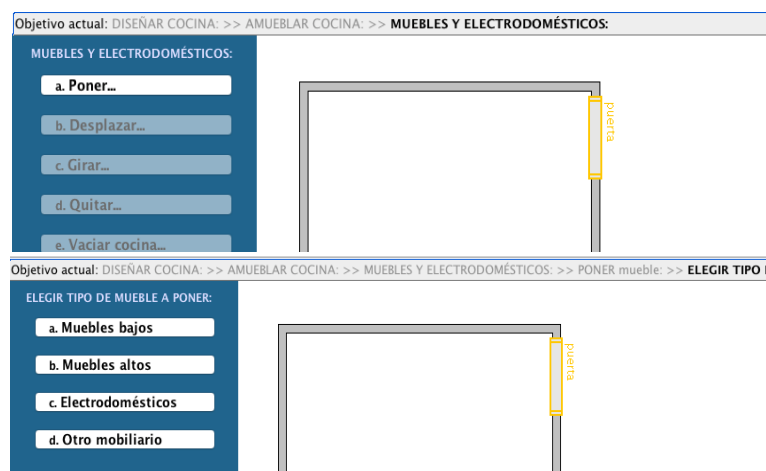


Figura 3.8 Dos ejemplos en los que para alcanzar el *objetivo actual* se debe realizar una *selección* (en la VO). En el primer caso (el superior), no todas las *alternativas* están habilitadas en ese momento.

Si se considera adecuado, también se podrán ofrecer las clásicas *selecciones* a realizar, no ya en la *Ventana de Objetivos*, sino en la *Ventana de Trabajo*. En este caso, habría que evaluar si es o no necesario proporcionar el *método* detallado a seguir para realizar dicha elección, o si es suficiente con indicarlo mediante un simple *paso* (como ocurre en la Figura 3.9).



Figura 3.9 Ejemplo en el que el *objetivo* actual no se alcanza realizando la típica *selección* IGO (en la *Ventana de Objetivos*), sino siguiendo los *pasos* de un *método*, y dónde el *paso* actual implica realizar la elección de una de las diversas opciones presentadas en la VT.

3.4.1.2. Cancelación de Objetivos.

Aunque se pueden considerar otras formas de cancelación (o de “deshacer operaciones”), la que proponemos por defecto (y la que se ha utilizado en la aplicación analizada y evaluada en el siguiente capítulo), es la que denominamos *cancelación a nivel de objetivo*.

En principio, el usuario podría querer cancelar la realización del *método* o de la *selección* mostrada en la VO o, siendo más precisos, del *objetivo* que se alcanza siguiendo ese *método* o realizando esa *selección*. Como se muestra en la Figura 3.9, esta posibilidad se puede reflejar en la interfaz añadiendo, tras el último *paso* del *método* actual, otro *paso* adicional, y que debería estar disponible a la vez que el *paso* que corresponda en ese momento. En caso de que el *objetivo* se alcanzase siguiendo una *selección*, se añadiría una nueva *alternativa* a las ya existentes (Figura Ejemplo M.5).

Siempre que se cancele un *objetivo*, la VO actualizará su contenido para volver a ofrecer el *método* (o la *selección*) asociado al *objetivo* padre (en la jerarquía de objetivos activa) de aquel que se acaba de cancelar, y al mismo estado y *paso* que desvió el flujo de la interacción hacia el *objetivo* cancelado. De esta forma, si el *paso* i de un *método* A deriva

la interacción hacia otro *método* B, al cancelar este último se retornaría de nuevo al mismo *paso* i del *método* A, deshaciendo todas las operaciones realizadas por el método B (Figura 3.10).

Sin embargo, en algunas situaciones el sistema actuará sobre otros sistemas externos sobre los que no tendrá un control completo. En esos casos, como, por ejemplo, enviar un email, será imposible "deshacer" o cancelar la operación (el mail ya se ha enviado). En la jerarquía de *objetivos*, un *objetivo no cancelable* representaría un nodo fijo de no retorno para el resto de la interacción con el sistema. Evidentemente, si se pasa por un *objetivo no cancelable*, al volver con el sub*objetivo* alcanzado al *objetivo* padre (tras pulsar el paso return with goal accomplished), éste pierde su posibilidad de cancelación. Estos casos se dan cuando es imposible retomar un estado previo del sistema una vez realizadas este tipo de acciones irreversibles.

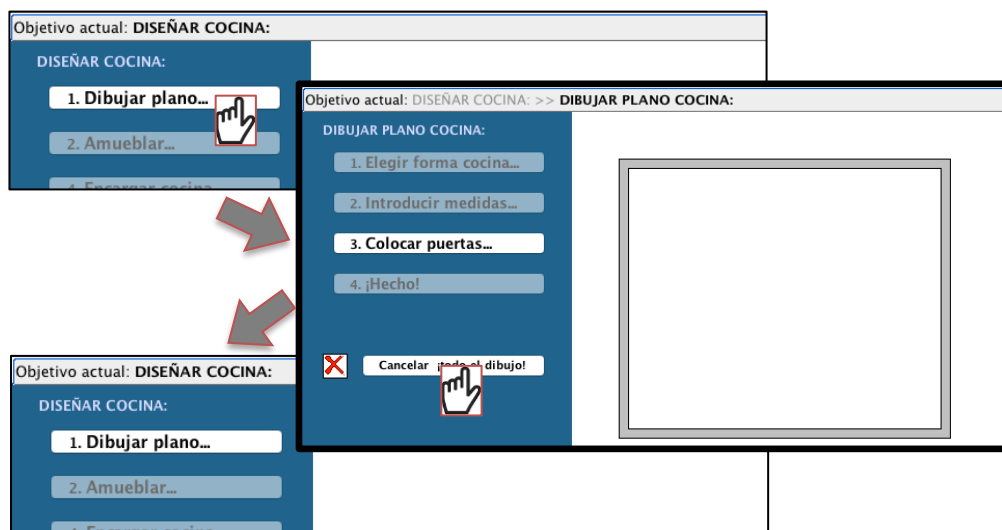


Figura 3.10 Ejemplo que muestra una interfaz IGO en 3 momentos distintos de uso de un sistema (los mismos que en la Figura 3.6). En la imagen central, la VO está ofreciendo el *método* que permite alcanzar un *objetivo cancelable*. En esta ocasión se puede apreciar que, si el usuario no realiza el *paso* actual, sino que elige *cancelar* el *objetivo*, se retornaría al mismo punto y al mismo estado del *objetivo* padre (que desvió la interacción hacia el *método* que acaba de cancelar).

3.4.1.3. Navegación hacia delante y hacia atrás.

Acabamos de presentar el concepto de *objetivo cancelable* y *no cancelable*. Se puede dar un paso más en este sentido y, aunque no ha sido testeado ni evaluado formalmente,

“complicar” algo más la interfaz y la interacción, con la finalidad de intentar hacerla más flexible y eficiente. Se trataría de ofrecer al usuario la posibilidad de “navegar” entre los pasos ya realizados, es decir, de retroceder al paso anterior o de avanzar hacia el siguiente, siempre que estos sean reversibles. Esto se podría plasmar en la interfaz colocando en la parte inferior de la VO los dos elementos interactivos correspondientes (por ejemplo, botones), como si se tratase de dos pasos más, disponibles en la VO. Sin embargo, esto podría romper la consistencia y sencillez del mecanismo básico de interacción. Habría que evaluar, según el nivel de expertise de los posibles usuarios, si es realmente adecuado.

3.4.2. La Ventana de Trabajo (VT)

Por lo general, las interfaces IGO destinarán la mayor parte de la interfaz a lo que llamaremos *Ventana de Trabajo* (VT). En esta zona de la interfaz se ofrecerá la representación visual del trabajo realizado por el usuario y por el sistema, el estado de la tarea en desarrollo, el estado del sistema, la representación de los objetos de interés, etc. El usuario sólo podrá interactuar físicamente con esta zona, de manera efectiva, cuando el *paso* actual del *método* mostrado en la VO así lo indique y permita.

3.4.3. La Jerarquía de Objetivos Activos (JOA)

Esta pequeña zona de la interfaz (situada generalmente en la parte superior de la misma) tiene como misión mostrar el *mapa de objetivos*, es decir, la *jerarquía de objetivos activos* que el usuario mantiene iniciados desde su primer *objetivo* hasta llegar al actual, de manera similar a como actúan las “migas de pan” en una interfaz Web.

Sin embargo, aunque con el resto de usuarios no se detectó ningún problema, en las pruebas realizadas con *novice elderly users* sí que se apreció que el mapa de objetivos, o resultaba “invisible” para ellos, o les generaba confusión, por lo que se optó por eliminarla de la interfaz caso de ir dirigida a estos usuarios (como se verá en el capítulo 5).

3.5. Metodología y notación para la especificación de *Interfaces*

IGO

Como ya se anticipó en apartados anteriores, un aspecto fundamental para el desarrollo de estas interfaces es el modelado o especificación de las mismas, y en especial, de su mecanismo de guía. También se ha comentado que dicha guía se implementa en la VO, que

es la encargada de proporcionar al usuario el conocimiento necesario para interactuar con el sistema, y que lo hace en base a *objetivos, acciones, métodos y selecciones*, de forma parecida a como lo hace un modelo GOMS, pero integrándolo en la propia interfaz IGO.

Hay varias versiones derivadas del GOMS original [18]. En particular, Kieras [72, 73] contribuyó, con esta familia de técnicas, con la idea de un lenguaje natural estructurado, que llamó NGOMSL (Natural GOMS Language). Estas metodologías se enmarcan dentro de un conjunto más amplio de técnicas que permiten realizar un *análisis jerárquico de tareas*. El objetivo principal de estas técnicas es la descomposición de las tareas complejas en subtareas, de modo que el método de resolución se pueda expresar paso a paso. Por ejemplo, HTA [2], TAG [127], TKS [69] o la más reciente, CTT [126, 103] que también incluye la posibilidad de expresar la concurrencia de las tareas. Ninguna de ellas está realmente orientada a su lectura por parte del usuario, sino más bien a la formalización y modelado, por lo que no resultan del todo apropiadas para nuestro caso. En contraste, NGOMSL presenta una más amplia y detallada descripción, así como métodos prácticos de construcción [72, 73] que podremos usar en gran medida. Pero, sin duda, la característica más interesante para la IGO es su cercanía al lenguaje natural del usuario, el cual podrá leer y seguir un modelo de este tipo (así como su “reflejo”, la forma en que se plasma, en la VO de la interfaz) con la facilidad de una receta de cocina. Además, esta característica también facilita la conversión automática, de un modelo ya realizado, a un prototipo inicial de la interfaz [25].

Sin embargo, como veremos a continuación, NGOMSL no resulta del todo adecuada o no permite especificar ciertos aspectos de interés en el modelado de una *interfaz IGO*. Para superar este inconveniente ha sido necesario ampliar y adaptar NGOMSL [19, 24] con ciertos elementos que se irán describiendo en los próximos párrafos. A continuación, presentaremos cada componente que puede aparecer en la especificación de una interfaz de este tipo, y detallaremos la notación concreta a utilizar para realizar dicha especificación, empleando una sintaxis algo menos estricta que la de un lenguaje de programación formal. Para expresar dicha sintaxis haremos uso de elementos típicos de la notación BNF [106].

Lógicamente, y al igual que ocurre al emplear cualquier metodología de análisis de tareas, generalmente la fase de especificación deberá estar precedida por otra fase de estudio y análisis del problema, y de la forma y procesos que sigue el usuario para resolverlo, prestando especial atención al orden de realización de los mismos. Siempre que

sea posible, habrá que analizar: cómo los usuarios conciben las tareas (en términos de sus objetivos naturales), cómo descomponen las tareas en subtareas, qué pasos naturales hay en los métodos del usuario, ordenación o relación temporal de esas tareas, objetivos y pasos y, por último, determinar qué se puede mantener y reflejar directamente en el modelo y en la aplicación a desarrollar, y qué cambios son necesarios realizar.

3.5.1. Objetivos

Un *objetivo*, al igual que en NGOMSL, será cualquier propósito o intención que el usuario pueda querer alcanzar con el sistema. Generalmente, la denominación de un *objetivo* tomará la forma abreviada acción-objeto como, por ejemplo, “crear formulario”, “enviar mensaje”, etc, aunque se podrá detallar cuanto sea necesario, tanto para distinguir unos *objetivos* de otros, como para una correcta comprensión por parte del usuario. El analista deberá identificar todos los posibles *objetivos*, generando el árbol jerárquico de objetivos correspondiente, ya que, generalmente, para alcanzar un *objetivo* (comenzando por el más general o inicial) será necesario, satisfacer uno o varios *objetivos* más concretos (*subobjetivos*). Además, cada *objetivo* siempre deberá tener asociada una descripción que especifique cómo alcanzar dicho *objetivo*, lo que se podrá conseguir, tal y como se detallará en breve, siguiendo los pasos de un *método* (sección 3.5.3) o realizando una *selección* (sección 3.5.4).

3.5.2. Acciones

Una *acción* (denominada *operador* en NGOMSL) será cualquier actividad o tarea a realizar por el usuario, cuya realización podrá afectar también al sistema o sólo a su estado mental, y que ya no será especificada con más detalle. Esto ocurrirá bien porque se trate de una *acción primitiva* (cognitiva, perceptiva, o motora) que ya no se puede descomponer más, o bien porque ya no se considere necesario hacerlo por estimar que el usuario tendrá los conocimientos suficientes para llevarla a cabo por si solo y que sabrá interpretarla (*acción de alto nivel*).

Ejemplos de acciones elementales son: “pulsar una tecla”, “localizar un elemento en la pantalla”, “hacer doble click con el ratón” sobre un objeto, “introducir una tarjeta en el cajero”, etc., mientras que se podrían considerar acciones de alto nivel el “introducir la clave”, “desplazar un objeto por la pantalla”, “elegir un elemento de una lista”, etc., acciones que, aunque se pueden especificar con mayor nivel de detalle, el analista puede

considerar que no es necesario hacerlo, siempre en función del tipo de usuarios al que esté destinada la aplicación.

Si en la posterior fase de testeo o prueba de la interfaz se detectan problemas con alguna de estas *acciones de alto nivel*, el analista deberá pasar a considerarlas como otro *subobjetivo*, y continuar detallándolas cuanto sea necesario.

3.5.3. Métodos

Como ya se ha indicado, cada *objetivo* deberá tener asociado, de manera directa, un *método* o una *selección*. Un *método* especificará la secuencia de *pasos* que debe realizar el usuario para alcanzar el *objetivo* correspondiente.

Cada uno de estos *pasos* expresará, o la necesidad de iniciar un nuevo *subobjetivo* (que de nuevo deberá tener asociado otro *método* u otra *selección*), o de realizar una *acción* concreta (primitiva o de alto nivel). Todos los pasos deberán llevarse a cabo de manera secuencial, según el orden en que estén escritos, si bien, puede haber pasos que expresen un *salto*, *condicional* o *incondicional*, a cualquier otro de los que componen ese método, generalmente para expresar una repetición o para dar la opción de corregir errores, dentro del mismo método.

También es conveniente que el número de *pasos* que integran un *método* no sea excesivo. En caso contrario, habría que contemplar la opción de agrupar *pasos* bajo nuevos *subobjetivos*.

Para la especificación de los *métodos* recomendamos seguir el siguiente formato:

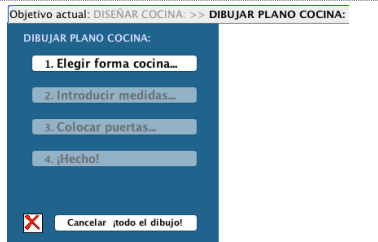
```
Method for: <goal>
  [cancellable
    [disable if <condition_for_system>]
    [effect <effect_on_system>]]

  1. <step1>
  2. <step2>
  ...
  n. Return with goal accomplished
    [effect <effect_on_system>]
```

Como se puede observar, la especificación comienza asociando el *método* al *objetivo* que permite alcanzar. La cláusula **cancelable** es opcional y no existe en NGMOSL. Su inclusión expresaría que el usuario debe poder elegir entre realizar el *paso* actual o cancelar el *objetivo* al que se asocia ese *método*. Tal y como se indicó en el apartado 3.4.1.2, esto último expresaría que se debe detener la ejecución de ese *método* y retornar al *objetivo* padre (el inmediatamente superior en la jerarquía de objetivos activa) y al mismo estado del sistema y al mismo punto que dirigió el flujo de la interacción hacia el *método* que se acaba de cancelar. El significado de esta cláusula quedará modificado si se le añaden las cláusulas **disable_if** o **effect**. Según se detallará en breve, la primera permitirá indicar cuándo se debe deshabilitar la posibilidad de cancelar el *método*, y la segunda, el efecto que la cancelación, si se consume, debe producir sobre el estado del sistema, si se desea especificar.

A diferencia de lo que ocurre en NGMOSL, menos estricta en este aspecto, todos los métodos deberán terminar con un último paso (**Return with goal accomplished**, o simplemente **Return**) que indicará que el objetivo ya ha sido alcanzado y que se debe volver al *método* inmediatamente superior en la jerarquía de objetivos activa, pero al *paso* siguiente al que dirigió el flujo de la ejecución hacia el *método* que acaba de terminar.

Ejemplo M.1:

<pre>Method for: Dibujar plano cocina cancelable 1. Accomplish: Elegir forma cocina... 2. Accomplish: Introducir medidas... 3. Accomplish: Colocar puertas y ventanas... 4. Return</pre>	
--	---

El resto de pasos (desde el 1 hasta el $n-1$) podrán ser de cada uno de los cuatro tipos siguientes (pudiendo ir acompañado, cada uno de ellos, por una o por dos cláusulas que no existen en NGMOSL, **disable_if** y/o **effect**):

```

<stepi> ::= <Iniciar_objetivo> |
             <Realizar_acción> |
             <Salto_condicional> |
             <Salto_incondicional>
             [disable_if <condition_for_system>]
             [effect <effect_on_system>]

```

A continuación, concretaremos la especificación, significado y traslación a la VO de la interfaz, de cada uno de esos 4 tipos de pasos.

```

<Iniciar_objetivo> ::= Accomplish: <goal>

```

Indica que el usuario debe iniciar un nuevo *subobjetivo* (goal) e implicará la ejecución del *método* o de la *selección* que permite alcanzarlo.

Ejemplo M.2:

Method for: Poner mueble
cancelable

1. **Accomplish:** Elegir tipo de mueble...
2. Pinche un mueble
3. Pinche su lugar
4. **Return**
 effect num_muebles++

Objetivo actual: DISEÑAR COCINA: >> AMUEBLAR COCINA: >> MUE

PONER mueble:

1. Elegir tipo de mueble...

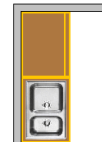
2. Pinche un mueble

3. Pinche su lugar

4. ¡Hecho!



Cancelar



Ejemplo M.3:

Method for: Poner mueble
cancelable

1. **Accomplish:** Elegir tipo de mueble...
2. Pinche un mueble
3. Pinche su lugar
4. **Return**
 effect num_muebles++

Objetivo actual: DISEÑAR COCINA: >> AMUEBLAR COCINA: >> MUEBLES Y ELE

PONER mueble:

1. Elegir tipo de mueble...

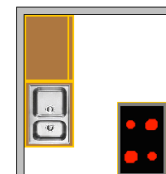
2. Pinche un mueble

3. Pinche su lugar

4. ¡Hecho!



Cancelar



```
<Salto_condicional> ::= Decide, if <condition_for_user>
                        then goto <stepj>
```

Implica la realización de una *acción* de carácter mental, según la cual, el usuario debe decidir (en función de la condición a evaluar, *condition_for_user*) si saltar al paso indicado (*step_j*) o continuar por el paso siguiente.

Permite pues expresar un salto condicional a otro *paso* del mismo *método*, permitiendo crear bucles condicionales para la repetición de *pasos* o la corrección de errores.

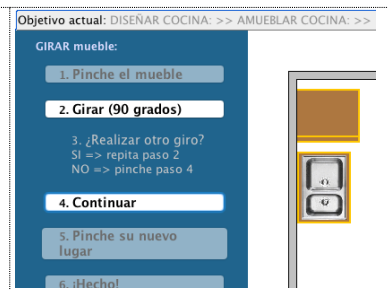
A diferencia de lo que ocurre en NGOMSL, la cláusula **then** sólo podrá implicar la realización de un *salto incondicional* (**goto**), no pudiendo ir seguida de ningún otro tipo de operación. Esta limitación simplifica la especificación, su implementación sobre la VO, y el mecanismo de interacción.

Ejemplo M.4:

Method for: Girar mueble

cancelable

1. Pulse sobre el mueble que desea girar
2. Girar 90 grados
3. **Decide, if** ¿desea realizar otro giro?
then goto step 2
4. Continuar
5. Pinche nuevo lugar del mueble
6. **Return**



```
<Salto_incondicional> ::= Goto <stepj>
```

Expresa un *salto incondicional* a otro *paso* (*step_j*) dentro del mismo *método* en el que aparece. Este tipo de *paso* nunca se ofrecerá directamente al usuario en la VO, sino el *paso* al que deriva (*step_j*). Se podría decir que tiene un efecto transparente para el usuario. Se debe usar con cautela.

Además, y de manera opcional, se podrá añadir, a cada *paso* que lo requiera, la cláusula **disable_if** y/o la cláusula **effect**.

La cláusula **disable_if** daría lugar a lo que denominaremos *paso condicional*. Dicha cláusula requiere de una condición (*condition_for_system*) que, a diferencia de las que incluyen los *saltos condicionales*, no deberá evaluarla el usuario, sino el sistema, justo antes de habilitar y ofrecer el paso al usuario, en la VO. Si el estado interno del sistema, representado por las variables que lo definen, hace que la condición sea evaluada como verdadera, entonces el usuario no deberá tener acceso a este *paso*. Por tanto, en la VO, o no se mostrará, o se mostrará deshabilitado (opción usada para los tests del Capítulo 5), pasando a ofrecerse el *paso* siguiente.

Ejemplo M.5:

Method for: Elegir materiales comunes...

cancelable

1. Pinche la encimera que prefiera
disable_if (es_posible_encimera==false)
2. Pinche la madera que prefiera
disable_if (num_puertas==0)
3. Pinche el tirador que prefiera
disable_if (num_puertas==0)
4. **Return**

Aclaración del ejemplo: En este caso, el mobiliario colocado en la cocina es sólo un horno-vitrocerámica, y una estantería (que no lleva puertas). El primero no requiere ninguno de los tres materiales comunes contemplados (encimera, madera y tiradores). En cambio, aunque la estantería sí necesitará encimera, tampoco llevará ni madera ni tiradores. Por tanto, tras realizar el paso 1, no se deberá ofrecer la posibilidad de elegir estos dos materiales comunes: los pasos 2 y 3 deberán aparecer deshabilitados (o no aparecer).

La cláusula **effect** permite describir los efectos de interés (*effect_on_system*) que la realización del *paso* que la incluye debe producir en el estado del sistema. Dicho de otra forma, permite especificar (con una notación libre) qué factores o variables del estado del sistema se verán afectadas y de qué manera, una vez realizado dicho *paso*. En general, estas variables serán las que se utilicen para componer las condiciones de las cláusulas **disable_if** de los *pasos* condicionales.

Ejemplo M.6:

Method for: Poner mueble

cancelable

1. **Accomplish:** Elegir tipo de mueble...
2. Pinche el mueble deseado
3. Pinche el lugar donde colocarlo
4. **Return**
effect num_muebles++

Aunque con el uso de estas dos cláusulas estamos entrando en aspectos externos al usuario, y que generalmente suelen abordarse en la etapa de programación de la interfaz, su uso, además de permitir destacar ciertos detalles de interés del proceso interactivo, facilitar la creación automática de un prototipo inicial de la VO, a partir de la especificación realizada [20].

3.5.4. Selecciones

Las *selecciones* tienen aquí un papel parecido al de las *Reglas de selección* en NGOMSL, y consistirán en un grupo de *alternativas (opciones)*. Se deberá definir una *selección* siempre que se identifiquen o existan varias formas de alcanzar un mismo *objetivo*. Generalmente, esto ocurrirá cuando alcanzar un *objetivo* más general (p.ej: “realizar una operación bancaria”) consiste en alcanzar uno de varios *objetivos* más concretos (como “consultar saldo”, “sacar dinero”, o “ingresar dinero”).

En caso de que una *selección* estuviese compuesta por demasiadas *alternativas* habría que considerar la agrupación de *alternativas* bajo nuevos *subobjetivos*.

Una *selección* tendrá el siguiente formato:

```
[For the system,] Selection for: <goal>
    [cancellable [disable if <condition_for_system>]
    [effect <effect_on_system>]]
    a) <option_a>
    b) <option_b>
    ...
    Return with goal accomplished
```

La cláusula **cancelable** sólo tendrá sentido si se trata de una *selección* “para el usuario” (y no “para el sistema”, que es un tipo de *selección* que se verá a continuación). Tendrá el mismo cometido que en un *método*: permitir cancelar esa *selección* y retornar al *objetivo* padre (el inmediatamente superior en la jerarquía activa). Se puede considerar como otra *alternativa* más de esa *selección*. También puede estar acompañada de las cláusulas **disable_if** y **effect**.

Por otro lado, cada $option_i$ podrá especificarse siguiendo el siguiente formato:

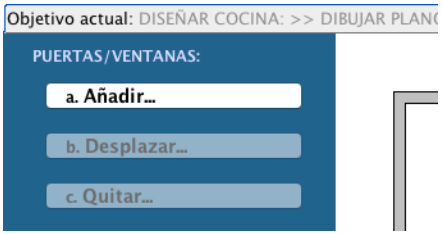
```
<option_i> ::= If <condition_i> [then accomplish <goal_i>]
    [disable if <condition_for_system>]
    [effect <effect on system>]
```

Cada una de las distintas *alternativas* ($option_i$) que componen una *selección* deberá incluir una condición ($condition_i$) que deberá evaluar el usuario (o el propio sistema, según se concretará en breve). Todas las condiciones de las distintas *alternativas* deberán

ser mutuamente excluyentes. Además, siempre que sea posible y ventajoso, conviene ofrecer dichas *opciones* siguiendo algún criterio de ordenación adecuado al caso, como su relevancia, su frecuencia, orden alfabético, etc.

Existen dos tipos de *alternativas*, las que implican iniciar nuevos *subobjetivos*, y las que no. En el primer caso, la *alternativa* incluirá la parte **then accomplish** *goal_i*, e implicará la iniciación de ese nuevo *subobjetivo*, que de nuevo podrá ser alcanzado siguiendo un *método* o realizando otra *selección*. Posteriormente, una vez conseguido dicho *objetivo*, ya no se volverá a esta *selección* (ni a ninguna otra), sino al *método* correspondiente al *objetivo* previo más cercano en la jerarquía de objetivos activa en ese momento.

Ejemplo S.1:

<pre> Selection for: Elegir tarea con puertas Cancelable disable_if (num_puertas == 0) a) If desea poner una puerta then accomplish: Añadir... b) If desea quitar una puerta then accomplish: Quitar... disable_if (num_puertas == 0) c) If desea desplazar una puerta then accomplish: Desplazar... disable_if (num_puertas == 0) Return </pre>	
---	--

El segundo caso será menos habitual, y la elección de estas *alternativas* (por ejemplo, los distintos tipos de tarjetas de crédito que puede elegir el cliente de un banco; o los distintos tipos de muebles disponibles, si se está realizando el diseño de una cocina) sólo provocaría un efecto o un cambio en el estado del sistema, en las variables que lo definen (por ejemplo, el tipo de tarjeta o de electrodoméstico escogido), o activarían un comando del sistema. En este caso, dichas *alternativas* no incluirán la parte **then accomplish**, pero sí podrían incluir, si se quiere especificar el efecto producido, la cláusula **effect**.

Ejemplo S.2:

Selection for: Elegir tipo de mueble a poner
cancelable

a) **If** muebles bajos
effect (tipo_mueble = bajo)

b) **If** muebles altos (de pared)
effect (tipo_mueble = alto)

c) **If** electrodomésticos
effect (tipo_mueble = electrod.)

d) **If** otro mobiliario
effect (tipo_mueble = otro)

Return

Aclaración del ejemplo: En este caso, ninguna de las *alternativas* contempladas implica iniciar un nuevo sub-*objetivo*, sino dar un comando al sistema, concretando el tipo de muebles con los que quiere trabajar el usuario.
Además, al elegir una de ellas, se retornará (en la VO) al *método* asociado al *objetivo* más cercano en la jerarquía de objetivos activos.

Como se ha visto en los ejemplos anteriores, al igual que ocurría con los *pasos* de un *método*, también se podrán expresar *alternativas condicionales* (inhabilitadas o no presentes, caso de que el sistema verifique que se cumple la condición de la cláusula **disable_if**), y/o *alternativas con semántica asociada*, que producirán un efecto concreto sobre el sistema caso de ser seleccionadas, para lo cual, deberán incluir la cláusula **effect**.

Generalmente las *selecciones* están destinadas o encaminadas al usuario, que será el encargado de evaluar o realizar la *selección*. Sin embargo, durante el proceso de interacción también puede surgir la necesidad de que la VO ofrezca al usuario un *objetivo* concreto de varios posibles, pero en función del estado del sistema, y no de una elección a realizar por el usuario. En estos casos diremos que se trata de *selecciones para el sistema*, que será el encargado de evaluar las condiciones, excluyentes entre sí, y en función de la que sea cierta, desviar el flujo hacia el *objetivo* correspondiente. Para indicar que una *selección* es de este tipo se deberá comenzar su especificación con la cláusula **For the system**. Estas *selecciones* no se mostrarán nunca en la VO, sino el *objetivo* al que deriven.

Ejemplo S.3:

<pre> For the system, Selection for: Introducir medidas a) If (forma == cuadrada) then accomplish: Introducir medidas paredes (forma cuadrada)... b) If (forma == rectangular) then accomplish: Introducir medidas paredes (forma rectangular)... c) If (forma == otra) then accomplish: Introducir medidas paredes (forma libre)... Return </pre> <p>-----</p> <p>Aclaración del ejemplo: En este caso, cuando (en la VO de la 1ª imagen) el usuario inicie el <i>objetivo</i> asociado al paso 2 (“Introducir medidas” de las paredes de la cocina cuya forma se habrá elegido en el paso 1), se activará la <i>selección para el sistema</i> del ejemplo, y ésta derivará al <i>método</i> concreto (mostrado ya en la VO de la 2ª imagen) que permite introducir las medidas de las paredes para la forma de cocina elegida.</p>	
---	--

En el **anexo 1** se incluye la especificación completa de la *interfaz IGO* utilizada como ejemplo en el proceso de evaluación empírica realizado en el siguiente capítulo.

3.6. Ejemplo de especificación NGOMSL-IGO

En esta sección se presenta y explica parte de la especificación o modelo correspondiente a una aplicación que permitiría realizar, usando precisamente esta notación, la especificación de *interfaces IGO*. Para clarificar la especificación, las palabras y cláusulas correspondientes a la notación empleada aparecerán en **negrita** y en **minúscula**, los nombres de los objetivos y acciones que debe iniciar o realizar el usuario irán en **mayúsculas**, y el resto del texto (que se puede utilizar para mostrárselo también al usuario como *descripción emergente* o *texto alternativo*) sólo en **minúscula**.

```

Method for INICIO
1) Accomplish: ELEGIR UN PROYECTO
2) Accomplish: TRABAJAR EN EL PROYECTO
    disable if (hay proyecto cargado == false)
3) Accomplish: CERRAR EL PROYECTO
    disable if (hay proyecto cargado == false)
4) Decide: if desea trabajar en otro proyecto
    then goto 1
5) return

```

En el método anterior tenemos dos ejemplos de *pasos condicionales* (el 2º y 3º). En el momento de la ejecución, dichos pasos deberán aparecer como activos o ejecutables sólo si el sistema comprueba que “hay cargado algún proyecto”, siendo además activado este estado del sistema mediante cláusulas *effect* incluidas en la descripción de un paso (el 1º) que se habrá ejecutado previamente, y cuya especificación es la siguiente:

```

Selection for: ELEGIR UN PROYECTO
  cancelable effect (hay proyecto cargado = false)
  a) If desea crear un nuevo proyecto then
      effect (hay proyecto cargado = true)
  b) If desea abrir un proyecto ya existente then
      accomplish: ABRIR PROYECTO EXISTENTE
  Return with goal accomplished

```

En esta selección se pueden destacar varios aspectos. En primer lugar, que se trata de una *selección cancelable* y, además, con efecto o *semántica asociada*, ya que si el usuario decide cancelarla, entonces habrá que reflejar que el sistema se encontrará en un estado en el que “no hay ningún proyecto cargado”.

En segundo lugar, también se puede apreciar que la primera alternativa no es la típica que implica la iniciación de una nueva subtarea, sino que tan sólo producirá un efecto en el sistema: establecer que “ya hay un proyecto cargado”. La segunda opción sí que daría inicio a un nuevo subobjetivo: “ABRIR PROYECTO EXISTENTE”.

```

Method for: TRABAJAR CON EL PROYECTO
  cancelable
  1) Accomplish: ELEGIR TRABAJO A REALIZAR
  2) Decide: if desea seguir trabajando en el
                proyecto actual, then goto 1
  3) Return

```

El método anterior muestra cómo se puede expresar una *repetición* dentro de un método.

```

Selection for: ELEGIR TRABAJO A REALIZAR
  cancelable
  a) If desea realizar una especificación a través
      del grafo correspondiente al proyecto then
      accomplish: ESPECIFICAR EN GRAFO
  b) If desea realizar especificación a través de
      la “ventana conductora de objetivos” then
      accomplish: ESPECIFICAR EN VCO
  c) If desea simular la interacción then
      accomplish: SIMULAR

```

```

d) If desea imprimir el proyecto actual then
    accomplish: IMPRIMIR PROYECTO
e) If desea almacenar el proyecto actual then
    accomplish: GRABAR PROYECTO
Return

```

```

Selection for: ESPECIFICAR EN GRAFO
cancelable
a) If desea trabajar con un nodo ya existente then
    accomplish TRABAJAR CON NODO EXISTENTE
    disable if (existe algún nodo == false)
b) If desea crear un nuevo nodo then
    accomplish CREAR UN NUEVO NODO
Return

```

Esta última selección presenta una *opción condicional* (la 1ª), por lo que podrá aparecer inhabilitada (o no aparecer) en el momento de la ejecución, ya que “si no existe ningún nodo”, el usuario aún no podrá iniciar el objetivo de “trabajar con nodos existentes”.

```

Method for: TRABAJAR CON NODO EXISTENTE cancelable
1) Accomplish SELECCIONAR UN NODO
2) Accomplish ELEGIR TRABAJO A REALIZAR CON NODO
    disable if (hay nodo seleccionado == false)
3) Decide: if desea seguir trabajando con el
           mismo nodo, then goto 2
    disable if (hay nodo seleccionado == false)
4) Decide: if desea seguir trabajando con nodos
           existentes, then goto 1
    disable if (hay nodo seleccionado == false)
5) Return

```

En el método anterior aparecen *pasos condicionales* (el 2º y el 3º), de nuevo dependientes de un estado del sistema, que será establecido durante la realización de una tarea previa: la de “seleccionar un nodo”.

En cambio, el método correspondiente a dicha tarea está formado sólo pasos consistentes en *acciones elementales*, una de ellas, la última, *con semántica asociada*, ya que al completar el método será el momento de establecer en el estado del sistema que “ya hay un nodo seleccionado”. En este método se muestra también cómo se puede abordar el problema de la *corrección de errores* por parte del usuario (paso 3º):

```

Method for: SELECCIONAR UN NODO cancelable
  1) LOCALIZA EL NODO DESEADO EN EL GRAFO
  2) COLOCA EL RATÓN SOBRE EL NODO DESEADO
    y PULSA EL BOTON IZQUIERDO DEL RATÓN
  3) Decide: if el nodo seleccionado no es el
    deseado then goto 1
  4) Return
    effect (hay nodo seleccionado = true)

```

```

Selection for: ELEGIR TRABAJO CON NODO cancelable
  a) If desea trabajar en la especificación
    correspondiente al nodo actual then
    accomplish ESPECIFICAR NODO
  b) If desea eliminar el nodo actual then
    accomplish ELIMINAR NODO
  c) If desea eliminar el nodo actual y todos sus
    nodos hijos then
    accomplish ELIMINAR JERARQUIA DE NODOS
  d) If desea cambiar tipo del nodo actual then
    accomplish CAMBIAR TIPO DEL NODO
  e) If desea crear un nuevo nodo then
    accomplish CREAR UN NUEVO NODO
Return

```

Si el usuario inicia el objetivo de “trabajar en la especificación correspondiente al nodo actual”, será el sistema el encargado de concretar y mostrarle el método adecuado que debe seguir (“especificar un método”, “una selección”, o “una acción”) en función del tipo que sea el nodo actual. Evidentemente, esta comprobación no tiene sentido que la realice el usuario. Sirve pues como ejemplo de *selección para el sistema*:

```

For the system, Selection for: ESPECIFICAR NODO
  a) If (tipo nodo actual = método) then
    accomplish ESPECIFICAR METODO
  b) If (tipo nodo actual = selección) then
    accomplish ESPECIFICAR REGLA
  c) If (tipo nodo actual = acción) then
    accomplish ESPECIFICAR OPERADOR
Return

```

```

Method for: ESPECIFICAR METODO cancelable
  1) Accomplish ELEGIR OPERACION EN METODO
  2) Decide: if desea realizar otra operación en
    este método then goto 1
  3) Return

```

```
Selection for: ELEGIR OPERACION EN METODO
cancelable
a) If desea añadir un nuevo paso al final del
    método actual then
    accomplish AÑADIR PASO AL FINAL
b) If desea eliminar un paso del método then
    accomplish ELIMINAR PASO
c) If desea modificar un paso del método then
    accomplish MODIFICAR PASO
d) If desea insertar un paso intermedio then
    accomplish INSERTAR PASO
e) If desea cambiar el nombre al método then
    accomplish CAMBIAR NOMBRE METODO
f) If desea gestionar la cláusula "cancelable" de
    este método then
    accomplish GESTIONAR "CANCELABLE"
Return
```

Aunque se podría continuar y modelar alguna otra situación, consideramos que la parte presentada es suficiente para reflejar la filosofía de la especificación IGO, al mismo tiempo que recoge los aspectos fundamentales y todos los elementos de la notación propuesta.

3.7. Resumen de las aportaciones de la notación NGOMSL-IGO

Vamos a concluir destacando las principales aportaciones de la metodología y de notación presentada. Como ya se ha indicado anteriormente, se ha elegido NGOMSL como notación base o de referencia, tanto por adaptarse muy bien a la descomposición jerárquica de objetivos y tareas, como por su fácil lectura y cercanía al lenguaje natural del usuario. Sin embargo, NGOMSL resulta insuficiente para modelar ciertas situaciones o aspectos del proceso de *Interacción Guiada por Objetivos* que conviene poder considerar y reflejar a la hora de realizar la especificación de este tipo de interfaces, especialmente, si se desea facilitar la generación automática de las mismas, y especialmente, de la parte que hace de guía, la VO. Por ello, ha sido necesario adaptarla y extenderla con los siguientes elementos:

- **Pasos y Alternativas condicionales:** son los que dependerán del estado del sistema para aparecer o no habilitados dentro del *método* o *selección* presentado al usuario para satisfacer un *objetivo* concreto.
- **Pasos y Alternativas con efecto:** su realización o elección debe producir un efecto concreto (que interesa destacar) sobre el estado del sistema, y que

generalmente será consultado por *Pasos o Alternativas condicionales* o por *Selecciones para el sistema*.

- **Objetivos cancelables:** son los que ofrecen la posibilidad de cancelar el *método* (o la *selección*) asociado a ese *objetivo*, y retornar al mismo estado y punto en que se encontraba el sistema justo antes de iniciarse el *objetivo* cancelado.
- Y, por último, además de las típicas *selecciones* dirigidas al usuario, también se pueden definir *Selecciones para el sistema* que, en función de su estado, sería el encargado de elegir la *alternativa* adecuada, de manera transparente para el usuario.

3.8. Limitaciones de la IGO

Evidentemente, como ocurre con el resto de estilos de interacción, la IGO también tiene ciertas limitaciones y casos para los que no sería acertada (o incluso ni factible) su aplicación.

Una limitación, aunque autoimpuesta a conciencia, con objeto de reducir la complejidad cognitiva del usuario y simplificar el proceso de interacción, es su naturaleza secuencial. De todas formas, si se estima conveniente, se podría considerar la posibilidad de “flexibilizar” la interacción y definir *métodos* con grupos de *pasos* (que puedan estar activos simultáneamente) cuya secuencia pudiese ser elegida libremente por el usuario, que sería el encargado de decidir el orden en que desea llevarlos a cabo.

Siguiendo con las limitaciones, y siempre dentro del contexto del uso ocasional, consideramos que existen, al menos, tres casos para los que la IGO podría no ser adecuada:

- Para aplicaciones de **alto carácter creativo**, en las que es la propia interacción la que puede inspirar o influir notablemente en el resultado final, como ocurriría con un programa de dibujo.
- Para aplicaciones, sean o no de tipo creativo, en las que se deben ofrecer **muchos comandos o funciones** continua y simultáneamente, como podrían ser potentes hojas de cálculo, procesadores de texto, o aplicaciones de edición de video con multitud de opciones.

- En sistemas que requieran un **alto grado de descomposición de los objetivos**, suponemos que el continuo “guiar y conducir al usuario” por la jerarquía de objetivos iniciados, junto con la secuencialidad que implica la IGO, traería consigo, lo mismo que suele ocurrir con los menús muy profundos y estrechos: una disminución, tanto de la *eficiencia* (que podría convertir a esta forma de interacción en tediosa, lenta y “pesada”), como de la *eficacia* (al aumentar el riesgo de que el usuario se pierda por dicha jerarquía de objetivos tan profunda). Sin embargo, hay que tener en cuenta que este tipo de aplicaciones no son las que suelen destinarse al uso ocasional, sino más bien al contrario, a usuarios frecuentes y con gran “expertise” sobre dichos sistemas y sus interfaces.

En cambio, la especificidad de los objetivos en tareas menos creativas es de suma importancia cuando se trata de guiar la interacción. Reservar una habitación de hotel, elegir un asiento para una función de teatro o aplicar transformaciones específicas a un lote de archivos, son algunos ejemplos para los que IGO sí podría ser una buena alternativa.

En el siguiente capítulo se pondrá a prueba la IGO mediante un estudio empírico y comparativo entre dicha forma de interacción y las tan extendidas *interfaces de Manipulación Directa*, cuyos resultados respaldan el empleo de este tipo de interacción guiada para el uso ocasional. Posteriormente, en el Capítulo 5, se propondrá (y evaluará) una variante de este tipo de interfaces, pero especialmente enfocada a *personas mayores con poca experiencia con la tecnología (novice elderly users)*.

Capítulo 4. Interacción Guiada por Objetivos vs Manipulación Directa: un estudio empírico

En el capítulo anterior se presentaron los detalles de la *Interacción Guiada por Objetivos*. En este capítulo se evalúa esta forma de interacción mediante un estudio empírico, tanto cuantitativo como cualitativo, con usuarios reales, obteniéndose resultados que confirman lo adecuado de dicha propuesta.

4.1. Introducción

Una vez presentada la *Interacción Guiada por Objetivos* (IGO, o GDI en inglés), el siguiente paso es probar su adecuación con usuarios reales. Para ello, hemos realizado un estudio comparativo [19, 20] entre dicha interacción guiada y el tradicional estilo de *Manipulación Directa* (MD).

Este análisis empírico nos proporcionará datos y resultados, tanto objetivos como subjetivos. En concreto, queremos probar dos hipótesis de carácter objetivo, relacionadas con el *tiempo de realización de las tareas* y el número de *errores* cometidos, y una tercera hipótesis de carácter subjetivo, relacionada con la *satisfacción* del usuario.

Los *Usuarios Ocasionales*, no expertos en informática:

- **(H1)** operan o trabajan más rápido con una IGO que con la MD estándar,
- **(H2)** cometen menos errores con una IGO que con la MD estándar,
- **(H3)** prefieren la IGO a la típica MD.

Como ya se ha indicado y se detallará en el siguiente apartado, dichas hipótesis no consideran participantes con altos conocimientos informáticos ni tecnológicos.

Por otro lado, con esas hipótesis se tratan los tres pilares básicos de la usabilidad (ISO 9241-11) [59]: la *efectividad* (**H2**), la *eficiencia* (**H1**) y la *satisfacción* (**H3**).

4.2. Participantes

El estudio involucró a veinte participantes voluntarios ($n = 20$), un número que cumple con los criterios [82, 119, 160, 7] para este tipo de estudios. Participaron 12 mujeres y 8 hombres. Sus edades oscilaban entre los 10 y los 52 años, con una edad media de 44 años. Todos los sujetos desconocían el objetivo final de la investigación. Ninguno había participado en estudios de usabilidad anteriores ni recibió ningún incentivo para participar en el experimento. Siguiendo una estrategia *contrabalanceada* (que se detallará más adelante), cada participante tuvo que usar dos versiones diferentes de la misma aplicación, cada una con un tipo de interfaz diferente, acorde al tipo de interacción asociado (IGO o MD).

La Tabla 4.1 resume los principales datos y características de los participantes, recopilados mediante un *cuestionario Pre-test (Background Questionnaire)*.

Para la elección y “criba” de los participantes se exigían dos condiciones fundamentales. El primer requisito consistía que ningún participante tuviese experiencia previa con aplicaciones similares a las evaluadas en este estudio. Sin embargo, y de manera intencionada, se consideraron dos excepciones al incluir dos participantes profesionales en el dominio de la tarea y que, en su trabajo diario, utilizaban un software comercial similar a nuestra versión basada en MD, aunque más completo y potente.

El segundo requisito implicaba descartar a usuarios con alto nivel en informática. Como se refleja en Tabla 4.1: ningún participante puntuó su experiencia en informática (*Computer expertise*) por encima de 3, en una escala de 0 a 5 puntos. Sin embargo, como este parámetro es subjetivo, para poder confirmar dichos valores auto-asignados, hicimos las siguientes preguntas complementarias (Tabla 4.1):

- PUS (*Previously Used Software*), cuyas opciones son: Internet, Email, Word Processor, Spreadsheet, Multimedia, Accounting & management, Databases, social Networks, Others)
- KDSK (*Kitchen Design Software Knowledge*), con respuesta de tipo si/no
- TSE (*Tablets and Smartphones Expertise*), puntuada de 0 (nada) a 4 (experto)
- TDK (*Task Domain Knowledge*), puntuado de 0 (nada) a 4 (experto)

- ASL (*Academic Studies Level*), puntuado de 0 a 3:
0-No studies; 1-Primary school; 2-Secondary school; 3-University
- ORK (*Other Relevant Knowledge*), de respuesta libre

Tabla 4.1 Resumen de los principales datos y características de los participantes, recopilados mediante el Cuestionario Pre-test (Background Questionnaire): Gender (**G**), Age (**A**), Computer Expertise (**CE**), Previously Used Software (**PUS**) [opciones: **I**nternet, **E**mail, **W**ord Processor, **S**pread_sheet, **M**ultimedia, **A**ccounting & management, **D**atabases, social **N**etworks, **O**thers], Kitchen Design Software Knowledge (**KDSK**), Tablets and Smartphones Expertise (**TSE**), Task Domain Knowledge (**TDK**), Academic Studies Level (**ASL**) puntuado de 0 a 3 [0-No studies; 1-Primary school; 2-Secondary school; 3-University], y Other Relevant Knowledge (**ORK**). La Computer Expertise (**CE**) de cada participante se verificó y validó de acuerdo con estos ítems.

#	G	A	CE	PUS	KDSK	TSE	TDK	ASL	ORK
	M/F		[0..5]	[I,E,W,S,M,A,D,N,O]		[0..4]	[0..4]	[0..3]	
1	F	51	0		No	1	3	1	
2	F	48	0		No	1	3	1	
3	F	48	0	I	No	1	3	1	
4	M	12	0	I	No	2	1	0	
5	F	42	1	I,E	No	2	3	1	
6	F	50	1	I,M	No	2	3	2	
7	M	24	1	I,M,O	No	2	2	1	Occasional use of IKEA kitchen design app
8	F	46	2	I,E,W,S,M,A (forgotten)	No	2	3	2	Forgotten computer courses
9	F	44	2	I,E,W,M,N,O	No	2	3	1	
10	F	50	2	I,E,W,M,O	No	2	3	1	Medical software
11	F	40	2	I,E,W,M,O	No	2	3	3	Medical software
12	M	52	2	I,E,W,S,M,O (forgotten)	No	2	3	1	Pantograph. Forgotten computer courses
13	M	43	2	I,E,W,M,A,N,O	No	2	2	1	Photoshop
14	F	35	3	I,E,W,S,M,A,D	No	2	3	2	
15	M	43	3	I,E,W,S,M,A,D,	No	2	2	3	Veterinary software
16	M	50	3	I,E,W,M,O	No	2	3	1	AutoCAD and Presto user
17	F	43	3	I,E,W,S,M,A,D, O	No	1	3	2	Forgotten computer courses
18	M	52	3	I,E,W,S,M,A,D,	No	2	3	2	
19	M	52	3	I,E,W,S,M,A,O	Kitchens	3	4	2	Photoshop. Kitchen design professional
20	F	52	3	I,E,W,S,M,O	Kitchens	2	4	1	Kitchen design professional

También se realizó un estudio complementario con Eye-Tracker a tres de los participantes, cuya justificación y detalles se incluyen en la sección 4.7.4.

4.3. Material empleado y tareas a realizar

En cuanto al material a emplear, lo primero a decidir fue el tipo de aplicación más apropiada para el estudio. Las premisas eran las siguientes. En primer lugar, que fuese una



aplicación factible para un uso ocasional. En segundo lugar, que el dominio de la aplicación resultase familiar a la mayoría de personas, independientemente de su edad, género, nivel de estudios, etc., pero que, al mismo tiempo, permitiese realizar un amplio y variado conjunto de tareas. Y, por último, que fuese un tipo de aplicación que, caso de existir previamente, sólo fuese bajo el clásico estilo de MD, y que, a su vez, incluyese varios objetos interactivos que el usuario debiera manipular. El hecho de que estas características pareciesen jugar a favor del estilo de la MD lo convertía en un reto extra para nuestro estudio.

Por todo lo anterior, se optó por una aplicación que permitiese al usuario realizar el diseño de su propia cocina, elegir el mobiliario a comprar y ubicarlo en dicha cocina, y realizar modificaciones y operaciones complementarias, como mover y eliminar muebles, consultar el presupuesto, ver la cocina en 3D, y encargar o guardar la cocina.

Se desarrollaron pues, dos variantes de la misma aplicación, una con *interfaz IGO* y la otra con la clásica *interfaz de MD*. Ambas versiones, desarrolladas en Java, comparten la mayor parte del código, excepto las partes relativas a la IU. Las dos versiones ofrecen la misma funcionalidad, permiten realizar las mismas tareas, y lograr los mismos objetivos. Sólo cambia la interfaz y los mecanismos de interacción correspondientes, en función del estilo de interacción aplicado. La versión de MD está inspirada en la versión de escritorio ofrecida por la empresa IKEA [57], aunque con menor funcionalidad y con un catálogo de productos muchísimo más reducido.

Las Figura 4.1, Figura 4.2 y Figura 4.3 se corresponden a algunas de las etapas por las que va pasando el usuario. Las capturas de pantalla situadas a la izquierda de cada figura corresponden a la interfaz de MD; las capturas de la derecha, a la interfaz IGO. Con la *interfaz de MD*, el usuario debe comenzar introduciendo la forma y las dimensiones de una cocina dada, y luego amueblarla. Para ello, debe elegir el mobiliario deseado y colocarlo en la posición adecuada de la cocina, todo ello empleando los mecanismos típicos de la MD, es decir, usando “deshacer” y “rehacer”, etiquetas descriptivas para los distintos botones de comandos (títulos), menús contextuales, etc. En cambio, la *interfaz IGO* va presentando al usuario, continuamente, en la VO, las selecciones y/o la lista de pasos a seguir. Cada paso o acción a realizar en el área de trabajo, implicará interacciones de manipulación directa, pero cortas, sencillas y claramente descritas en la VO. El usuario no

podrá continuar hasta que se complete la acción anterior, siempre de acuerdo a la filosofía de esta forma de interacción.

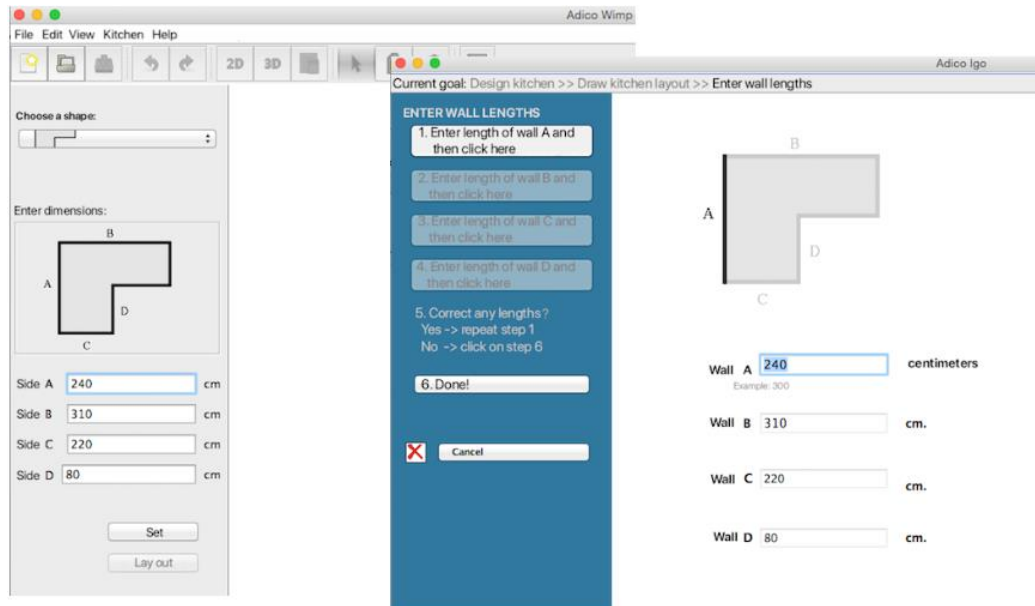


Figura 4.1 Capturas de pantalla de las interfaces testeadas, relacionadas con el proceso de introducción de las medidas de las paredes de la cocina: a la izquierda, la *interfaz de MD*; a la derecha, la *interfaz IGO*.

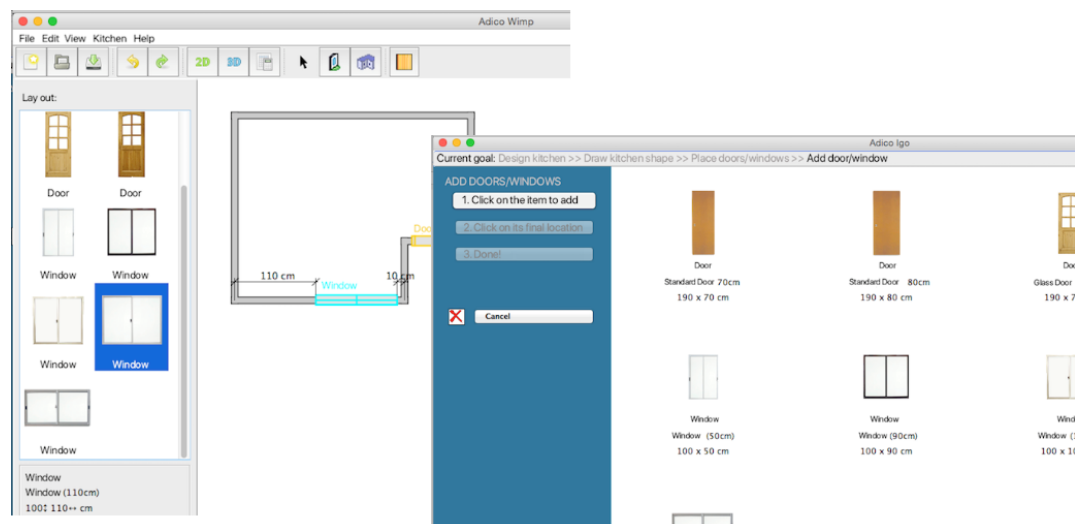


Figura 4.2 Capturas de pantalla de las interfaces testeadas, relacionadas con el proceso de selección de un tipo de ventana y de su colocación sobre la forma o contorno de la cocina: a la izquierda, la *interfaz de MD*; a la derecha, la *interfaz IGO*.

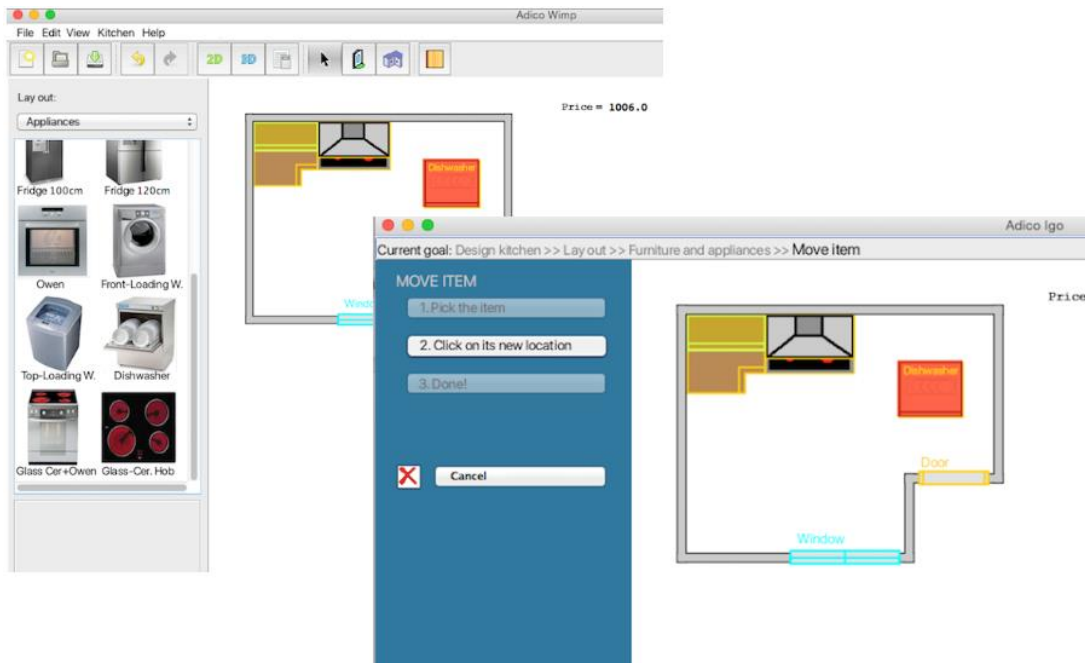


Figura 4.3 Capturas de pantalla de las interfaces testeadas, relacionadas con el proceso de mover o cambiar un mueble de posición: a la izquierda, la *interfaz de MD*; a la derecha, la *interfaz IGO*.

A cada participante se le entregó la misma *hoja de tareas*, con la información que se muestra a continuación en la Figura 4.4. En ella se describe tanto el escenario de uso de la aplicación como los tres grupos de tareas a realizar.

4.4. Diseño del estudio

Para este experimento se utiliza un *diseño Intra-Sujetos (Within-Subjects)* con una variable independiente: el *tipo de interacción*. Dicha variable independiente tiene dos niveles: *interacción IGO*, e *interacción MD*.

Este *diseño Intra-Sujetos*, o *diseño de Medidas Repetidas*, era necesario, ya que queríamos que cada participante experimentara ambos estilos de interacción. Era importante que ambos grupos de tratamiento incluyeran exactamente a los mismos participantes, de manera que pudieran comparar directamente ambos tipos de interfaces y expresar sus preferencias y, además, sin tener que preocuparnos por las características personales y habilidades concretas de cada participante, que podrían sesgar nuestros resultados.

Por otro lado, para mitigar el posible efecto de transferencia de aprendizaje, debido al uso de un tipo de interfaz antes que el otro, se aplicó una *estrategia contrabalanceada*, alternando el orden de uso de las dos interfaces entre los usuarios. Así pues, la mitad de los participantes comenzó con IGO, mientras que la otra mitad comenzó con MD.

ESCENARIOS Y TAREAS A REALIZAR	
<p>Imagine que llega a una tienda de cocinas, decidido a elegir los muebles para la suya, y se encuentra que todos los encargados están ocupados. Sin embargo, descubre que hay una mesa de ordenador y un cartel que le invita a sentarse para, haciendo uso de un programa, elegir usted mismo el mobiliario que desea para su cocina (y que quedaría pendiente de repaso y confirmación con un comercial de la tienda). Con objeto de ganar tiempo, usted decide hacer uso de dicho sistema y realizar un boceto de su cocina deseada.</p>	
<p>Tarea 1: Indique cómo es su cocina</p> <p>Suponga que trae, hecho de casa, un croquis (como el de la figura de la derecha) en el que dibujó la forma de su cocina, anotó las medidas de las paredes, e indicó dónde se encuentra la puerta y la ventana que hay en su cocina y cuyas medidas son:</p> <ul style="list-style-type: none"> • Puerta de 190 (alto) x 70 (ancho) cm • Ventana de 100 (alto) x 110 (ancho) cm 	
<p>Tarea 2: Elija el mobiliario</p> <p>Suponga que los muebles que desea comprar son los que se indican a continuación. Elíjalos y colóquelos según se muestra en la figura adjunta:</p> <ol style="list-style-type: none"> 1- Rinconera (80cm de ancho) 2- Vitrocerámica+Horno 3- Vitrina (80cm, con 2 puertas) 4- Campana extractora metálica 5- Lavaplatos <p style="text-align: right; color: green;">Anote el presupuesto: _____ (euros)</p>	
<p>Tarea 3: Retoques finales</p> <ol style="list-style-type: none"> 1- Coloque el lavaplatos al lado de la vitrocerámica. 2- Un familiar le va a regalar una campana, por tanto, quítela. 3- Si considera que queda algo por hacer (antes de proceder a encargar la cocina), hágalo. 	<p>4- Busque el presupuesto detallado (donde pueda consultar el precio de cada elemento adquirido):</p> <p style="color: green;">Anote nº de artículos incluidos: _____ y presupuesto final (euros): _____</p>
<p>Tarea 4: Guarde o encargue el boceto realizado</p>	

Figura 4.4 Hoja de tareas entregada a los participantes, en la que se describe tanto el escenario de uso de la aplicación como los tres grupos de tareas a realizar.

4.5. Medidas empíricas

4.5.1. Medidas objetivas: tiempos y errores.

El primer dato objetivo a considerar en este estudio es el *tiempo invertido en completar las tareas*. En este sentido, denotaremos como T_{task1} , T_{task2} y T_{task3} al tiempo, en segundos, que los participantes dedicaron a cada uno de los tres grupos de tareas considerados. Dichos grupos se etiquetaron en la *hoja de tareas* (Figura 4.4) como “Tarea 1: Especificar cómo es la cocina”, “Tarea 2: Elegir el mobiliario” y “Tarea 3: Retoques finales”, respectivamente. Finalmente, T_{total} hará referencia al tiempo total empleado en completar todas las tareas, es decir, a la suma de T_{task1} , T_{task2} y T_{task3} . La Tabla 4.2 refleja las medidas de los *tiempos* (en segundos) correspondientes a cada participante.

Tabla 4.2 *Tiempos* (en segundos) invertidos por cada participante *en completar las tareas*.

Time on task (in seconds)														
#	Ttask1			Ttask2			Ttask3			Ttotal				
user	T-order	CE	DM	GDI	diff	DM	GDI	diff	DM	GDI	diff	DM	GDI	diff
1	DM-GDI	0	1023	512	511	1108	345	763	460	265	195	2591	1122	1469
2	GDI-DM	0	863	380	483	686	560	126	692	491	201	2241	1431	810
3	DM-GDI	0	590	510	80	800	280	520	770	286	484	2160	1076	1084
4	GDI-DM	0	185	155	30	597	206	391	508	176	332	1290	537	753
5	DM-GDI	1	425	330	95	690	420	270	420	210	210	1535	960	575
6	GDI-DM	1	280	371	-91	318	487	-169	1075	377	698	1673	1235	438
7	DM-GDI	1	150	135	15	155	220	-65	541	105	436	846	460	386
8	DM-GDI	2	440	182	258	500	336	164	380	168	212	1320	686	634
9	GDI-DM	2	171	218	-47	284	283	1	381	244	137	836	745	91
10	DM-GDI	2	400	350	50	355	165	190	565	205	360	1320	720	600
11	GDI-DM	2	122	120	2	103	207	-104	285	118	167	510	445	65
12	DM-GDI	2	406	150	256	190	303	-113	488	105	383	1084	558	526
13	GDI-DM	2	277	252	25	231	225	6	288	318	-30	796	795	1
14	DM-GDI	3	370	120	250	216	145	71	274	164	110	860	429	431
15	GDI-DM	3	150	165	-15	156	215	-59	254	150	104	560	530	30
16	DM-GDI	3	250	173	77	245	247	-2	405	250	155	900	670	230
17	GDI-DM	3	134	228	-94	182	279	-97	280	203	77	596	710	-114
18	DM-GDI	3	330	205	125	123	205	-82	310	140	170	763	550	213
19	GDI-DM	3	355	150	205	110	255	-145	120	105	15	585	510	75
20	DM-GDI	3	274	138	136	220	136	84	233	203	30	727	477	250
	Median		305	194	79	238	251	4	393	203	183	880	678	409
	Mean		360	242	118	363	276	88	436	214	222	1160	732	427
	SD		236	124	168	276	110	242	219	99	179	605	288	397

El otro gran dato cuantitativo de interés es el *número de errores* cometidos, tanto el global (E_{total}), como según el tipo del error (E_{slight} , $E_{moderate}$, y E_{severe}). Se han considerado tres categorías de errores en función del grado de severidad de los mismos:

- El *error grave (severe)* es aquel se producía cuando el usuario estaba totalmente “atascado” o bloqueado con algo de la interfaz que le impedía finalizar la tarea. Se decidió que el moderador o facilitador de la prueba siempre ayudase a los participantes ante este tipo de error, de manera que siempre pudiesen completar la tarea (algo que realmente no habrían podido conseguir sin la ayuda recibida) con objeto de poder recopilar datos del resto del proceso.
- Los *errores moderados (moderate)* son aquellos que, sin llegar a bloquear al usuario, sí que podrían alterar el resultado de la tarea. Podría ocurrir que el propio usuario acabara dándose cuenta del error, solucionándolo por sí solo, o que le pasara inadvertido, en cuyo caso, el facilitador de la prueba debería proporcionarle algún tipo de advertencia o aviso, a fin de que el usuario intentase solucionarlo, pero sin indicarle cómo solucionarlo.
- Finalmente, los *errores leves (slight)* podrían no clasificarse como errores reales, ya que hacen referencia a situaciones que provocan al usuario algún tipo de retraso, incertidumbre o duda, o problema, pero que el participante identifica como tales y consigue resolverlos sin ningún tipo de aviso ni de ayuda por parte del moderador.

Además, estos errores se pueden considerar un complemento importante y necesario para el análisis de los *tiempos*. En concreto, los más importantes serían los errores *graves*, ya que, sin la ayuda del moderador, el tiempo necesario para completar la tarea se vería drásticamente afectado, o incluso peor, podría impedir que el usuario pudiese completar la tarea (requiriendo, potencialmente, un tiempo que tendería a infinito), situación que se quería evitar en este estudio.

La Tabla 4.3 recoge el número de errores por participante, y la Tabla 4.4 la lista concreta de errores detectados durante las pruebas, explicitando, entre otros aspectos, el tipo de interfaz y la tarea en la que éstos ocurrieron.

Tabla 4.3 Resumen de los errores (Ligeros/Moderados/Severos) detectados durante las pruebas, incluyendo el tipo de interfaz involucrada (MD o IGO) y las tareas en que sucedieron (1, 2, y/o 3).

Type	Inter face	#Task	Incidence description	Effect	Assistance
Slight	MD	1	UNDO was confused with CLEAR when wanting to remove an item like a door (the user thought the system had failed because the item was not deleted)		
	MD	1	The user accepted the default shape of the room; later he was aware of it and changed it		
	MD	1	Window dimensions were confused with its distance to the walls	Put the proper window by chance	
	MD	2 & 3	Tried to change the position of a door/window, while another operation (the placement of another door/window) was being in progress		
Moderate	IGO	1	Not consciously placing a window, believing it did not matter	Consciously leaves an incomplete task (though not decisive for a correct final result)	
	IGO	2	Instead of moving a piece of furniture (because this option was not found), deleted it and put another one at the targeted location	Inadequate procedure	
	IGO	3	The user could not see the kitchen in 3D	Consciously leaves an incomplete task (though not decisive for a correct final result)	
	MD	1	Entered the measurements in the default kitchen shape (without choose the correct shape), and did not realize below	Wrong final result, unconsciously	Warning
	MD	1	The participant was unable to change the shape of the default kitchen, so he decided to start again	Inadequate procedure	
	MD	2 & 3	UNDO was confused with CLEAR (wanting to remove a cabinet), throughout the experiment	Persistent misinterpretation of an interaction (Inadequate procedure)	
	MD	2 & 3	Learnt not to DRAG to move furniture → Choose delete and put it back at the targeted location	Inadequate procedure	
	MD	3	Chosen wrong furniture (high corner instead of low)	Wrong final result, unconsciously	Warning
	MD	3	Oblivious to the only sub-task is not explicitly stated: “choose the common elements (granite, wood and handles)”	Wrong final result, unconsciously	Warning
Severe	IGO	3	With everything done, the user would choose “design a plan” (which implies starting again) instead of the “order kitchen” option, so intend to start again	It would enlarge the experiment time without limit	Help
	MD	2	Does not learn to SCROLL (necessary to choose the desired furniture)	Blocking	Help
	MD	2	Does not learn to DRAG (needed for “placing” the furniture)	Blocking	Help
	MD	2 & 3	Does not learn to DELETE furniture	Blocking	Help
	MD	3	Does not learn to DRAG (necessary to relocate furniture)	Blocking	Help
	MD	3	Does not learn to DELETE furniture → Try to start again	It would enlarge the experiment time without limit	Help

Tabla 4.4 Cantidad de *errores* por participante y tipo de interfaz: MD vs IGO (GDI en Inglés).

#	user	T-order	CE	#Errors						Total	
				Slight		Moderate		Severe		DM	GDI
				DM	GDI	DM	GDI	DM	GDI	DM	GDI
1	DM-GDI	0	0	2	0	2	1	1	0	5	1
2	GDI-DM	0	0	0	0	1	1	2	0	3	1
3	DM-GDI	0	0	2	0	2	0	3	0	7	0
4	GDI-DM	0	0	0	0	1	0	0	0	1	0
5	DM-GDI	1	0	0	0	3	0	3	0	6	0
6	GDI-DM	1	0	0	0	2	0	1	0	3	0
7	DM-GDI	1	0	0	0	2	0	0	0	2	0
8	DM-GDI	2	0	1	0	1	0	1	0	3	0
9	GDI-DM	2	0	0	0	0	0	0	0	0	0
10	DM-GDI	2	0	0	0	2	0	0	0	2	0
11	GDI-DM	2	0	0	0	1	0	0	0	1	0
12	DM-GDI	2	0	0	0	3	0	1	0	4	0
13	GDI-DM	2	0	1	0	2	0	0	0	3	0
14	DM-GDI	3	0	1	0	0	0	0	0	1	0
15	GDI-DM	3	0	0	1	3	0	0	0	3	1
16	DM-GDI	3	0	1	0	2	1	0	0	3	1
17	GDI-DM	3	0	1	0	1	0	0	0	2	0
18	DM-GDI	3	0	0	0	1	0	0	1	1	1
19	GDI-DM	3	0	1	0	2	0	0	0	3	0
20	DM-GDI	3	0	1	0	0	0	0	0	1	0
Median				0	0	2	0	0	0	3	0
Mean				0,55	0,05	1,55	0,15	0,60	0,05	2,70	0,25
SD				0,69	0,22	0,94	0,37	0,99	0,22	1,78	0,44

4.5.2. Medidas subjetivas; el cuestionario *Post-interfaz* y el cuestionario *Comparativo final*

Para la recogida de los datos subjetivos se utilizaron dos cuestionarios distintos. El primer conjunto de datos se recopiló empleando un cuestionario, que denominamos *cuestionario Post-interfaz*, y que los participantes tenían que rellenar dos veces, una por cada tipo de interfaz que iban a testear: usaban la primera interfaz y rellenaban el cuestionario una vez; usaban la otra interfaz, y lo rellenaban por segunda vez, pero ahora opinando sobre el otro estilo de interacción. Este cuestionario incluye siete preguntas (Q_i). Todas ellas se puntúan con un valor numérico, excepto la quinta (Q_5), según se indica en la Tabla 4.5 y en la página siguiente.

El cuestionario *SUS (System Usability Scale)* [12, 13], al considerarse un estándar, fue nuestra primera opción a usar como *cuestionario post-interfaz*, pero finalmente consideramos más adecuado sustituir sus preguntas, muy genéricas, por otras más concretas



y relevantes para nuestro estudio. Dichas preguntas, así como las aclaraciones que se incluían para los participantes, son las siguientes:

- Q₁: “Considera que la aplicación le ha ayudado a saber *qué* hacer en cada momento?” (según una escala lineal, de 1 a 7, donde un 1 quiere decir “nada”, y 7 “completamente”).
Aclaración: ¿Sentía que le indicaba qué o cuál objetivo había que abordar en cada momento? No “cómo” conseguirlo, sino “qué” es lo que había que conseguir. Por ejemplo: cuándo era el momento de elegir la forma de la cocina, cuándo tocaba colocar puertas/ventanas, cuándo elegir un mueble o electrodoméstico, cuándo moverlo, borrarlo, elegir los materiales comunes, etc.
- Q₂: “¿Considera que la aplicación le ha ayudado a saber *cómo* hacerlo (aquello que necesitaba hacer)?” (según una escala lineal, de 1 a 7, donde un 1 quiere decir “nada”, y 7 “completamente”).
Aclaración: ¿Sentía que ésta le indicaba “cómo” hacer las cosas que había que hacer, los pasos concretos a realizar? Por ejemplo, ¿sabía cómo introducir la medida de un muro concreto de la cocina (sabiendo que era el momento de introducirla)? o ¿cómo colocar una puerta en el muro deseado (una vez que ya sabía que tenía que ponerla)? ¿sabía cómo consultar el presupuesto (cuando así lo deseaba), o cómo poner los muebles, o cómo quitarlos, desplazarlos, etc.?
- Q₃: “¿Qué fue más difícil? [1] saber *qué* hacer, [2] saber *cómo* hacerlo, [0] ambas cuestiones fáciles, o [3] ambas difíciles”.
Nota: Las puntuaciones fueron asignadas a cada opción de acuerdo a un rango desde el mejor (“ambas cuestiones fáciles”) al peor escenario (“ambas difíciles”).
- Q₄: “¿Hubiera agradecido o necesitado algún otro sistema de ayuda?” (según una escala lineal, de 1 a 5, donde un 1 quiere decir “ninguno”, y 5 “todos los posibles”).
Aclaración: Manual de usuario, ayuda en línea, ayuda de un experto, video explicativo...
- Q₅: (Opción múltiple) “¿Para qué frecuencia de uso considera útil y adecuada la aplicación? [A] adecuada para una única vez, [B] para usos esporádicos y ocasionales (como una vez al año), [C] para una vez al mes, y/o [D] para uso diario”.
- Q₆: “¿Usaría una aplicación similar para el diseño de su próxima cocina?” (según escala lineal, de 1 a 5, donde un 1 quiere decir “nunca”, y 5 “si, seguro”).
Aclaración: No considere si le gusta su aspecto, o si está más o menos completa, o si le faltan muebles o no le gustan los que hay, etc., sino sólo en si cree que sabría usarla.
- Q₇: “Para resumir, puntúe la facilidad de uso de la aplicación” (según una escala lineal, de 1 a 7, donde un 1 quiere decir que es muy fácil de usar, y un 7 que es muy difícil de usar).

La Tabla 4.5 recoge las respuestas de cada participante al *cuestionario Post-interfaz*.

Tabla 4.5 Respuestas de cada participante al *questionario Post-interfaz*

#user	Q1		Q2		Q3		Q4		Q5		Q6		Q7	
	[1..7]		[1..7]		[0..3]*		[1..5]		[0,1,2,3]**		[1..5]		[1..7]	
	MD	GDI	MD	GDI	MD	GDI	MD	GDI	MD	GDI	MD	GDI	MD	GDI
1	2	4	2	5	3	0	5	2	C+D A+B+C+D	1	5	2	5	
2	3	7	1	7	2	0	3	2	C+D A+B+C+D	3	3	2	5	
3	2	6	2	6	2	0	4	1	C+D A+B+C+D	3	4	2	6	
4	2	7	3	7	2	0	2	1	C+D A+B+C+D	3	5	5	7	
5	3	7	2	5	2	0	4	2	D A+B+C+D	2	4	3	6	
6	4	7	6	6	2	0	2	1	D A+B+C+D	3	5	3	6	
7	4	5	3	7	2	0	2	1	C+D A+B+C+D	3	5	5	7	
8	2	7	2	6	2	0	4	1	C+D A+B+C+D	2	4	3	6	
9	6	6	2	6	2	0	2	1	C+D A+B+C+D	2	5	3	6	
10	3	6	4	6	2	0	1	1	A+B+C+D A+B+C+D	4	4	5	6	
11	5	7	4	6	2	0	5	1	B+C+D A+B+C+D	5	5	5	6	
12	2	6	4	6	2	0	4	2	C+D A+B+C+D	2	5	3	6	
13	5	7	4	7	3	0	4	1	D A+B+C+D	3	5	5	7	
14	1	6	5	6	1	0	3	1	B+C+D A+B+C+D	4	5	5	6	
15	3	7	3	6	2	0	3	1	C+D A+B+C+D	2	4	3	6	
16	3	6	2	6	2	0	4	1	D A+B+C+D	4	5	4	6	
17	2	7	4	6	2	0	2	1	A+B+C+D A+B+C+D	5	5	5	7	
18	3	6	2	7	1	0	5	5	C+D A+B+C	3	5	5	7	
19	5	6	4	6	1	0	4	2	D A+B+C+D	5	5	4	6	
20	5	6	4	6	2	0	2	1	C+D A+B+C+D	5	5	6	7	
Median	3,0	6,0	3,0	6,0	2,0	0,0	3,5	1,0		3,0	5,0	4,0	6,0	
Mean	3,3	6,3	3,2	6,2	2,0	0,0	3,3	1,5	(C+D) (A+B+C+D)	3,2	4,7	3,9	6,2	
SD	1,4	0,8	1,3	0,6	0,5	0,0	1,2	0,9		1,2	0,6	1,3	0,6	

El segundo grupo de datos subjetivos se obtuvo gracias a un *questionario Comparativo* final, que los participantes debían completar al terminar su participación, y una vez que ya habían utilizado ambas interfaces. Este cuestionario los hacía comparar directamente ambos estilos de interacción a través de las siguientes ocho preguntas (C_i):

- C_1 : “¿Con qué interfaz es más fácil saber *qué* hay que hacer en cada momento?”
- C_2 : “¿Con cuál le parece más fácil saber *cómo* hacer lo que quiere hacer?”
- C_3 : “¿Cuál debería incluir más sistemas de ayuda?”
- C_4 : “¿Cuál considera que es más fácil de usar y requiere menos aprendizaje?”
- C_5 : “¿Con qué tipo de interfaz cree que se trabaja más rápido?”
- C_6 : “Si un experto en informática, ocasionalmente, necesitase utilizar una de las dos ¿cuál le recomendaría?”
- C_7 : “¿Y a un profesional del diseño y venta de cocinas, para usarla diariamente en su trabajo?”
- C_8 : “Si realmente tuviese que diseñar y amueblar su cocina ¿Cuál elegiría?”

La Tabla 4.6 recoge las respuestas de cada participante al *questionario Post-test*.

Tabla 4.6 Respuestas de cada participante al *cuestionario Post-interfaz*

#user	C1	C2	C3	C4	C5	C6	C7	C8
1	GDI	GDI	DM	GDI	GDI	GDI	GDI	GDI
2	GDI	GDI	DM	GDI	GDI	GDI	GDI	GDI
3	GDI	GDI	DM	GDI	GDI	GDI	GDI	GDI
4	GDI	GDI	DM	GDI	GDI	GDI	DM	GDI
5	GDI	GDI	DM	GDI	GDI	GDI	GDI	GDI
6	GDI	GDI	DM	GDI	GDI	DM	DM	GDI
7	GDI	GDI	DM	GDI	GDI	GDI	GDI	GDI
8	GDI	GDI	DM	GDI	GDI	GDI	GDI	GDI
9	GDI	GDI	DM	GDI	GDI	GDI	GDI	GDI
10	GDI	GDI	DM	GDI	GDI	GDI	GDI	GDI
11	GDI	GDI	DM	GDI	GDI	GDI	GDI	GDI
12	GDI	GDI	DM	GDI	GDI	DM	DM	GDI
13	GDI	GDI	DM	GDI	GDI	GDI	GDI	GDI
14	GDI	GDI	DM	GDI	GDI	GDI	DM	GDI
15	GDI	GDI	DM	GDI	GDI	GDI	GDI	GDI
16	GDI	GDI	DM	GDI	DM	GDI	DM	GDI
17	GDI	GDI	DM	GDI	GDI	GDI	GDI	GDI
18	GDI	GDI	DM	GDI	GDI	GDI	GDI	GDI
19	GDI	GDI	DM	GDI	GDI	GDI	GDI	GDI
20	GDI	GDI	DM	GDI	GDI	GDI	GDI	GDI
GDI	100%	100%	0%	100%	95%	90%	75%	100%
DM	0%	0%	100%	0%	5%	10%	25%	0%

4.6. Procedimiento

El tiempo estimado para que un participante completase todo el experimento, incluyendo la presentación del mismo, las instrucciones, la realización de las tareas con los dos tipos de interfaces, y el rellenado de los cuestionarios, osciló entre 45 y 75 minutos. Las pruebas se realizaron individualmente, ya fuese en la propia casa del participante o en una sala habilitada para ello en la universidad, pero siempre en entornos libres de elementos distractores, y con la única presencia del moderador o facilitador de la prueba. Se usó el mismo ordenador portátil para todos los tests. Esto facilitó su reubicación, al mismo tiempo que resultaba menos intimidante y más familiar para los participantes que cualquier otro tipo de equipo.

Los dispositivos de entrada utilizados fueron el ratón o el trackpad (a elección del participante) y el teclado. Se utilizó un software de grabación pantalla y del audio (Quicktime de Apple) para registrar, tanto el proceso de interacción con ambas interfaces, como el de rellenado de los cuestionarios. Ello permitió, posteriormente, y mediante un minucioso proceso de revisión de las grabaciones, anotar las mediciones de los tiempos invertidos en la realización de las tareas y del número de errores cometidos, así como revisar los comentarios emitidos por los participantes durante la prueba.

Antes del estudio principal se realizó una prueba piloto con un grupo de 4 participantes. Esto nos permitió ajustar el procedimiento para la prueba, la duración de la misma, la hoja de tareas, y las preguntas y aclaraciones de los cuestionarios.

El procedimiento a seguir para la realización del experimento final constaba de los siguientes pasos:

1. Todos los participantes firmaron un *consentimiento informado* y fueron informados sobre el hecho de que las pruebas eran de carácter voluntario y que podían abandonarlas en cualquier momento sin tener que dar ninguna justificación. También se les insistió en que el objetivo no era evaluarlos a ellos, sino a las interfaces en estudio.
2. El moderador procedía a leerles el guion con las *instrucciones* sobre la prueba, con objeto de asegurarse de que cada participante recibiese la misma información sobre el carácter de la misma, sobre el proceso a seguir, y sobre el propósito del estudio, del cual apenas se les dio detalles; tan solo que queríamos que no ayudasen a encontrar errores en ambas interfaces. También se les insistía y aclaraba que estas pruebas no estaban destinadas a realizar ninguna evaluación personal ni psicológica del participante, sino al contrario, que gracias a él podríamos encontrar los errores o puntos débiles de las dos interfaces involucradas en el experimento. Al terminar con las instrucciones, el moderador procedía a responder las preguntas o a aclarar las dudas de los participantes, si las hubiera.
3. Los participantes debían completar un *cuestionario Pre-test (preliminary background questionnaire)* mediante el cual se recopilaban las características más relevantes del usuario, como su edad, sexo, nivel informático y experiencia con la tecnología, conocimiento sobre el dominio de la tarea, etc. Esto nos permitiría verificar que los participantes realmente cumplían con los requisitos solicitados y descritos en la sección 4.2.
4. Los usuarios recibían la *hoja de tareas* en la que se le describían los tres grupos de tareas a realizar (detalladas en la sección 4.3 y en la Figura 4.4).
5. Como ya se comentó anteriormente, se le pidió a cada participante que usara ambas versiones de la aplicación, pero siguiendo una *estrategia contrabalanceada*, es decir,

alternando el orden de uso de ambas interfaces, de un usuario a otro. También se permitía la técnica *Thinking Aloud*, pero de forma muy relajada. Los participantes no estaban “obligados” a comentar u opinar sobre lo que estaban haciendo, pero sí eran libres de expresarse en voz alta. El moderador trató de no interferir ni responder dudas de los participantes, excepto ante: a) "errores graves", b) algunos "errores moderados", o c) para redirigir la prueba si el usuario se entretenía o despistaba. Los errores concretos relacionados con a) y b) y el tipo de asistencia brindada para cada uno de ellos se muestran en la Tabla 4.4. En algunos de estos casos, se insistía en que estábamos evaluando las interfaces, y no al participante.

6. Justo al terminar de usar una interfaz, pero antes de comenzar con la otra, los participantes debían completar el correspondiente *cuestionario Post-interfaz* (detallado en la sección 4.5.2) sobre aspectos subjetivos y específicos relativos a la usabilidad de la forma de interacción que acababan de experimentar, los mismos para cada interfaz.
7. Finalmente, los participantes terminaban la prueba completando el *cuestionario Comparativo* final (detallado en la sección 4.5.2), en el que ya sí, explícitamente, se les pedía que comparasen directamente ambas formas de interacción y sus correspondientes interfaces, y expresaran directamente sus preferencias.

4.7. Resultados

Por un lado, tenemos un *diseño con medidas repetidas* con dos *condiciones*, ya que el mismo participante debía usar los dos tipos de interfaces. Por otro lado, como los datos no siguen una distribución normal tampoco cumplen los requisitos para poder realizar tests paramétricos. Por lo tanto, para verificar el nivel de significación de los resultados, el método estadístico más apropiado para nuestro caso es el test *no paramétrico* para *muestras pareadas* de Wilcoxon [177], también conocido como el *test de rango con signo* de Wilcoxon.

A continuación, presentamos los resultados en el siguiente orden: primero, en la sección 4.7.1, los correspondientes a las medidas objetivas sobre tiempos y número de errores; segundo, en la sección 4.7.2, los resultados de las medidas subjetivas sobre la usabilidad correspondientes a cada interfaz, de manera independiente; tercero, en la sección 4.7.3, las

preferencias entre los dos tipos de interacción. Finalmente, en la sección 4.7.4, comentamos los resultados de un pequeño y complementario estudio realizado con Eye-Tracker.

4.7.1. Resultados objetivos: tiempos y errores.

En la Tabla 4.7 (y, gráficamente, en la Figura 4.5) y en la Tabla 4.8 (y Figura 4.6) se muestran, respectivamente, varios resultados estadísticos relativos a los *tiempos* invertidos en completar las tareas (T_{task1} , T_{task2} , T_{task3} , and T_{total}) y al número de *errores* cometidos (E_{slight} , $E_{moderate}$, y E_{severe}).

En concreto, los tests de Wilcoxon correspondientes a T_{task1} , T_{task3} y especialmente para T_{total} revelaron diferencias significativas ($p < 0.01$) a favor de IGO, es decir, los participantes invirtieron menos tiempo en completar dichas tareas usando la interfaz IGO que con la interfaz MD (Tabla 4.5). Solo el test aplicado a T_{task2} , una tarea de marcado carácter repetitivo al requerir realizar cinco veces la misma operación (elegir un mueble y colocarlo en la cocina), no mostró diferencias significativas entre las dos formas de interacción. Por lo tanto, podemos aceptar parcialmente **H1**.

Tabla 4.7 Estadísticas correspondientes a los *tiempos invertidos en completar de las tareas* (T_{task1} , T_{task2} , T_{task3} , y T_{total}).

Tiempos para completar las tareas (en segundos)								
	TASK 1: “Designing the kitchen” T_{task1}		TASK 2: “Furnishing the kitchen” T_{task2}		TASK 3: “Retouching and ordering” T_{task3}		TOTAL T_{total}	
	MD	IGO	MD	IGO	MD	IGO	MD	IGO
Mínimo	122	120	103	136	120	105	510	429
1st Qu.	181	150	175	206	283	147	754	525
Mediana	305	193	238	251	393	203	880	678
3rd Qu.	410	335	524	311	516	253	1374	836
Máximo	1023	512	1108	560	1075	491	2591	1431
Media	360	242	363	276	436	214	1160	732
σ	236	124	276	110	219	99	605	288

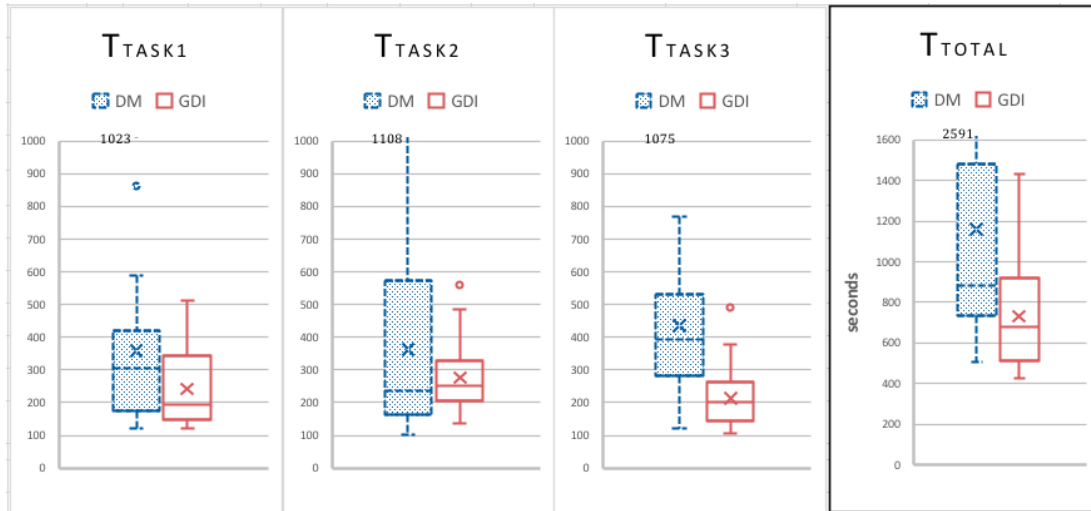


Figura 4.5 Diagramas de cajas y bigotes correspondientes a los tiempos invertidos en completar de las tareas (T_{task1} , T_{task2} , T_{task3} , y T_{total}).

Tabla 4.8 Estadísticas correspondientes al número de errores e incidencias leves, moderadas, severas y totales detectadas (E_{slight} , $E_{moderate}$, E_{severe} and E_{total}).

	Número de Errores							
	Slight		Moderate		Severe		Total	
	MD	IGO	MD	IGO	MD	IGO	MD	IGO
Mínimo	0	0	0	0	0	0	0	0
1º cuartil	0	0	1	0	0	0	1	0
Mediana	0	0	2	0	0	0	3	0
3º cuartil	1	0	2	0	1	0	3	0.2
Máximo	2	1	3	1	3	1	7	1
Media	0.55	0.05	1.55	0.15	0.6	0.05	2.7	0.25
σ	0.69	0.22	0.94	0.37	0.99	0.22	1.78	0.44

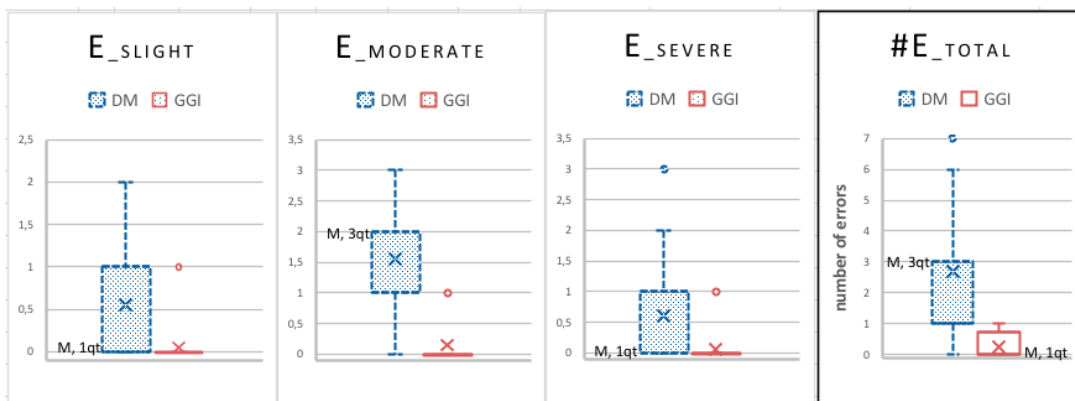


Figura 4.6 Diagramas de cajas y bigotes correspondientes al número de errores e incidencias leves, moderadas, severas y totales detectadas (E_{slight} , $E_{moderate}$, E_{severe} and E_{Total}).



Por otro lado, los tests de Wilcoxon (Tabla 4.9) aplicados al cálculo del número de errores *leves* (E_{slight}), *moderados* (E_{moderate}), *severos* (E_{severe}) y *totales* (E_{total}), también mostraron diferencias significativas ($p < 0.01$) a favor de IGO. O sea, el uso de la interfaz MD supone un aumento significativo en el número de incidencias o errores, en comparación con los cometidos al utilizar la interfaz IGO. Por lo tanto, podemos aceptar **H2**.

Tabla 4.9 Resultados de los tests de Wilcoxon sobre los tiempos invertidos en completar de las tareas y sobre el número de errores.

Time on task	p-value	W	type of test		#Errors	p-value	W	type of test
T_{task1}	0.00250	180.5	one-tailed		E_{slight}	0.00750	50.5	one-tailed
T_{task2}	0.29430	134.0	two-tailed		E_{moderate}	0.00019	136.0	one-tailed
T_{task3}	0.00007	207.5	one-tailed		E_{severe}	0.01080	33.0	one-tailed
T_{total}	0.00001	204.0	one-tailed		E_{total}	0.00009	171.0	one-tailed

Finalmente, la Tabla 4.10 resalta cuán evidentes son los resultados, si nos fijamos en los siguientes porcentajes: primero, de acuerdo con T_{total} , el 95% de los usuarios terminaron antes usando IGO que usando la interfaz MD; segundo, el 91% de los errores ocurrieron bajo la interacción MD, frente a solo un 9% que tuvieron lugar al usar IGO.

Tabla 4.10 Algunos resultados interesantes relacionados con los tiempos invertidos en completar las tareas y sobre el número de errores.

Estilo de Interacción	% participantes que terminaron antes la prueba	% Errores		
		Slight	Moderate	Severe
MD	5%	92%	91%	92%
IGO	95%	8%	9%	8%

4.7.2. Resultados subjetivos

En la sección 4.5.2 indicamos que el *questionario Post-interfaz*, que incluía siete preguntas (Q_i), debía ser rellenado dos veces por cada participante, cada una justo al terminar de realizar las tareas con un tipo de interfaz, antes de comenzar con la siguiente.

La Tabla 4.11 (y, gráficamente, la Figura 4.7) muestran las principales estadísticas asociadas a cada Q_i , excepto las de Q_5 , cuya respuesta no se corresponde con valores numéricos. De nuevo, los tests de Wilcoxon correspondientes vuelven a ofrecer diferencias

significativas ($p < 0.001$ para cada Q_i) a favor de IGO. Por lo tanto, podemos aceptar **H3**, y concluir que los OU prefieren la interacción guiada a la típica MD.

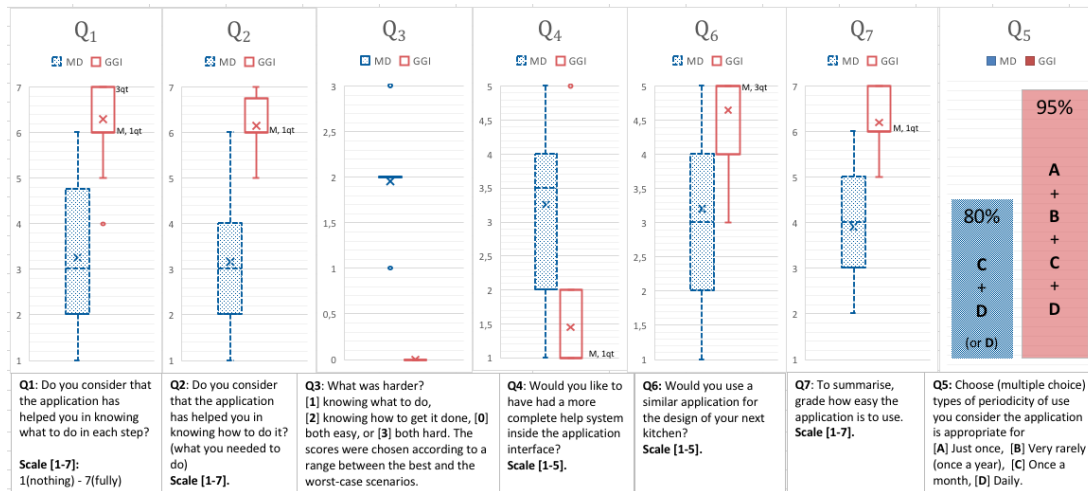


Figura 4.7 Diagramas de cajas y bigotes correspondientes a las 7 preguntas (Q_i) del cuestionario *Post-interfaz*. La escala para Q_1 , Q_2 y Q_7 va de 1 (nada) a 7 (completamente); para Q_3 va de 0 a 3; para Q_4 y Q_6 , de 1 a 5. Y para Q_5 se muestra el porcentaje, para cada interfaz, de la opción elegida con más frecuencia.

Con respecto a Q_5 , cabe destacar que el 80% de los participantes no considera que la interacción MD sea apropiada para ser utilizada una sola vez, ni para un uso ocasional. Inesperadamente, el 100% de los participantes, incluidos los dos profesionales en el dominio de la tarea, consideran que la interacción IGO sí es apropiada, no solo para un único uso o para el ocasional, sino también para uso frecuente e incluso diario.

4.7.3. Resultados comparativos directos; preferencias de los participantes

Con respecto a los datos recogidos por el último cuestionario, compuesto por ocho preguntas (C_i) que implican una comparación directa entre ambas interfaces, la Tabla 4.12 muestra el porcentaje de participantes que prefieren una forma de interacción sobre la otra. Como se puede observar, el 100% de ellos consideran que con IGO es más fácil saber “qué es lo que hay que hacer” en cada momento y “cómo hacerlo”; también opinan que es más fácil de usar y que requiere menos entrenamiento esta interfaz guiada que la de MD, y que

sería el tipo de interacción que preferirían si realmente tuviesen que elegir una para amueblar su próxima cocina. Además, un 90% recomendarían usar IGO antes que MD a un UO experto en informática, y un 75% (incluidos los dos profesionales que participaron en el estudio) también se lo recomendarían a un profesional del diseño de cocinas para su uso diario. Estos datos serán discutidos en la siguiente sección.

Tabla 4.11 Estadísticas correspondientes a las respuestas (Q_i) al cuestionario Post-interfaz.

Q_i	Descripción	Escala	Media		σ		Mediana		1º cuartil		3º cuartil	
			MD	IGO	MD	IGO	MD	IGO	MD	IGO	MD	IGO
Q ₁	“Considera que la aplicación le ha ayudado a saber <i>qué</i> hacer en cada momento?”	[1-7]	3.25	6.30	1.37	0.80	3	6	2	6	4.25	7
Q ₂	“¿Considera que la aplicación le ha ayudado a saber <i>cómo</i> hacerlo (aquello que necesitaba hacer)?”	[1-7]	3.15	6.15	1.26	0.58	2	6	3	6	4	6.25
Q ₃	“¿Qué fue más difícil? [1] saber qué hacer, [2] saber cómo hacerlo, [0] ambos fáciles, o [3] ambos difíciles”	[0-3]	2.00	0	0.51	0	2	0	2	0	2	0
Q ₄	“¿Hubiera agradecido o necesitado algún otro sistema de ayuda?”	[1-7]	3.25	1.45	1.20	0.94	2	1	3.5	1.45	4	2
Q ₅	“¿Para qué frecuencia de uso considera útil y adecuada la aplicación? [A] para una única vez, [B] para usos esporádicos u ocasionales (como una vez al año), [C] para una vez al mes, y/o [D] para uso diario”	Opción múltiple	C+D	A+B +C+D								
Q ₆	“¿Usaría una aplicación similar para el diseño de su próxima cocina?”	[1-5]	3.20	4.65	1.96	0.58	2	4	3	5	4	5
Q ₇	“Para resumir, puntúe la facilidad de uso de la aplicación (según una escala lineal, de 1 a 7, donde un 1 quiere decir que es muy fácil de usar, y un 7 que es muy difícil de usar)”	[1-7]	3.90	6.20	1.25	0.61	3	6	4	6	5	7



Tabla 4.12 Resultados de las respuestas al Cuestionario Comparativo: porcentaje de participantes que prefieren un estilo de interacción al otro.

C_i	Descripción	MD	IGO
C ₁	“¿Con qué interfaz es más fácil saber <i>qué</i> hay que hacer en cada momento?”	0	100
C ₂	“¿Con cuál le parece más fácil saber <i>cómo</i> hacerlo?”	0	100
C ₃	“¿Cuál debería incluir más sistemas de ayuda?”	100	0
C ₄	“¿Cuál considera que es más fácil de usar y requiere menos aprendizaje?”	0	100
C ₅	“¿Con cuál cree que se trabaja más rápido?”	5	95
C ₆	“Si un experto en informática, ocasionalmente, necesitase utilizar una de ellas ¿cuál le recomendaría?”	10	90
C ₇	“¿Y a un profesional del diseño y venta de cocinas, para usarla diariamente en su trabajo?”	25	75
C ₈	“Si realmente tuviese que diseñar y amueblar su cocina ¿Cuál elegiría?”	0	100

A continuación, incluimos algunas de las opiniones expresadas por los participantes, una vez usadas ambas interfaces, que se podrían considerar representativas de la opinión mayoritaria:

- “... [la interfaz IGO] me gustó mucho por la confianza que me daba, al explicarlo todo... el otro sistema [MD] te deja un poco perdido”
- “No importa que no lo sepas, este sistema [IGO] te indica continuamente lo siguiente a realizar”
- “Me gusta más la otra forma [IGO], que es mucho más fácil; ésta [MD] no te dice lo que tienes que hacer, tienes que saberlo tú; la otra sí te guía”

4.7.4. Resultados de un pequeño estudio realizado con Eye-Tracker

En paralelo al principal, se realizó otro estudio complementario, mediante el uso de *Eye-Tracker*, pero solo con tres participantes, y elegidos al azar. El objetivo no era extraer conclusiones importantes de dicho estudio, sino simplemente conocer las regiones de pantalla que los usuarios miraban más frecuentemente con cada tipo de interfaz.

Se decidió ejecutar el *algoritmo k-means*, utilizando el paquete estadístico R, para buscar las regiones calientes a través de la agrupación de puntos. El valor óptimo de k para dicho algoritmo se obtuvo utilizando el *método elbow*. Este algoritmo ayuda a encontrar

automáticamente, a través de una evaluación iterativa, el número óptimo de grupos para una nube de puntos (por ejemplo, [45]). Como se puede observar en la Figura 4.8, este análisis mostró muchos más puntos y un mayor número de clústeres al usar la MD convencional que al usar IGO. El mayor número de puntos se corresponde con el mayor tiempo requerido por los participantes para completar las tareas, mientras que la existencia del doble de grupos (12 frente a 6) se podría asociar con una mayor complejidad. Estos resultados parecían reforzar el estudio principal, pero como no se podían sacar conclusiones relevantes, se decidió simplificar la prueba global y no aplicar este estudio a más participantes.

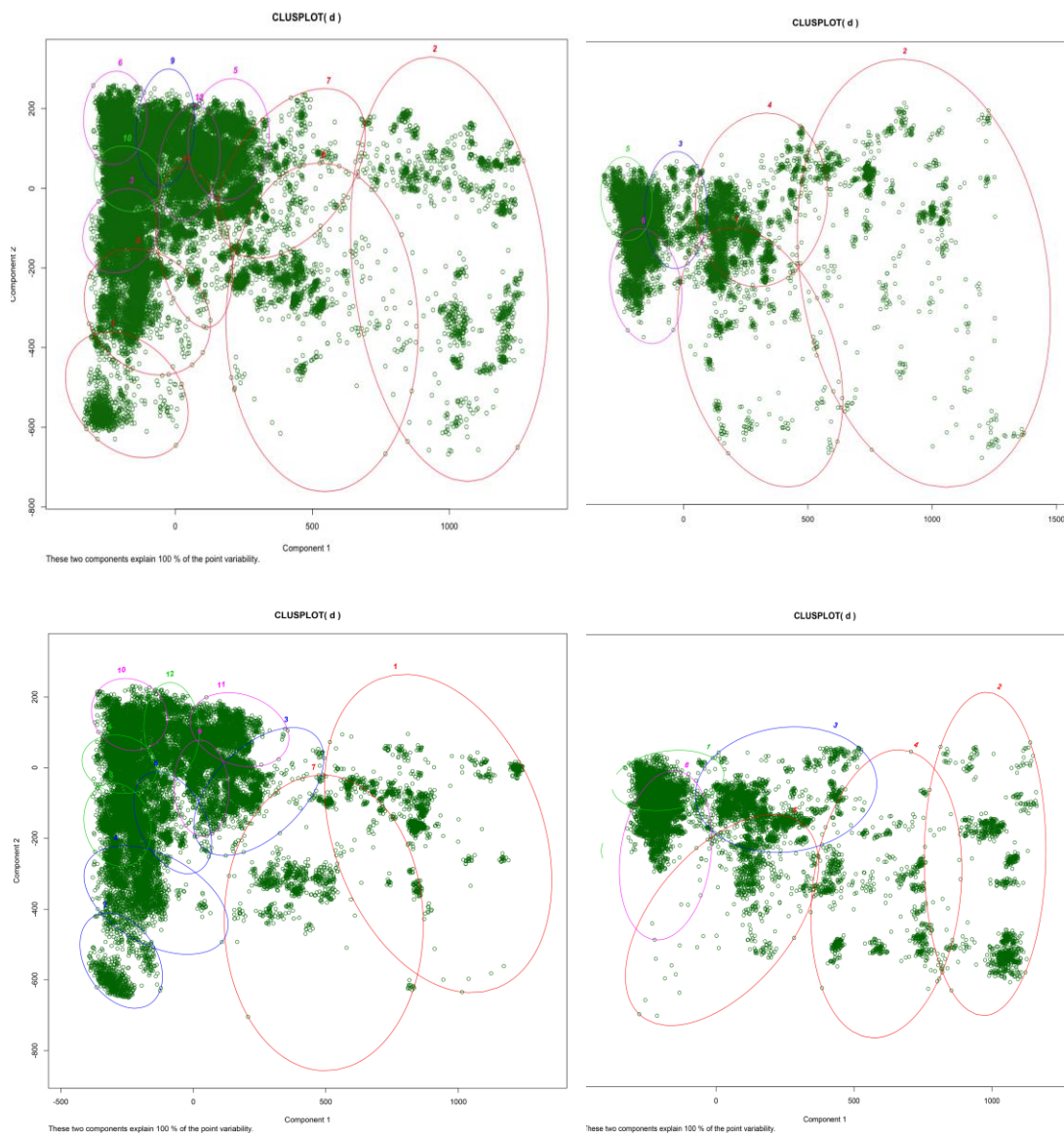


Figura 4.8 Áreas calientes y clústeres asociados a las interfaces de MD (izquierda) e IGO (derecha). Cada fila corresponde a un participante.

4.8. Discusión y conclusiones

Los continuos y recientes cambios en la tecnología están incorporando gran cantidad de nuevos y variados de tipos de usuarios, muchos de los cuales necesitan usar ciertos sistemas en diversos escenarios y de forma ocasional, sin querer o poder permitirse una curva de aprendizaje. Ha habido muchos intentos por facilitarles la interacción con dichos sistemas, como mediante la incorporación de agentes, ventanas de ayuda en línea, tooltips, videos demostrativos, etc. Todos ellos son mecanismos de ayuda o guía, pero que, inevitablemente, agregan una capa extra de material de soporte al tradicional estilo de MD.

Nosotros proponemos la IGO como base para un enfoque completamente diferente: en vez de añadir elementos extra y de esperar que un UO comprenda y use las metáforas, hacemos que sea la propia interfaz la que los guíe durante todo el proceso de interacción. Ahora, el sistema de ayuda forma parte de la interacción; está integrado en la interfaz. El usuario será continuamente guiado tanto en el "qué hacer" como en la forma de conseguirlo, paso a paso, de manera jerárquica, y siempre en función de su objetivo actual. En consecuencia, estos usuarios no tienen mucha libertad de acción. No necesitan estar familiarizados con los widgets gráficos o pensar cuál debería ser el siguiente comando. Y lo más importante para ello, prácticamente no se les requiere conocimientos previos, lo que reduce significativamente la curva de aprendizaje, si es que la hay.

Para poder respaldar las afirmaciones anteriores, se decidió desarrollar dos versiones de una misma aplicación destinada al uso ocasional, una de ellas siguiendo el tradicional y omnipresente estilo de MD, y la otra, en base a la alternativa que proponemos para dicho uso ocasional, la IGO. El propósito era que usuarios ocasionales reales pudiesen "experimentar" ambos tipos de interacción, y decidir cuál consideraban más apropiada para ellos.

Para dicho experimento se establecieron tres hipótesis, **H1**, **H2** y **H3**. Una de ellas, **H2**, está especialmente relacionada con la *eficacia* de la interacción, ya que hace referencia al número de *incidencias* o *errores* cometidos. Parece lógico pensar que el hecho de guiar al usuario hace que cometa menos errores, y que necesite menos asistencia por parte del moderador o facilitador de la prueba. Los resultados obtenidos han demostrado totalmente esta hipótesis: los usuarios ocasionales no expertos en informática tienen menos dudas y cometen menos errores con una interfaz IGO que con la clásica de MD.

Por otro lado, la primera hipótesis, **H1**, estaba más relacionada con la eficiencia, ya que hacía referencia al *tiempo* empleado en completar las tareas, y según la cual, los usuarios ocasionales no expertos trabajan más rápido con IGO que con la MD estándar. Esta hipótesis puede parecer contradictoria: seguir una secuencia de pasos parece ser más lento que manipular elementos directamente. Sin embargo, el análisis de los datos recopilados muestra lo contrario para dos de los tres grupos de tareas establecidos, así como para el tiempo total. El único grupo de tareas para el que, a pesar de suponer un menor tiempo medio, no hubo diferencias significativas, fue para el segundo, el denominado *Task2*. Sin embargo, este caso no nos sorprendió por dos razones. En primer lugar, porque se trataba de una tarea muy repetitiva, que requería realizar 5 veces, prácticamente, la misma operación (elegir un mueble del catálogo y colocarlo en el lugar deseado de la cocina), lo que reducía el factor ocasional y contribuía a su aprendizaje. Y, en segundo lugar, porque la diferencia real en tiempos habría sido muy superior si se hubiese decidido no advertir al usuario de algunos errores moderados ni, especialmente, asistirlo en las situaciones de bloqueo (errores severos).

Por último, la hipótesis **H3** se pudo demostrar de dos maneras: indirectamente, a partir de las respuestas de los participantes a los *cuestionarios Post-interfaz* y, de forma directa, a partir de los *cuestionarios Comparativos* finales. Dichas respuestas resaltaron el hecho de que todos los participantes tenían una clara preferencia por la IGO. Por ejemplo, según C_4 , el 100% de ellos considera que la IGO es más fácil de usar y requiere menos conocimientos y aprendizaje que la MD. Las respuestas de Q_1 a Q_4 y de C_1 a C_3 también dejan claro que los usuarios se sentían más guiados tanto el "qué hacer" en cada momento, como en el "cómo hacerlo", confirmando unos de los principales objetivos de la IGO.

Una de las preguntas de mayor interés para nosotros es Q_5 , ya que nos da una opinión directa sobre la periodicidad o frecuencia de uso para la que los participantes consideran adecuado cada tipo de interacción. A pesar tratarse de una pregunta subjetiva, y que quizás se deba tomar como una indicación de satisfacción y no tanto como una predicción precisa, cabe destacar que el 80% de los participantes opinaba que la MD no es adecuada ni para un único uso, ni para un uso ocasional. En cambio, y algo que sí resultó inesperado fue que el 100% de los participantes, incluidos los dos profesionales en el dominio de la tarea, sí consideraron apropiada a la IGO, no solo para uso ocasional (que era nuestro propósito),

sino también, para uso frecuente. En consecuencia, en lo que respecta a estas opiniones, ningún usuario encontró molesto el uso de la guía que incluye la interfaz IGO.

Los usuarios profesionales en el dominio de la tarea (venta de mobiliario de cocinas) proporcionaron resultados inesperados adicionales. En primer lugar, lograron hacer el trabajo en menos tiempo y con menos errores e incidencias con la IGO. Y, en segundo lugar, también expresaron su preferencia por dicha forma de interacción. En principio, se podría pensar que esta preferencia se podría deber a una menor calidad y potencia de nuestra versión MD, en comparación con las aplicaciones comerciales que ellos usaban. Sin embargo, no parece ser por eso, ya que, aparte de que dicho argumento podría aplicarse también a la versión IGO, ninguno comentó nada en este sentido, siendo conscientes de que su opinión sólo debía estar basada en la forma de trabajar con cada interfaz.

Evidentemente, con este trabajo no tratamos de minimizar la gran importancia que las interfaces de MD tienen, han tenido durante décadas, y tendrán en el futuro. Lo que mostramos con este estudio es solo que un sistema de guía integrado, como el de la IGO, es generalmente una mejor opción para aquellos usuarios que no desean o no pueden permitirse una curva de aprendizaje, ofreciendo una alternativa a diseñadores y desarrolladores de sistemas interactivos que tienen a los *usuarios ocasionales* como sus principales usuarios objetivo, y especialmente, cuando estos tengan poca experiencia con la informática o la tecnología.

Capítulo 5. *AGILE*: Interacción Guiada y Asistida para personas mayores sin experiencia con la tecnología (*novice elderly users*)

El trabajo presentado en este capítulo tiene como objetivo mejorar la usabilidad, accesibilidad y satisfacción con la interfaz de un grupo específico de usuarios, los *novice elderly users*, usuarios sin experiencia con las nuevas tecnologías y con necesidades especiales debidas a la edad, pero intentando, en base a un paradigma inclusivo, beneficiar también al resto de usuarios. Los requisitos derivados de un análisis holístico de estos usuarios, de sus objetivos y necesidades, en el contexto de las actuales tecnologías móviles y táctiles, nos conducirán a la propuesta, desarrollo y evaluación de la *interfaz AGILE*, un tipo de interfaz *guiada y asistida*, derivada de la *Interacción Guiada por Objetivos*, especialmente adecuada para estos usuarios, como demostrarán los resultados de un proceso de evaluación transaccional y el uso de eye-tracker.

5.1. Introducción

5.1.1. ¿Qué es AGILE?

La *Interacción AGILE* [94] es una forma de interacción guiada y asistida, derivada de la *Interacción Guiada por Objetivos*, pero enfocada especialmente a *personas mayores sin experiencia con la tecnología*, con necesidades especiales (principalmente cognitivas) debidas a la edad (*novice elderly users*). El objetivo es permitirles conseguir un rendimiento óptimo en la realización de transacciones digitales, minimizando al máximo el conocimiento previo y el periodo de aprendizaje necesarios para usar adecuadamente la interfaz.

AGILE es el acrónimo de *Assisted and Guided Interaction with No Learning nor Experience Required Interaction*. El nombre es similar a la *metodología AGILE* de Ingeniería de Software [40]. Tienen en común el proceso iterativo típico de cualquier diseño de interfaz centrado en el usuario. Más allá de esa característica, no hay inspiración o relación entre los dos términos.

Los beneficios de la *interacción AGILE* se reflejan en sus interfaces de usuario, diseñadas para ayudar y guiar a estos usuarios inexpertos durante todo el proceso de interacción con la interfaz, facilitando la realización de transacciones digitales concretas, y mejorando el rendimiento y satisfacción de dichos usuarios. El objetivo final es adaptar el diseño de la interfaz, adecuándolo a los requisitos y necesidades propias de estos usuarios, pero intentando beneficiar también al resto de usuarios. Para ello, será necesario realizar previamente una buena identificación y representación de este usuario objetivo. Este será nuestro paso inmediato.

5.1.2. Comprensión de los usuarios y sus necesidades

La variedad, en tipo y número, del rango de discapacidades diferentes existentes, hace difícil el uso del término *usuario discapacitado promedio*. En este trabajo, perseguiremos el *Diseño Inclusivo* [142], lo que implica que es la diversificación, no la homogeneidad, la que debe dirigir las decisiones de la investigación en el contexto de tipos específicos de usuarios, para poder aplicar los resultados al mayor número posible de usuarios, con o sin las mismas características. Para plantear un diseño correcto de interfaz será necesario el conocimiento de las necesidades del usuario al unísono con sus capacidades y limitaciones. Sin embargo, las razones de tal importancia rara vez se describen. Las ventajas intrínsecas son: interfaces más apropiadas, sistemas más usables y accesibles, menos procesos de prueba y error tanto en el uso como en el diseño, y menos necesidad de entrenamiento o aprendizaje por parte del usuario [132]. El estudio de las limitaciones del usuario y de su contexto de uso debe hacerse considerando la interacción con la tecnología como un proceso cognitivo, que involucra una multitud de acciones diferentes, como la percepción, la atención, el reconocimiento, el razonamiento, el pensamiento, el uso de la memoria, etc. Es importante conocer sus mecanismos y los límites intrínsecos de su funcionamiento para adaptar adecuadamente las metodologías de diseño de la interfaz. Por ello, la *Psicología Cognitiva* se está volviendo una disciplina muy importante en la identificación de los problemas que los usuarios tienen cuando usan la tecnología, en función de su perfil y

contexto. En la misma línea de pensamiento, el enfoque del *Diseño inclusivo* antes mencionado aborda los problemas encontrados cuando usuarios con diversidad funcional o alguna discapacidad se enfrentan la tecnología. Persigue dos metas: reducir la exclusión de estos usuarios y la frustración que los usuarios sin discapacidad experimentan al usar los mismos productos. Este trabajo aplica la intersección de ambas disciplinas, *Psicología Cognitiva y Diseño Inclusivo*, para contribuir con guías y métodos que favorezcan el diseño de productos interactivos apropiados.

5.1.3. Nuestro usuario objetivo: personas mayores sin experiencia con la tecnología (novice elderly user)

Desde una perspectiva de *diseño inclusivo*, aceptamos que todos los usuarios tienen necesidades. El problema es encontrar esas necesidades que pueden ser cubiertas y solventadas gracias al diseño de una interfaz efectiva. También consideramos que un verdadero enfoque de usabilidad y accesibilidad para un subconjunto específico de usuarios es transferible a todos o, al menos, a la mayoría de usuarios. La mejora podrá ser apreciada por nuestro usuario objetivo, pero también por aquellos que están fuera del alcance considerado inicialmente (por ejemplo, usuarios sin discapacidad, pero sin experiencia ni tiempo para aprender, usuarios con leves impedimentos cognitivos, etc.). Citando a Newell [108],

“Diseñar interfaces que beneficien a los usuarios con necesidades especiales puede beneficiar a todos los usuarios.”

El envejecimiento es un proceso que depende de varios factores, como la genética, el estilo de vida, la salud, etc., por lo que la edad de partida para comenzar a considerar a alguien como “*persona mayor*” (“*elderly*”, en inglés) es relativa y depende del contexto. En la mayoría de los países desarrollados, incluida la Unión Europea, se establece ese valor de referencia en los 65 años [38].

En este trabajo, nuestro usuario objetivo serán *personas mayores* (a partir de 65 años) y con poca experiencia con la tecnología, y que pretenden realizar tareas o transacciones digitales con un umbral suficiente de conocimiento y de confianza. Estos usuarios suelen tener requisitos o necesidades especiales debidos, generalmente, al deterioro de sus procesos o mecanismos cognitivos como consecuencia de la edad, y que habría que considerar de forma especial en el diseño de la interfaz de usuario.

La siguiente lista muestra los sentidos y habilidades cognitivas típicamente afectadas por el proceso del envejecimiento [51, 163, 162], así como ejemplos de las limitaciones asociadas:

- Visión → baja visión, ceguera.
- Sonido → discapacidad auditiva, sordera
- Tacto → trastorno háptico, paraplejia.
- Lectura → analfabetismo, dislexia.
- Atención → trastorno por déficit de atención, autismo.
- Emoción → ansiedad, depresión, autismo.

Como se verá más adelante, las *interfaces AGILE* se centran en mejorar la interacción relacionada con los canales de la visión, el sonido y el tacto. La edad introduce déficits sensoriales que afectan no solo a los datos sensoriales, sino también a los procesos cognitivos asociados con ellos. Por ejemplo, el tiempo de respuesta y los procesos cognitivos se ven afectados gradual y negativamente con la progresión de la edad. También resulta más difícil mantener enfocada la atención, realizar operaciones mentales, o de aprendizaje. Las habilidades previamente adquiridas o aprendidas, junto con la memoria a largo plazo y el razonamiento, todavía pueden ser bien ejercitados. Sin embargo, las capacidades generalmente más involucradas con tareas desconocidas o que se van a realizar por primera vez, como el razonamiento instantáneo, la memoria de trabajo y el control ejecutivo suelen estar más deterioradas con el aumento de la edad (adaptado de [77]).

5.2. Revisión de la literatura

5.2.1. La inmerecida doble exclusión

La mayoría de los diseños tecnológicos son realizados por personas entre 20 y 50 años, los cuales, en muchos casos, carecen de conocimientos prácticos sobre las necesidades concretas de los grupos de usuarios más jóvenes, o de los de mayor edad, así como del conocimiento del estilo de interacción más adecuado para ellos [28]. Históricamente, los diseñadores han ignorado a los usuarios que no se ajustaban al objetivo previsto, como son aquellos que tienen discapacidades físicas o cognitivas [108]. En respuesta a ese enfoque marginal, varios estudios sobre formas de interacción para grupos de usuarios con

diferentes edades [173, 75] han reportado diferencias y necesidades ocasionadas por la edad. Además, los estilos de interacción ampliamente aceptados [183, 157] y vistos como paradigmas revolucionarios para el diseño de interfaces, como las interfaces de *manipulación directa tipo WIMP*, estaban dejando de lado la consideración de la posible exclusión. Por ejemplo, los usuarios ciegos tenían que “ver” los elementos para interactuar efectivamente con ellos (es decir, para procesos interactivos como la selección de los elementos, el arrastrar y soltar, etc.) [108].

Por otro lado, teniendo en cuenta el hecho de que los jóvenes tienen una tendencia natural a usar las tecnologías y pueden fácilmente adoptar y adaptarse a otras nuevas, la mayoría de las inversiones de marketing se dirigen a involucrar o atraer a aquellos usuarios con altas posibilidades de usar nuevos dispositivos y servicios relacionados. Sin embargo, los usuarios de edad avanzada generalmente no tienen fácil ni el uso ni el acceso a dicha tecnología. Pieper, en 2002, ya dejó bien claro que su concepto de brecha digital se basa en la divergencia entre aquellos usuarios que pueden acceder a la tecnología y aquellas personas, potenciales usuarios, que no pueden [48, 128]. Por ejemplo, en 2008, había más personas mayores de sesenta años que menores de dieciséis en el Reino Unido [75]. Los datos del envejecimiento anticipan la dirección que puede tomar la investigación de los próximos años, ya que muestran que, en 2008, un 7% de la población tenía más de 65 años, porcentaje con expectativas de duplicarse en los próximos 30 años.

5.2.2. La validez de la inclusión

La investigación sobre el *Acceso Universal* a la tecnología se ha centrado principalmente en grupos de *usuarios con necesidades especiales*. Por lo general, estos requisitos especiales están representados por diferentes factores, tales como problemas en el habla, motores, auditivos y visuales, limitaciones cognitivas, problemas emocionales y de aprendizaje, así como en factores ambientales o debidos al envejecimiento [112]. Los campos que estudian el acceso al ordenador para personas con discapacidad o diversidad funcional comenzaron a principios de la década de los 70, cuando los métodos para la reducción de las pulsaciones de teclas eran de especial interés para los ingenieros de la rehabilitación. Desde entonces, ha habido muchos diseños para personas mayores o discapacitadas que han dado lugar a productos de uso común hoy día. Por ejemplo, en 1985, anticipando la dirección del *Diseño Inclusivo*, se demostró que una interfaz predictiva para reducir el esfuerzo de la escritura es beneficiosa para las personas sin discapacidad obvia,

más allá de problemas con la ortografía [164]; las “calculadoras parlantes” diseñadas para ciegos acabaron siendo útiles también para tareas que acaparaban mucho la vista o para condiciones de baja luminosidad; la primera grabadora de cassette fue diseñada específicamente para personas ciegas, pero debido a la mala calidad de sonido, no había muchas expectativas de que acabase siendo un producto tan universal como al final sí acabó siendo [109]; los teléfonos con botones grandes, producidos por British Telecom, fueron un ejemplo de un producto comercialmente exitoso tanto para clientes con discapacidad visual como sin ella; etc. En [136], por ejemplo, se puede consultar una lista mayor de casos similares.

5.2.3. Modelos Inclusivos, modelos de Tareas y modelos Cognitivos

Los *Modelos Inclusivos* son los más recientes de los modelos de usuario existentes en la bibliografía. En términos generales, consideran al usuario como un amplio conjunto de características organizadas en varios niveles: cognitivo, perceptivo y físico. Sus objetivos persiguen la adaptación para el usuario con necesidades especiales y una proyección de inclusividad para el espectro más amplio de usuarios. Por ejemplo, Jacko y Vitense [62] hicieron referencia a habilidades cognitivas, perceptivas y físicas específicas para dibujar una clasificación de las capacidades. Estaban interesados en la degeneración de la retina relacionada con la edad y construyeron un perfil de usuario basado en la información sobre la pérdida de la visión central. Es importante destacar que establecieron algunas pautas para superar y conseguir la adaptación al nuevo estado de visión del usuario. Esas pautas involucraban al tamaño de la fuente, al color del fondo, y a la entrada y salida por voz. Newell y Gregor “rompieron” con el diseño tradicional de la IU y dieron prioridad a las necesidades cognitivas y físicas del usuario no estándar para luego acomodar el diseño convencional en ese proceso, entendiendo como usuario no estándar a aquel con requisitos más allá del erróneamente considerado perfil del usuario promedio (adaptado de [110]). Langdon et al. [78] analizaron los procesos cognitivos desde la perspectiva de los modelos de procesamiento de la información. Sus estudios empíricos señalaron los altos niveles de adaptación que un modelo de usuario tendrá que reunir, y para el cual el usuario tendría que ser necesariamente estudiado en base a un conocimiento holístico de sus capacidades. Hanson [47] revisó la situación de la brecha digital, describiendo la interconectividad entre los adultos mayores, los servicios de red y la tecnología de los ordenadores y plataformas móviles. Otro enfoque interesante es el *Modelo de referencia de acceso universal* (UARM)

que se centra en la accesibilidad de la interacción entre usuarios y sistemas. Su objetivo es descubrir el conocimiento común y las habilidades compartidas entre los usuarios y los sistemas, reduciendo sus desventajas. Se utilizó el *perfil de accesibilidad común* para describir las discapacidades del perfil del usuario [39].

Los *Modelos de Tareas* son valiosos para sintetizar las transacciones en etapas que puedan ser construidas y analizadas. Por ejemplo, *ConcurTaskTrees* (CTT) [103] es una notación para la especificación o modelado de las tareas a realizar con un sistema. Se basa en la descomposición jerárquica de las tareas utilizando una sintaxis gráfica, distingue 4 tipos de tareas, y ofrece operadores temporales que permiten expresar también la concurrencia. Sin embargo, es una notación enfocada más bien al trabajo del analista. Desafortunadamente, el usuario no participa en la construcción de esa solución. La especificación y el tipo de interacción resultante de ese proceso es claro, pero no distingue entre tipos de usuarios ni sus requisitos. Por lo tanto, un usuario novato habrá sido alienado al mismo nivel que un usuario experto. Esto puede traer algunos problemas de rendimiento y experiencia general que no se reflejarían en el modelo, pero sí en la satisfacción y efectividad de la interacción desarrollada.

Con respecto a los *Modelos Cognitivos*, las metodologías GOMS [18] (ya consideradas en el Capítulo 3) han tenido gran influencia sobre éstos. De ellas se toma el concepto de que el objetivo principal es conducir al usuario, paso a paso y de manera progresiva, por una predefinida estructura jerárquica de objetivos. Para lograr estos objetivos de manera satisfactoria, las tareas y acciones deben llevarse a cabo de la misma manera que son descritas. Los mecanismos a seguir para conseguirlas deberían ser simples y consistentes en todo momento de uso de la interfaz de usuario.

Y lógicamente, también hay estándares con influencias importantes en el presente trabajo, como las WCAG 2.1 [171] y el ISO 9241-171 [60].

5.2.4. ¿Por qué los modelos principales no cubren las necesidades de los *novice elderly users*?

El diseño de la IU ha empleado, tradicionalmente, dos factores clave para dar forma al usuario sin discapacidad: la *experiencia* y el *aprendizaje*. Generalmente, se asume para estos parámetros un nivel tal que hace que no se considere necesario dar explicaciones

sobre cómo usar la interfaz, ya que podría consumir tiempo y molestar al usuario experto en el uso futuro de la interfaz. Sin embargo, hoy día, ya son más comunes las aplicaciones que incluyen tutoriales o demostraciones (cuya visualización puede ser voluntaria u obligatoria) que muestran o enseñan, en mayor o menor medida, cómo usar la interfaz. En cualquier caso, el número de tales aplicaciones todavía es reducido, especialmente para el caso de usuarios sin experiencia o con necesidades especiales, como podrían ser las personas mayores.

Para construir una forma de interacción adecuada para nuestros usuarios, es importante comprender la relación entre el usuario, su experiencia y el aprendizaje respecto a la tecnología. Para esto, necesitamos estudiar por qué los modelos tradicionales no consideran a estos “usuarios especiales”. A continuación, vamos a analizar, para unas condiciones cognitivas normales, cómo la experiencia con la misma o con otra tecnología análoga afecta al conocimiento que adquiere un usuario sobre ella. La Figura 5.1 muestra esta influencia en dos gráficas, donde la de la izquierda representa la experiencia y conocimiento acumulada sobre el sistema y su interfaz, con el uso continuado, mientras que el de la derecha muestra la progresión de dicho conocimiento. En esta segunda gráfica (derecha) se puede observar cómo la progresión del conocimiento sobre el sistema comienza desde un máximo y disminuye a medida que crece el uso del sistema (Figura 5.1, región A). Esta situación permanece hasta que se alcanza un punto específico (Figura 5.1, punto 1), después del cual, la progresión del conocimiento se estabiliza. Este punto representa el momento en que el usuario alcanza un conocimiento suficiente de la aplicación, de manera que el uso posterior del sistema apenas le proporcionará conocimiento sobre él, en comparación con las sesiones anteriores (Figura 5.1, región B). Esta región terminará en el punto en que más sesiones no proporcionarán ningún nuevo conocimiento sobre el sistema (Figura 5.1, punto 2).

El *usuario promedio* discutido en la apartado 2.2.1 se ubicaría hipotéticamente en el centro de la curva (*región B*), representando al conjunto de usuarios con una experiencia y conocimiento promedio del sistema. La *región B* descrita delimita otros dos conjuntos de usuarios fuera de la región central de la distribución (*región A* y *región C*). Desafortunadamente (tanto por las razones indicadas en la sección 5.1.3 como por las que se expondrán en siguientes apartados), el *novice elderly user* tiene muy difícil encajar en las últimas *regiones (B o C)*. El punto clave es determinar cómo guiar al usuario sobre el

uso correcto de la interfaz, aumentando su motivación a través de un enfoque didáctico que le enseñe cómo usarlo, desde el primer momento y también en los siguientes (región A).

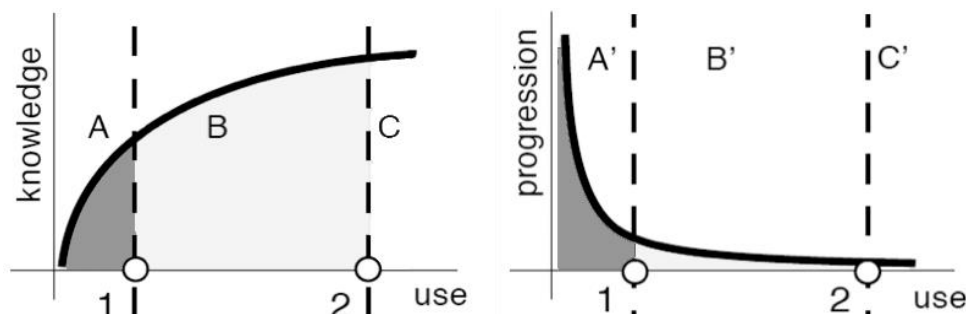


Figura 5.1 Gráficas que muestran, para unas condiciones cognitivas normales, cómo suele evolucionar el conocimiento de un sistema y de su interfaz con el uso (la de la izquierda), y la progresión de dicho conocimiento (gráfica de la derecha).

Esta idea de que el conocimiento se asocia a la experiencia previa y a los usos futuros esperados de la misma interfaz suele explicar (como ocurría en el caso del *usuario ocasional*), la falta de éxito con interfaces de usuario *de un solo uso*, o con cualesquiera con las que el usuario no está familiarizado. A menudo, el diseño de las mismas se basa en la suposición de que las futuras sesiones con la misma interfaz proporcionarán el conocimiento suficiente para mejorar la interacción. El proceso de exploración, basado en la *prueba y error*, además de no ser óptimo, falla cuando hablamos de usuarios sin experiencia o usuarios que no están familiarizados con la interfaz a la que se enfrentan. Esta es la razón por la cual existe un claro desajuste entre los modelos aplicados al *usuario promedio*, y los que se aplican a *usuarios mayores* o *con necesidades especiales*. El resultado es que estos últimos suelen percibir que la interacción no resulta adecuada para ellos, y muchas veces sienten miedo de usar la tecnología o se ven a sí mismos como usuarios incompetentes.

5.3. Fundamentos de la *Interacción AGILE*

5.3.1. Importancia del entorno

El trabajo presentado en este capítulo está principalmente enfocado a usuarios con limitaciones específicas ocasionadas por la edad, que dan lugar a necesidades, que se presentan también cuando interactúan con la tecnología. Estas interacciones ocurren en escenarios específicos y son tan importantes como la tecnología misma. El entorno debe

considerarse como otro elemento clave en la ecuación del diseño. Citando a Whiteside et al. [173]:

“[...] There is nothing absolute about user characteristics; they are only meaningful within a context. Removing individuals from the context destroys the meaning of the characteristics used to classify them.”

Para ilustrar la consecuencia del desajuste entre las necesidades y las habilidades del usuario respecto al entorno, analizaremos la evolución de la definición de “*discapacidad*”. En 1980, la Organización Mundial de la Salud la definía como una “limitación en el funcionamiento de un individuo al realizar una actividad concreta, o al desarrollar la participación en la sociedad con carácter general”. Según esta definición, una deficiencia, producida a su vez por una enfermedad, produce una discapacidad. Sin embargo, en 2001, la OMS rectificaba dicha definición añadiendo que “La discapacidad está caracterizada por el resultado de una compleja interacción entre el estado de salud de la persona y los factores personales y ambientales (contexto), que representan las circunstancias en las que vive la persona” [175]. Por tanto, la desventaja que puede tener un individuo debido a algún impedimento o discapacidad, ahora se ve desde la perspectiva de la adaptación y la interacción con el contexto o entorno de la persona [86]. Así, distintos ambientes o contextos (con o sin barreras) pueden tener impactos diferentes en el desempeño realizado por el mismo individuo. Por ejemplo, una persona podría tener alguna discapacidad, pero al mismo tiempo evitar el posible hándicap: un usuario de silla de ruedas podría evitar el problema ocasionado por la discapacidad si el entorno propicia la desaparición de tal impedimento, por ejemplo, mediante el uso de bordillos rebajados para permitir un fácil acceso de la acera a la calle (adaptado de [113]). Por lo tanto, se encuentra una fuerte correlación entre la medida en que el entorno se adapta a las necesidades del usuario y el hecho de que pueda superar el obstáculo o hándicap. Hay un traslado de la responsabilidad sobre el usuario con necesidades especiales hacia el entorno y sus diseñadores, reconociendo la necesidad de adaptación. No nos referimos a limitaciones o capacidades de forma aislada; es un enfoque holístico que debe centrarse en cómo el entorno puede ayudar al usuario a conseguir sus objetivos, incluida la dependencia entre sus condiciones y el entorno. Por lo tanto, es el entorno, no la persona, lo que se ve *discapacitado*, señalando la importancia de la interacción entre el individuo y el entorno [131].

5.3.2. La interfaz como entorno del usuario

En este apartado vamos a trasladar lo anterior al contexto que nos ocupa. Un problema al que normalmente se enfrenta el usuario, especialmente nuestro usuario objetivo, al interactuar con las interfaces de los sistemas que usa cotidianamente, es que el éxito de dichas interacciones depende, en muy gran medida, de cuán bien (o mal) diseñada estén dichas interfaces. Para describir mejor esta situación, utilizaremos algunos calificadores o conceptos incluidos en la *International Classification of Functioning, Disability and Health* (conocida como *ICF*) de la *Organización Mundial de la Salud* [174]:

- *Capacidad*: describe la habilidad de un individuo para ejecutar una tarea o acción, sin asistencia personal ni uso de dispositivos de ayuda;
- *Rendimiento (Performance)*: describe lo que consigue un individuo en su entorno actual, incluyendo asistencia personal o el uso de dispositivos de ayuda;
- *Factores del entorno*: establecen la distinción entre “barreras” y “facilitadores”, así como la medida en que un factor del entorno actúa de una forma u otra [63].

De este modo, y según lo incluido en el apartado anterior, puede existir un *rendimiento* diferente para la misma *capacidad*, dependiendo del nivel de adaptación de cada entorno en que se encuentre el usuario. En la Figura 5.2, podemos observar una situación común en el uso de la interfaz. En la figura de la izquierda (Figura 5.2a), el *rendimiento* del usuario depende de cada diseño de la interfaz, que no tiene por qué ser homogéneo y que, junto a otros *factores del entorno* involucrados, puede provocar que la misma transacción sea exitosa en algunos casos y que falle en otros. Por el contrario, la *metodología AGILE* que aquí presentamos persigue el objetivo de aumentar el *rendimiento* del usuario gracias a una interfaz guiada y autoexplicativa, basada en la consistencia y permanencia del estilo de la interfaz y de los elementos del diseño para las distintas interfaces, en consonancia con el entorno (véase la Figura 5.2b).

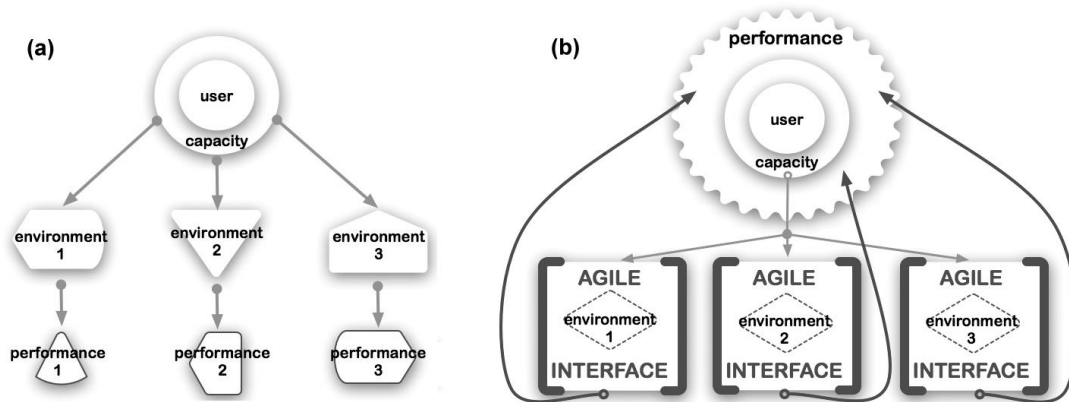


Figura 5.2 Enfoques del diseño de la IU que ignoran los factores ambientales, dando lugar a rendimientos diferentes para la misma capacidad del usuario. (b) Enfoque de interfaz AGILE, que encapsula el factor ambiental para ofrecer un rendimiento unificado y aumentado para el mismo usuario y capacidad.

5.3.3. Adaptación de la interfaz para el *novice elderly user*

Tradicionalmente, las interfaces adaptativas se han centrado en una serie de elementos seleccionados, sobre los que implementar el ajuste de la interfaz durante su uso. Brusilovsky and Millán [14] revisaron las cinco características más importantes a tener en cuenta en los sistemas web adaptativos, así como la forma de modelar el contexto de trabajo del usuario. A continuación, describiremos dichas características, su relación con los *novice elderly users*, y su aplicación, como punto de partida, para la construcción de nuestra propuesta de interacción para ellos:

- En ese artículo, los autores presentaron el *Conocimiento (Knowledge)* como la comprensión del tema que se enseña o del dominio representado por el sistema. Es una característica que cambia de una sesión a otra, o incluso en la misma sesión, en función del aprendizaje (aumenta) o del olvido (disminuye). Esta particular característica, la variación del conocimiento dependiente del proceso de aprendizaje sobre el uso de la interfaz, y dependiente del funcionamiento de la memoria, se debe tener especialmente en cuenta para los usuarios mayores.
- Los *Objetivos (Goals)* representaban lo que el usuario desea conseguir, y fueron catalogados para permitir que el sistema pudiese reconocerlos. En relación con este punto, la *jerarquía de objetivos y tareas* será, como veremos (y al igual que ocurre en la *Interacción Guiada por Objetivos*), algo fundamental en la *interfaz AGILE*.

- El *Background* lo constituye el conjunto de elementos relacionados con la experiencia previa del usuario, pero fuera del dominio central de un sistema web específico. En nuestro caso particular, al *novice elderly user* solo deberemos mostrarle la información mínima necesaria para poder ser entendida y utilizada.
- Dentro de las *Características Individuales (Individual traits)* incluyeron los *estilos Cognitivos y los de Aprendizaje*. Los primeros afectan a la forma en que se organiza y presenta la información; los segundos, a la forma en que las personas prefieren aprender. Ambos componentes también serán considerados en el tipo de interacción que aquí presentamos, pero convenientemente modificados para nuestro usuario destino. Es decir, la presentación y organización de la información se simplificará tanto como sea posible, y los requisitos de aprendizaje serán mínimos.
- El *Contexto de trabajo (Context of work)* también será considerado en las *interfaces AGILE*, tanto en la *dimensión del entorno* como en la *humana*. Respecto a la *dimensión del entorno*, el contexto físico que rodea al usuario juega un papel fundamental durante todo el proceso de interacción, convirtiéndose en parte de la interacción. La *dimensión humana* incluye la importante característica de la carga cognitiva del usuario, y que se deberá tener en cuenta en el proceso de diseño de la interfaz.

Como veremos en el apartado 5.3.6, nosotros realizaremos un proceso de adaptación de la interfaz, pero aplicándolo de forma estática en la etapa de diseño.

5.3.4. Variables que caracterizan al *novice elderly user*

Como ya se ha indicado en capítulos anteriores, la proliferación de nuevos contextos de uso y de nuevas tecnologías está demandando la redefinición de las relaciones entre las máquinas y sus usuarios. Las transacciones ya no son solo para la extracción de datos, sino para objetivos muy diversos y contextos muy variados. Los *novice elderly users* son personas mayores sin experiencia con las Tecnologías de la Información y las Comunicaciones (TICs). Suelen tener incluso menos motivación por aprender que los *one-time users* (que solo usarán una interfaz una única vez). La evidencia anecdótica sugiere

que muchos de ellos opinarían algo en la línea: “*Soy demasiado mayor para aprender nuevas habilidades*” [109].

El coste del aprendizaje aumenta, porque el tiempo que necesitan para adquirir nuevas habilidades es más alto, en comparación con la población más joven, cuyas habilidades cognitivas aún no están degradadas. En consecuencia, dicho costo y la falta de motivación por aprender es el primer obstáculo serio que encuentran estos usuarios para tener éxito en el uso de la misma tecnología en el futuro cercano, tarea que, sin embargo, resulta más fácil y presenta un incentivo común para los usuarios más jóvenes. Por lo tanto, como es posible que estos usuarios no tengan interés en los detalles del software ni del hardware, la motivación por el uso de los mismos quizás solo pueda conseguirse mediante métodos didácticos y guiados. Un proceso de interacción guiada, con ayuda y asistencia en caso de error, e información sobre la consecución de los objetivos forman el núcleo de una interacción apropiada para este tipo de usuarios. Como veremos, incluso si se consiguiese un aprendizaje adecuado sobre el uso de la interfaz, el uso futuro de la misma, si lo hay, prácticamente deberá considerarse como si volviese a ser el “primero”, en términos de la memoria y del recuerdo de lo aprendido.

A continuación, describimos un conjunto de *variables* relevantes y relacionadas con los usuarios que estamos considerando y el tipo de interfaz que implementaremos:

- *Motivación*: razón que desencadena el uso del sistema.
- *Nivel de habilidad*: la capacidad que tiene el usuario con respecto al empleo de la tecnología, en general, o de sistemas informáticos, en particular.
- *Costo del aprendizaje del uso*: el esfuerzo necesario para lograr un nivel suficiente de conocimiento sobre la interfaz y la forma de usarla apropiadamente.
- *Aspectos procedurales del aprendizaje*: los requisitos para que el usuario aprenda los aspectos necesarios de la metodología a seguir para usar satisfactoriamente el sistema.
- *Familiaridad y experiencia*: conocimiento de la interfaz y de sistemas similares.

Además, a esta lista de parámetros, añadiremos también los tres presentados en la sección 5.3.2:

- *Capacidad*: habilidad de un individuo para ejecutar una tarea o acción, sin asistencia personal ni uso de dispositivos de ayuda.
- *Rendimiento (Performance)*: lo que consigue un individuo en su entorno actual, incluyendo asistencia personal o el uso de dispositivos de ayuda.
- *Factores del entorno*: establecen la distinción entre “barreras” y “facilitadores”, así como la medida en que un factor del entorno actúa de una forma u otra.

5.3.5. Valores de las variables que definen al *novice elderly user*

Una vez descritas las variables asociadas a estos usuarios, e involucradas en el diseño de la interfaz AGILE, necesitamos especificar sus valores. Considerando todo lo visto y analizado hasta ahora, para el *novice elderly user*, esos valores son:

- *Motivación*: muy baja o nula
- *Costo del aprendizaje del uso*: alto o muy alto
- *Aspectos procedurales de aprendizaje* = f(implementación interfaz AGILE)
- *Nivel de habilidad*: muy bajo
- *Familiaridad y experiencia con la interfaz*: muy baja o nula
- *Capacidad*: inherente al usuario
- *Rendimiento* = f (capacidad, factores ambientales)
- *Factores del contexto* = f (implementación interfaz AGILE)
+ otros no considerados

Explicemos los valores que tomarían estos parámetros en el caso de nuestro usuario objetivo:

- En primer lugar, el valor de la *motivación* ya se ha argumentado que es muy bajo o nulo, una característica distintiva de los usuarios de edad avanzada. La *motivación* por usar la nueva tecnología será muy baja, a menos que se explique bien el propósito, los beneficios y el aprendizaje del uso.

- El valor del *costo del aprendizaje* será alto o muy alto, al ser, como ya se ha descrito anteriormente, una característica asociada a los usuarios de edad avanzada, debido al deterioro y limitaciones cognitivas debidas a la edad.
- El valor de los *aspectos procedurales del aprendizaje*, es decir, los conocimientos que demanda la interfaz sobre su metodología de uso dependerán de la implementación que se realice de la *interfaz AGILE*.
- Con respecto, tanto al *nivel de habilidad*, como a la *familiaridad o experiencia* previa que este usuario tiene con la interfaz o sistemas análogos, el valor asociado a cada uno será muy bajo, y casi nulo para la última de las variables precisamente por tratarse de usuarios novatos. Sin embargo, incluso si ya han tenido algún encuentro previo con la misma tecnología o similar, el deterioro de la memoria de trabajo, las dificultades para el aprendizaje y, en muchos casos, su falta de motivación, hacen que sea recomendable de cara al diseño considerar que, prácticamente, se enfrentan, en cada nuevo uso, a una interfaz desconocida.
- El valor de la *capacidad* del usuario quedará intrínsecamente determinado por el propio usuario.
- El valor del *rendimiento* estará en función de la *capacidad* del usuario, junto con los factores del *entorno*. En las *interfaces AGILE* desarrolladas para la evaluación que se realizará más adelante (sección 5.5), los factores del entorno que se tendrán en cuenta serán la implementación y la efectividad de la propia interfaz AGILE.

5.3.6. Los objetivos como determinantes del comportamiento del usuario

Newell y Simon [111] describieron los objetivos del usuario como los determinantes de su comportamiento. La interacción del usuario está motivada y dirigida por sus objetivos, aplicando sus conocimientos y limitada por la arquitectura de las tareas y por su propia capacidad cognitiva:

“Él [el usuario] intenta cumplir su objetivo de la forma más fácil posible, dentro de las limitaciones impuestas por la estructura de la tarea, por lo que él sabe, y por sus propios límites de procesamiento de la información.”

Para Moran [101], todas las acciones que realiza un usuario tienen como propósito la consecución de un objetivo. Él enumeró cuatro factores diferentes que determinan y ayudan a predecir el comportamiento del usuario. Estos determinantes pueden modificarse para mejorar el rendimiento del usuario en una situación dada (y presumiblemente, para aumentar su satisfacción y reducir su frustración):

- Modificación de la interfaz del usuario, para cambiar la estructura de tareas;
- Instrucción del usuario, para enseñarlo y aumentar su conocimiento;
- Asistencia en sus limitaciones, para una recuperación eficiente de errores;
- Y automatización de las subtareas, para facilitar la realización de las tareas principales.

También dio un ejemplo que describe dos problemas o limitaciones comunes en el usuario humano:

“Dos ejemplos de la limitación del procesamiento humano son la capacidad limitada de su memoria a corto plazo, y su tendencia a cometer errores de forma ocasional (por una variedad de razones). Estos límites se manifiestan de manera encubierta en la propensión humana a dividir las tareas más grandes en tareas más pequeñas, lo que les exige menos esfuerzo de memorización y tiende a limitar el alcance de los errores.”

A continuación, y al igual que ocurría con los *usuarios ocasionales*, vamos a destacar la *consecución de los objetivos* como un factor determinante en el proceso de diseño de la interfaz para los usuarios que nos ocupan. La importancia de la consecución del objetivo en cada sesión o uso de la interfaz impone restricciones a todos los niveles. En caso de un uso ocasional, como podría ocurrir al comprar un billete en una máquina expendedora en una estación de tren, o en el proceso de escaneo de un pasaporte en el aeropuerto, el conocimiento necesario para realizar la transacción correctamente no puede depender de usos futuros de la interfaz. En cada transacción con la máquina, obtener el billete es un objetivo inmediato, que no puede depender del uso futuro de la interfaz en próximas sesiones.

Los *novice elderly users* deben ser considerados en este aspecto como *usuarios ocasionales*, porque incluso si consiguen aprender a usar el sistema para una tarea frecuente, también pueden olvidarla fácilmente por las razones ya comentadas

anteriormente. En este trabajo argumentaremos que es más probable que los usuarios que nos ocupan tengan éxito en sus transacciones con la interfaz si, como punto de partida, se les aplica las mismas premisas que a un usuario ocasional, debido a que la edad suele acentuar los problemas de aprendizaje, memoria e interactividad.

5.3.7. Implicaciones y recomendaciones para el diseño

La selección de variables y concreción de valores que acabamos de realizar tiene varias implicaciones importantes de cara al diseño de la interfaz y al proceso de interacción.

(a) *Capacidad de aprendizaje*: las interfaces deberán desarrollarse suponiendo que el número de sesiones de uso del sistema, por el mismo usuario, queda limitado a una. Esto significa que no deben tenerse en cuenta posibles interacciones futuras. Los habituales mecanismos de aprendizaje de las funcionalidades y uso de la interfaz, por retención o repetición, son extremadamente limitados para estos usuarios. Mejor que esperar que el usuario aprenda cómo usar el sistema, se recomienda dedicar tiempo y esfuerzo a mostrarle cómo lograr sus objetivos. Una recomendación sería cuidar especialmente las metáforas utilizadas para que establezcan realmente vínculos intuitivos con los elementos del mundo real (ver el trabajo de Carroll and Tomas [26] sobre la metáfora).

(b) *Orientación y Guía*: siempre que sea posible, se deben proporcionar mecanismos eficientes de guía y orientación durante todo el proceso de interacción. Este aspecto es fundamental para tratar de compensar la deficiencia de capacidad de aprendizaje de estos usuarios. Por ejemplo, bien en cada etapa o bien en etapas claves del proceso interactivo, un mapa claro de los pasos ya realizados y/o de las posibilidades del flujo de interacción podría resultar de utilidad, tanto para estos, como para otros usuarios sin experiencia.

(c) *Asistencia*: debería existir un efectivo sistema de ayuda, especialmente para los aspectos más críticos, complejos o novedosos de la interacción. Dicho sistema también debería ayudar al usuario en caso de error o imposibilidad de lograr su objetivo. Este aspecto está muy relacionado con el feedback que se proporciona al usuario, y tendrá influencia directa sobre la idea que se formará el usuario del proceso de interacción y del sistema.

El proceso de adaptación de la interfaz al usuario (introducido en la sección 5.3.3) nosotros lo aplicaremos, de manera estática, en la etapa de diseño. En primer lugar, se deber

realizar una *adaptación de los contenidos*, mostrando en la interfaz solo aquellos elementos que son relevantes en cada momento de la interacción. El propósito es reducir al mínimo las distracciones que puedan provocar una pérdida de tiempo al usuario, como consecuencia de una inspección innecesaria o una interpretación errónea de esos elementos. En segundo lugar, también se adaptará el *sistema de guía* empleado, destinando un área apreciable de la interfaz a la implementación de dicho mecanismo de guía, que será responsable de dirigir la atención del usuario a los elementos interactivos de la interfaz que sean relevantes para cada paso de la transacción. Finalmente, los *mecanismos de asistencia* proporcionarán al usuario los procedimientos necesarios para ayudarlo y permitirle corregir la información o datos introducidos sin esfuerzo.

5.4. El modelo AGILE

5.4.1. Filosofía de interacción

En este capítulo, estamos proponiendo una nueva forma de interacción, derivada de la *Interacción Guiada por Objetivos* [19, 20] (descrita en el Capítulo 3) y del análisis jerárquico de tareas: la *interacción AGILE*. Persigue la facilidad de uso de la interfaz, pero sin apoyarse en mecanismos tradicionales de aprendizaje, como la repetición o la retención. Idealmente, trata de simplificar al máximo el conocimiento y el estudio previo del sistema que necesitan estos usuarios para poder usarlo convenientemente. La semántica debe ser lo suficientemente intuitiva como para permitir al usuario, con un mínimo esfuerzo, interpretar correctamente la estructura de la interfaz. Por ello, dicha interfaz deberá contener, esencialmente, descripciones cortas pero significativas, y un lenguaje y un contenido claro. Un enfoque especialmente adecuado para desarrollar este concepto es el uso correcto y efectivo de la *metáfora*. La metáfora es, hoy día, un pilar del diseño en casi cualquier interfaz. Es, en esencia, un buen vehículo para transferir un concepto, conocimiento o idea al usuario. En el contexto de usuarios con poca o ninguna experiencia, y con la función cognitiva reducida, esta transferencia se vuelve crítica para transmitir, interpretar y reorientar, si es necesario, el significado que representa la metáfora. Como consecuencia, es crucial incluir una nueva dimensión de *literalidad* en la metáfora para lograr un uso correcto de la *interfaz AGILE*. Esta *literalidad* será introducida por dos elementos. Un *asistente virtual* representará la parte encargada de realizar la asistencia y la guía que ofrecerá la interfaz, indicando y explicando las opciones en cada paso. En

definitiva, la literalidad añade el texto adecuado en los lugares adecuados. El segundo elemento será, siempre que sea posible, y acompañando al texto, un icono representativo de cada una de dichas opciones.

5.4.2. Aprendiendo de otros estilos de interacción

Los estilos de interacción tradicionales (descritos en el Capítulo 3) también pueden aportarnos algunas características útiles a tener en cuenta en nuestra propuesta de interacción para el *novice elderly user*.

Como ya se ha visto, la *Manipulación directa* [154, 157, 55] se encuentra, en mayor o menor medida, en casi cualquier interfaz actual. El manejo inmediato de los objetos está relacionado con la forma en que los vemos y usamos en el mundo real, y también se usará en nuestra propuesta de interacción. Características de otros estilos, como los *Formularios*, deben emplearse solo cuando sea absolutamente necesario (por ejemplo, autenticación con introducción de contraseña), e incluso en esos casos, debe examinarse cualquier alternativa con efectividad similar y la mínima introducción de texto. Por ejemplo, la interfaz debe ofrecer mejores formas de manejar la información que el tradicional “tecleo”, siempre que el contexto lo permita (por ejemplo, se puede ahorrar tiempo y esfuerzo al usuario si se considera como ubicación inicial o “por defecto” aquella en la que el usuario se encuentre en ese momento, y si se ofrece una búsqueda predictiva en base a los posibles destinos de su viaje). La interacción basada en líneas de comandos queda descartada. En cambio, el componente activo de las *interfaces basadas en Preguntas y respuestas* [139, 114] y el enfoque pedagógico de los *Wizards* [98, 10, 15, 165, 176] los hacen muy recomendables para las personas mayores.

Sin embargo, y dadas las necesidades y aspectos en común de los *novice elderly users* con los *usuarios ocasionales*, la forma de interacción que se puede considerar la principal referencia y punto de partida para estas *interfaces AGILE* es la *Interacción Guiada por Objetivos* [19, 20], de la que hereda gran parte de sus características, filosofía de uso, y estructura de la interfaz.

5.4.3. Metodología

Para ayudar a los diseñadores de las TICs, este trabajo está orientado a prevalecer sobre las limitaciones funcionales [56] de los usuarios. Esta perspectiva incorpora al diseño de la

interfaz lo que realmente importa para nuestros usuarios: la capacidad de realizar una tarea concreta de forma guiada y asistida. En función de las variables descritas anteriormente (sección 5.3) y teniendo en cuenta sus implicaciones para el diseño, los objetivos específicos de la metodología a seguir son los siguientes:

- (1) Simplificar la toma de decisiones en cada paso de la transacción.
- (2) Guiar al usuario en todos los pasos del proceso de interacción.
- (3) Asistir y demostrar cómo usar los componentes interactivos de la interfaz.

Para *simplificar la toma de decisiones*, proponemos realizar una transformación del árbol de tareas, acortando su profundidad para transformar la transacción en un proceso con mayor número de pasos con decisiones, pero reduciendo la complejidad de cada paso. Numerosas veces, en transacciones digitales, los pasos se sobrecargan con múltiples decisiones o demasiada información, sobrepasando a menudo la capacidad de perceptiva del usuario, especialmente en los primeros usos.

Para conseguir los objetivos 2 y 3, se proponen formas innovadoras de asistencia y guía. En primer lugar, un *asistente virtual* implementa la personificación del concepto de guía. Utilizando principios de animación, esta figura virtual puede hacer demostraciones rápidas y efectivas sobre el “qué hacer” y el “cómo hacerlo”. Además, mediante movimientos de cabeza, señales con la mirada o gestos, dicho asistente debe dirigir la atención del usuario hacia los elementos o áreas relevantes de la interfaz en cada momento. Este paradigma se basa en la técnica de realización de *señales físicas animadas multi-frame* utilizadas para orientar la atención del usuario en las interfaces digitales y táctiles [92], en combinación con los principios de animación para predecir la percepción del observador [93].

En la siguiente lista, que tiene como propósito ofrecer soluciones para la mayoría de los déficits (indicados a la izquierda), se establecen algunas recomendaciones (a la derecha) como recursos para el diseño de la interfaz, y que se deberán utilizar con discreción:

- Atención → diseño simple, señales físicas [92, 93], principios de animación [158]...
- Tacto → pantalla táctil, tamaño de elementos personalizable, redundancia de dispositivos de entrada (mouse, trackpad, llaves, teléfono, etc.)...

- Lectura → evitar botones con texto poco acertado, usar iconos,...
- Visión → principios de animación: exageración, silueta...
- Sonido → texto y sonido, indicaciones, flashes,...
- Emoción → personajes simples y exagerados.

Una vez identificados estos principios y recomendaciones, vamos a mostrar cómo se emplean algunas de ellas en el diseño de las *interfaces AGILE*.

5.4.4. Descripción del prototipo AGILE inicial

Nuestra investigación sobre la interacción AGILE comenzó con el desarrollo de un prototipo inicial de interfaz, pensado para ser usado en modo de pantalla horizontal (landscape), y que, como veremos, guardaba bastante similitud con la estructura y elementos de una *interfaz IGO*, y dividía la pantalla en cuatro áreas de interés (Figura 5.3):

- **Área de objetivos:** esta zona se corresponde con la parte superior de la pantalla. Tiene como propósito mostrar la jerarquía de objetivos activos en cada momento. Comienza mostrando el objetivo inicial y principal a alcanzar (llamémosle objetivo A) seguido por la secuencia de subobjetivos (por ejemplo, objetivo B, objetivo C, etc.) necesarios para conseguir el primero (el objetivo A). Cada vez que un objetivo necesita que se alcance un nuevo subobjetivo, la lista de objetivos mostrados se actualizará, añadiendo ese nuevo subobjetivo. Esta área no está destinada a ofrecer ningún tipo de interacción motora. Por lo tanto, su papel es meramente informativo.

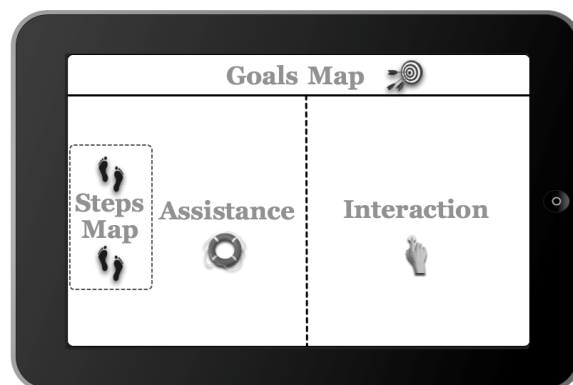


Figura 5.3 Estructura del prototipo inicial de *interfaz AGILE*, aplicado a una tableta (*IPad*). Consiste, de izquierda a derecha y de arriba a abajo, en un *mapa de objetivos*, un *mapa de pasos*, un *área de asistencia* y un *área de interacción*.

- **Área de asistencia:** Esta área incluye la zona en que se muestra la *secuencia* o *mapa de pasos* necesarios para lograr el objetivo actual. Los pasos estarán ordenados, secuenciados y numerados. Sin embargo, solo estará activo un paso en cada momento, el *paso actual* (indicado, en este caso, por el tipo de letra en negrita); el resto de *pasos* aparecerán desactivados (indicado por un tipo de letra sin negrita).

Además, un *asistente virtual* (caracterizado por un personaje femenino en este prototipo) será el encargado de señalar cuál es el paso actual. Otro de los objetivos principales de este *asistente virtual* será el de guiar y ayudar al usuario en el uso de los componentes interactivos de la interfaz. También servirá de vehículo para orientar la atención del usuario hacia las áreas relevantes de la interfaz, y en las que tendrán lugar las demostraciones o la interactividad principal.

Por tanto, esta *área de asistencia* (aparte de las interacciones de carácter cognitivo y perceptual que debe realizar el usuario para consultarla) sólo admitirá un único tipo de acciones interactivas (motoras) del usuario: una interacción de tipo *Point and Click*, que servirá para confirmar, cuando sea necesario, la finalización (realización) del paso actual.

- **Área de interacción:** es el área destinada a recibir la mayor parte de las interacciones del usuario. En ella se irá materializando los resultados del trabajo realizado, mostrando la información relativa al flujo principal de interacción. La selección de elementos, el rellenado de formularios y la manipulación directa ocurrirán en esta área.

5.5. Evaluación de la *interfaz AGILE*

5.5.1. Metodología de evaluación

El proceso de evaluación de la *interfaz AGILE* se distribuyó en tres fases diferentes.

- En primer lugar, se realizó el **desarrollo del prototipo inicial**, siguiendo la estructura ya comentada, con los objetivos expuestos previamente en la Metodología (sección 5.4.3): guiar continuamente al usuario, simplificar su toma de decisiones, y asistirlo y demostrarle cómo realizar el uso de los elementos de la interfaz.

- En segundo lugar, se realizó una **prueba del prototipo inicial** para realizar una transacción digital que nuestros usuarios objetivo realizarían principalmente de forma ocasional. El dispositivo elegido fue un dispositivo portátil (tableta) y el canal de entrada fue el tacto. Dichos usuarios objetivo eran los *novice elderly users*, con poca o ninguna experiencia con dispositivos táctiles, y con poca o ninguna experiencia con ordenadores u otros dispositivos tecnológicos similares.
- Y, por último, se realizó una **segunda iteración del proceso de diseño y prueba** de la interfaz, modificando el prototipo inicial, en base al análisis y a las conclusiones obtenidas de la evaluación anterior.

5.5.2. Iteración 1: desarrollo y evaluación del prototipo inicial

El prototipo inicial, implementado para la primera iteración del proceso de diseño y evaluación, consistió en un *mockup* interactivo de alta fidelidad. El objetivo era realizar un primer estudio y análisis sobre la estructura (de 4 partes) propuesta inicialmente para la *interfaz AGILE* [23]. En esta evaluación preliminar se utilizó una presentación de MS Power Point (MS Power Point 2008 para Mac, v. 12.1) relativa a una aplicación para el diseño de cocinas (ver Figura 5.4), inspirada en la aplicación para la planificación de cocinas ofrecida por la compañía IKEA [58]. Los controles se configuraron para avanzar a la siguiente diapositiva cada vez que el usuario seleccionaba una opción disponible. El número total de pasos (diapositivas) presentados fue 8. Cuando el usuario se quedaba claramente "atascado" en cualquier paso, el facilitador de la prueba lo ayudaba, explicándole cuidadosamente cómo realizar la acción correspondiente. Por lo general, el problema más común fue el de saber cómo hacer una selección (presionar un botón específico), cuando había más de un elemento activo en la pantalla (en la *zona de interacción*).

Intervinieron cinco participantes, con una edad promedio de 74 años. No presentaban ningún tipo de discapacidad aparente y sus habilidades cognitivas eran las típicas de la edad, sin que se describiesen ni detectasen deficiencias especiales. A todos los participantes se les dio las mismas instrucciones para diseñar y amueblar una cocina, utilizando la aplicación, y aclarándoles que el facilitador de la prueba sólo les ayudaría si quedaban bloqueados cierto tiempo en el mismo paso u operación. También se les insistió en que no

se preocupasen si esto ocurría, porque el propósito principal del estudio era detectar si eso pasaba y con qué elementos.

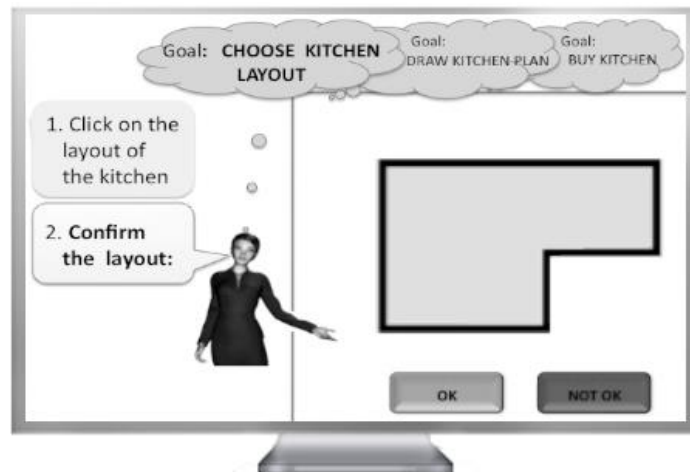


Figura 5.4 Primer prototipo de *interfaz AGILE*, mostrado en un PC con MS Power Point. Se puede apreciar, de izquierda a derecha y de arriba a abajo, nubes con los objetivos iniciados (*mapa de objetivos*); globos de texto explicando cada paso (*mapa de pasos*); un asistente virtual (*área de asistencia*); y el *área de trabajo o de interacción*.

Las conclusiones de esta iteración mostraron que el *mapa de objetivos* apenas resultó útil para estos usuarios, sino más bien al contrario. En vez de ayudarles, lo que hacía era perturbar la interacción normal, ya que cuando se les preguntó a los participantes si habían percibido dicha zona o elemento, solo la mitad afirmaron que sí la habían visto, pero que no la encontraron significativa (que no sabían para qué servía, y que más bien los había confundido). El resto ni la percibió. Lo mismo sucedió con el *mapa de pasos*, que pareció ser más bien un impedimento que una ayuda, ya que los participantes pensaban que tenían que hacer clic en varias de las burbujas explicativas correspondientes los pasos mostrados, interrumpiendo o estorbando al proceso “lógico” de interacción, en lugar de resultar clarificador. La tendencia del usuario era a pensar que todo lo que aparecía en esta zona de la pantalla era pulsable y relevante para la tarea actual, en lugar de distinguir lo que era informativo de lo interactivo, y a lo que tenían que prestar atención.

5.5.3. Iteración 2: rediseño, implementación y evaluación

Sobre la base de la iteración anterior se realizó un rediseño de la estructura inicialmente pensada para la interfaz. La nueva estructura se reflejó en el diseño de una nueva

transacción digital (otra aplicación), empleando de nuevo un dispositivo portátil y táctil (una tableta *iPad2*), y siendo implementadas para el sistema operativo *iOS* (Figura 5.5). Después, se volvió a realizar la evaluación de este *prototipo AGILE evolucionado*.



Figura 5.5 Ejemplo de pantalla correspondiente a la segunda iteración del prototipo de *interfaz AGILE* para la compra de billetes de tren, mostrado en un *iPad2*. Tenga en cuenta la ausencia de los mapas de objetivos y de pasos. En cambio, permanece la figura del asistente virtual (área de asistencia) y la zona de trabajo (área interactiva).

Proceso de evaluación comparativa con otras transacciones digitales

Para la evaluación del nuevo prototipo, se utilizó, como técnica principal, el *test retrospectivo* ([179]), grabando en vídeo la sesión y revisándola posteriormente con el usuario. Ahora el objetivo era comparar la realización de la misma transacción o tarea (en este caso, la compra de billetes de tren) usando por un lado una *interfaz AGILE*, y por otro, usando una *interfaz tradicional de manipulación directa* ofrecida por el sitio web de una compañía ferroviaria (Figura 5.6). En ambos casos se empleó el mismo dispositivo tipo tableta (*iPad2*).

Cada usuario tenía que realizar la misma tarea con cada interfaz (la de cada aplicación), siguiendo un diseño equilibrado (*counterbalanced design*). Esta evaluación se realizó en Escocia (Reino Unido), en el propio hogar de los participantes, y en Málaga (España), en un centro de salud y en un centro de aprendizaje para adultos.

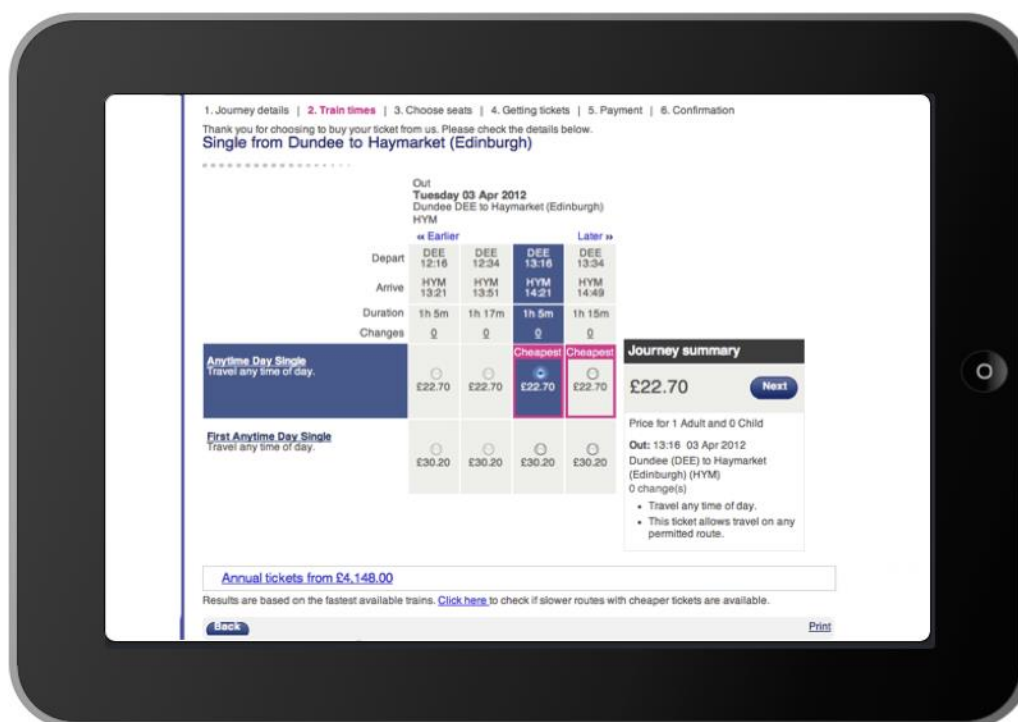


Figura 5.6 Ejemplo de pantalla correspondiente al sitio web ferroviario que los participantes tuvieron que utilizar para realizar la compra de billetes de tren. En particular, se corresponde con la pantalla en que el usuario tenía que seleccionar el horario, el tipo de billete y el precio.

En total, el número de participantes evaluados fue de 11, con un rango de edad entre 58 y 83 años (la edad promedio en Escocia fue de 77 años; en España de 69.58 años; y la global, de 70.93 años). Ninguno tenía discapacidad reconocida, sino tan solo los trastornos cognitivos e impedimentos físicos típicos de esas edades. Todos los participantes escoceses dieron su consentimiento informado bajo las regulaciones de la *School of Health and Social Sciences* de la Universidad de *Abertay Dundee* en inglés, y traducidas al castellano para los participantes españoles.

Para que la prueba fuese más adecuada, la ausencia de traducción al español del sitio web ferroviario nos llevó a realizar la selección de otro sitio web, en este caso, el de una conocida web de comercio electrónico (Amazon, Figura 5.7), y la transacción cambió a una compra de libros. El ajuste en la *versión AGILE* para la nueva transacción (en España) se realizó minimizando las diferencias con la versión desarrollada para la compra de billetes de tren (en Escocia). Esto dio como resultado pantallas y tareas con características similares, pero con una reducción en el número de pasos debido a las propiedades específicas de la transacción para la compra de libros.



Figura 5.7 Pantalla correspondiente al sitio web utilizado para la transacción de compra de libros. En particular, cuando el título del libro ya había sido encontrado y había que indicar la cantidad de ejemplares deseados.

Para los tests de usuario se utilizó una cámara de grabación (Panasonic SDR-H85 con trípode) y un dispositivo de Eyetracking (SMI Eye Tracking Glasses [148, 149]: non-invasive video based glasses-type eye tracker, 30 HZ binocular, spatial resolution 0.1°, gaze position accuracy 0.5° over all distances, 3-point calibration, HD scene camera and audio recording).

Todas las transacciones se realizaron en la misma tableta (*iPad2* con *iOS* versión 4.3.5, con la interfaz diseñada usando el lenguaje *Objective-C* en *XCODE* 4.2, *SDK* 5.0) en una sala controlada, con ausencia de ruido, perturbaciones o cualquier otra interrupción potencial. Las instrucciones dadas para realizar las transacciones fueron las mismas (distinguiendo entre el diseño y los datos correspondientes a la aplicación para la compra de billetes de tren, y el diseño y los datos para la de la compra de libros) y se proporcionaron en una hoja de papel, constantemente visible para el participante en todos los tests. Como se ha indicado, el orden de uso de cada tipo de interfaz (*AGILE* versus *sitio web tradicional*) se realizó de forma alternada (*counterbalanced*) entre los participantes.

La metodología de evaluación utilizada nos permitió obtener tanto datos cuantitativos como cualitativos, para tratar de enriquecer los resultados de la evaluación y poder refinar nuestra propuesta de interacción lo máximo posible. En primer lugar, todo el proceso de interacción se grabó en con una cámara de video, apuntando a la tableta para grabar las interacciones táctiles de los usuarios. Se grabó, para cada participante, tanto la realización de la tarea con ambas interfaces, como las preguntas y respuestas que se surgieron durante el uso de cada interfaz y las que se realizaron después de terminar de usar ambas interfaces. En segundo lugar, y para facilitar el análisis del rendimiento con la tableta, se utilizó el *eye-tracker* anteriormente descrito. Después de terminar las transacciones con los dos sistemas, se entregó a los participantes un cuestionario cualitativo sobre su experiencia en general, cuestiones particulares y recomendaciones sobre ambas aplicaciones.

5.5.4. Resultados de la evaluación comparativa de las interfaces para la compra de billetes de tren, en iPad2

Comparativamente, el análisis de las grabaciones realizadas con la videocámara mostró que el tiempo necesario para completar la transacción fue casi el doble cuando se usó el *sitio web de la compañía ferroviaria*. En cambio, el número de comentarios y de asistencias que tuvo que realizar el facilitador de la prueba fue cuatro veces menor cuando el participante usó el *prototipo AGILE*. Además, el nivel y contenido de dichas asistencias o ayudas tuvo que ser más detallado para el uso del *sitio web ferroviario*, mientras que las intervenciones realizadas con la *versión AGILE* solo consistieron en breves confirmaciones o recordatorios sobre las instrucciones que debían seguir los participantes.

Además de las diferencias cuantitativas, los principales hallazgos fueron cualitativos. Los comentarios y respuestas al cuestionario describían los problemas a los que se enfrentaron los usuarios durante el uso la interfaz del *sitio web ferroviario*. Principalmente, consistieron en saber cómo usar los controles para realizar la introducción de texto (algunos participantes echaban de menos el teclado físico y les costó cierto tiempo darse cuenta de que existía la posibilidad de mostrar uno en la pantalla al seleccionar un cuadro de texto), seleccionar hora y fechas (muy poco espacio para sus dedos), y cómo desplazarse hacia abajo o hacia arriba de la página para encontrar el botón para llevar a cabo el siguiente paso de la transacción. Además, había una pantalla en la que el botón para avanzar al siguiente paso no se mostraba inicialmente, y solo aparecía cuando se seleccionaba un servicio de

tren, lo que aumentaba la confusión sobre "qué hacer a continuación", en ese paso específico.

5.5.5. Resultados de la evaluación comparativa de las interfaces para la compra de libros, en iPad2

Con los tests de usuario realizados en España, asociados a la transacción correspondiente a la compra de libros, se obtuvieron resultados similares a los anteriores, para el *sitio web de la tienda real* y su contrapartida *AGILE*. A pesar de que solo había que pasar por tres pantallas, el tiempo transcurrido al usar la interfaz del sitio web de la compañía fue, nuevamente, mucho mayor (un promedio de 10.1 minutos, $SD = 0.68$, en comparación con el promedio de 3 minutos, $SD = 0.5$, invertido al usar *la interfaz AGILE*). Las quejas principales estaban relacionadas con la sobrecarga de la interfaz del sitio web, al considerar que tenían demasiados elementos innecesarios y no relevantes para la tarea a realizar en cada momento. La publicidad incluida en el sitio, y la exhibición de elementos no necesarios fueron distracciones importantes para los participantes. Argumentaron que tenían que pasar mucho tiempo leyendo mensajes e inspeccionando elementos, con el alto riesgo de hacer clic en ellos y pasar a pantallas no deseadas. Además, solo un participante pudo encontrar la barra de búsqueda sin ayuda. El resto de los participantes tuvieron que ser guiados por el facilitador de la prueba para encontrar con éxito el título del libro.

5.6. Discusión

La evaluación del *prototipo AGILE* (por ejemplo, Figura 5.8) ha proporcionado resultados diferentes e importantes. El primer prototipo mostró que el *mapa de objetivos* es irrelevante para el *novice elderly user*, al menos, cuando se enfrenta a una transacción desconocida. Inicialmente parecía relevante, o al menos, interesante (como sí ocurre con otros grupos de usuarios) mostrar al usuario dónde se encuentra dentro del flujo de la transacción, dentro de la jerarquía de objetivos activa en cada momento. Sin embargo, al menos cuando hay demasiados elementos en pantalla, la prioridad de ese conocimiento no parece ser un aspecto importante del diseño, sino más bien al contrario, una dificultad añadida para estos usuarios.

Algo similar ocurre con el *mapa de pasos* a seguir para alcanzar el objetivo actual. También se mostró completamente irrelevante, incluso más bien un elemento de confusión

para este grupo de usuarios. Muchos de ellos se distrajeron con la visualización del resto de pasos que acompañaban al actual (a pesar de que eran menos llamativos), que era el que realmente debían realizar en ese momento, y semánticamente, no les aportaban ningún significado útil de cara al paso a realizar. Fue una distracción más que una ayuda.

Este prototipo inicial nos dio la oportunidad de refinar la *interfaz* de cara a la siguiente iteración en el proceso de diseño y evaluación, eliminando ambos mapas, tanto el de *Objetivos* y como el de *Pasos*, de nuestra propuesta de estructura para las *interfaces AGILE*.

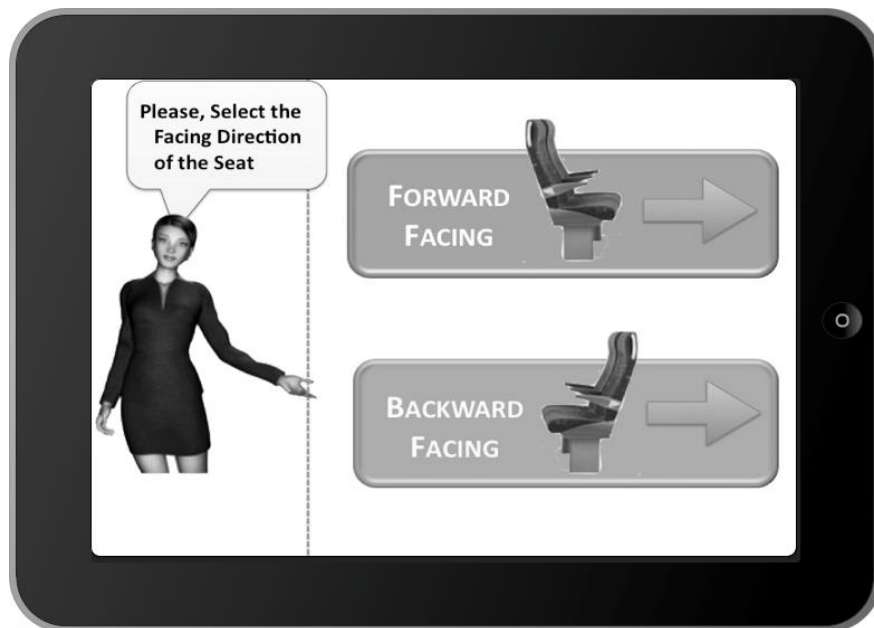


Figura 5.8 Una de las pantallas de la segunda iteración de la *interfaz AGILE* para la transacción para la compra de billetes de tren. En particular, para seleccionar la opción de orientación del asiento.

Con respecto a la evaluación de dichas *interfaces AGILE* para tableta, resultó muy útil la comparación directa entre ellas y las de los *sitios web oficiales* correspondientes que permiten realizar las mismas transacciones (ya sea la compra de un billete de tren o de un libro). El problema del pequeño espacio de la pantalla o de los controles usados se puede resolver parcialmente si el sitio web implementa los mejores controles disponibles para tales operaciones, como los que *Apple Corporation* proporciona para el manejo de horas y fechas en dispositivos iOS (ruedas de desplazamiento). Sin embargo, la forma de usar esos y otros controles “habituales” no es conocida por todos los usuarios, o al menos, no por la mayoría de nuestros *novice elderly users*. Tal es el caso, que cuando se les preguntó a dichos participantes si conocían la existencia de funciones de accesibilidad como el

“Acercar o alejar” usando los dedos (haciendo el gesto del pellizco a la pantalla), muchos de ellos se sorprendieron y argumentaron que “No nació sabiendo eso, nadie me lo enseñó. Ahora es demasiado tarde”. La figura del asistente virtual ayuda a superar este problema. En el prototipo realizado, el agente explica claramente y con buenos modales las instrucciones necesarias para realizar cada paso. Además, las futuras implementaciones de interfaces AGILE deberían incluir la animación de dicho agente, presentándola en coordinación y en secuencia con los elementos relevantes del *área interactiva*. También se podrían proporcionar tutoriales para explicar cómo utilizar los elementos de cada pantalla.

Otra de las conclusiones obtenidas del *user-testing* es relativa al rendimiento de las transacciones. En primera instancia, la transformación de la *interfaz AGILE* para la transacción correspondiente a la compra del billete de tren aumentó el número de pasos (de 7 obligatorios, a 14) (ver Figura 5.9). Sin embargo, las respuestas al cuestionario llevaron a la conclusión de que los usuarios, de hecho, preferían la *interfaz AGILE* en caso de tener que elegir una de las dos versiones para comprarla. A pesar de que se aumentó el número de pasos, el enfoque de la *interfaz AGILE* en general, y la simplicidad de la toma de decisiones en particular, han tenido influencia directa positiva en la satisfacción del usuario. Además, la forma en que los usuarios podían modificar o corregir las decisiones tomadas (paso 11) o, si tenían prisa, ir rápidamente a la última pantalla aceptando los valores predeterminados (paso 6), fueron aprendidas y usadas intuitivamente y sin esfuerzo.

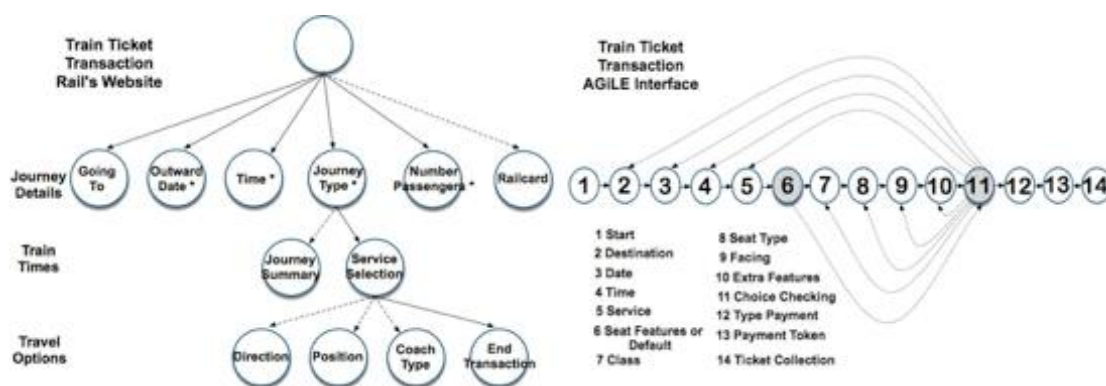


Figura 5.9 (a) Árbol de tareas correspondientes al sitio web ferroviario (con 12 tareas: 7 obligatorias con 5 con valores predeterminados y 5 opcionales). (b) Árbol de tareas correspondiente a la interfaz AGILE, y para la misma transacción digital (con 14 tareas, de las cuales 14 eran obligatorias y ninguna con valor predeterminado). Nótese la posibilidad de avanzar en el paso 6, y de volver atrás desde el paso 11.

A pesar de la clara preferencia de los participantes por la aplicación con *interfaz AGILE* (los 11 participantes la prefirieron; ninguno la otra), se deben mencionar algunas reflexiones críticas sobre dicho tipo de interfaz. Primero, que la versión AGILE no tenía ninguna opción de introducción de texto. La búsqueda estaba predefinida y entre los elementos mostrados se incluían los buscados (la ciudad de destino, para el billete de tren, y el título, para la compra del libro).



Figura 5.10 Una de las pantallas resultantes de la segunda iteración de *la interfaz AGILE* para la transacción para la compra de billetes de tren. En particular, para elegir la hora del viaje.

En segundo lugar, hubo un paso en la *interfaz AGILE* que claramente llevó más tiempo a los participantes realizarlo y aprenderlo. Dicho paso fue el que incluía el uso de un reloj en la aplicación ferroviaria, o un contador del número de artículos, en la aplicación para la compra de libros (Figura 5.10). Algunos participantes se quedaron "atascados" hasta que descubrieron cómo cambiar el valor de dichos elementos, valor que se configuró a cero deliberadamente. Al principio, se pensó que el problema provenía de los números digitales, y que podría mejorarse utilizando otro tipo de números. Sin embargo, los datos del *eye-tracker* mostraron que, por ejemplo (Figura 5.11), un participante estaba atrapado en ese paso, mirando a los números, luego miraba al globo explicativo correspondiente al paso actual y a la figura del asistente, y después volvía a mirar los números, en un bucle claro de no saber *cómo* hacer lo deseado.

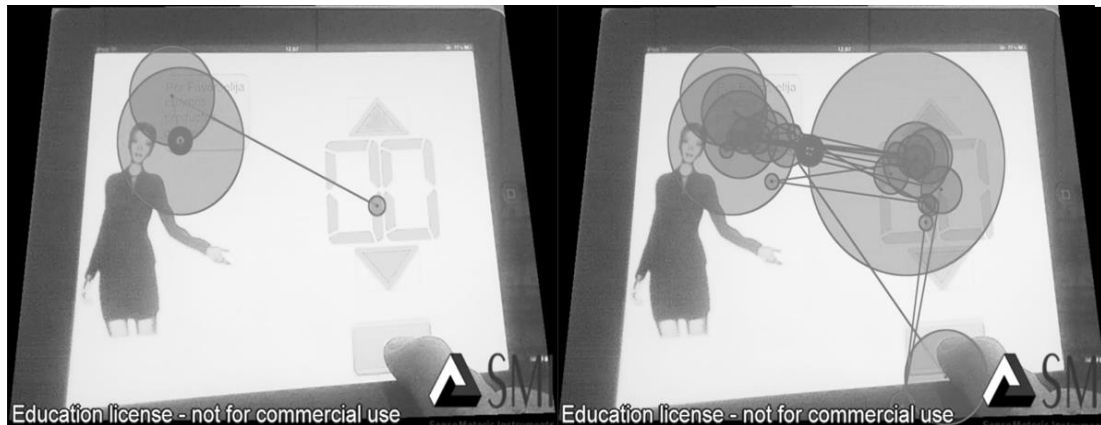


Figura 5.11 Detalle de la interfaz AGILE para la compra de libros. En particular, para indicar el número de libros a comprar. La línea representa el scanpath del participante (74 años), es decir, la trayectoria que la mirada del participante dibuja sobre la grabación de la escena. Los círculos representan las fijaciones del participante, y el tamaño de dichos círculos significa la duración de la fijación. A la izquierda se puede ver cómo, desde el número de elementos, la mirada va a la burbuja informativa (paso a realizar) y a la figura del asistente. En el lado derecho, el ciclo es claramente visible cambiando entre el contador de elementos y la burbuja informativa, con muy pocas líneas dirigidas al botón OK. El participante pulsaba el número, en lugar de las flechas situadas encima y debajo, tratando de aumentar el número de unidades. Sin duda, la falta de información para enseñar al participante cómo operar el control era un error de diseño.

El participante tocó los números tratando de cambiar su valor, y hasta un buen rato después no se dio cuenta de que los triángulos verdes por encima y por debajo de los números incrementaban y decrementaban el valor, respectivamente. Además, la flecha más presionada fue la inferior, probablemente por su proximidad al botón “Aceptar”. Otro problema con las flechas fue que algunos usuarios simplemente mantenían su dedo pulsado sobre la flecha, esperando que el valor aumentara automáticamente mientras lo mantenían presionado. Esa opción se descartó deliberadamente en el proceso de diseño, al pensar que era aún un mecanismo muy reciente y, quizás, demasiado avanzado para los *novice elderly users*. Sin embargo, el cuestionario aclaró la causa de tal comportamiento: algunas videocámaras o microondas ofrecen la opción de mantener presionado el botón para aumentar el valor de forma automática y más rápida, al igual que muchos relojes y despertadores digitales.

Se podría argumentar que mejor que usar un sitio web ejecutado en una tableta, la prueba debería haber ejecutado una versión móvil específica del mismo sitio web. Claramente, esto habría aumentado el rendimiento y la idoneidad de dicha aplicación en dispositivos portátiles. Sin embargo, muchas empresas minoristas y de transporte no tienen aplicaciones separadas, y muchos usuarios nuevos pueden no intentar descargar aplicaciones

específicas, incluso si están disponibles. Por lo tanto, todavía consideramos que la comparación es justa por las siguientes razones.

En el caso de las compañías propietarias de las interfaces webs hubiesen desarrollado sus aplicaciones específicas para la plataforma móvil, las diferencias estarían en la idoneidad de los controles y quizás en algunos cambios en el número y tamaño de los elementos mostrados. En esa situación hipotética, aún se estaría lejos de la solución óptima, ya que los principales problemas detectados con las aplicaciones webs seguirían permaneciendo: estos usuarios seguirían teniendo una carencia de conocimiento sobre cómo usar los controles, como el uso de las ruedas de desplazamiento para establecer la hora y la fecha, o el gesto de pellizcar para acercar y alejar, etc., normalmente desconocidos por la mayoría de estos usuarios (porque nadie ni nada les habría enseñado a usarlos). En este caso, el rendimiento del usuario podría aumentar según su intuición. Por lo tanto, para estos usuarios, la necesidad de asistencia, de explicaciones claras, y de mensajes concisos son esenciales para el uso correcto de la interfaz, especialmente para los que se enfrenten a ella por primera vez y, particularmente, para aquellos sin experiencia con estas tecnologías.

Estos problemas se superaron, en la medida de lo posible, en la segunda iteración de la *interfaz AGILE*. Ahora, el *asistente virtual* emite mensajes claros (en el globo informativo correspondiente el paso actual), y señala la zona relevante de la pantalla donde se debe producir la interacción. El tamaño de los botones, el *look and feel* o apariencia de la interfaz, y las pocas opciones de selección que se intentan ofrecer simultáneamente (2 opciones en la mayoría de los casos, 4 opciones en dos pantallas, y solo una vez con 8 opciones; además del calendario, donde técnicamente el número de opciones sería alrededor de 30, dependiendo del mes) también forman parte este enfoque.

Para superar los problemas pendientes y, en general, tratar de optimizar al máximo su rendimiento y facilidad de uso, una tercera iteración añadiría animación al *asistente virtual*, especialmente a sus brazos, y ofrecería “sobre la marcha” un rápido tutorial para explicar el uso de los controles complejos o probablemente desconocidos. En los dispositivos de tableta, un toque “incorrecto” sobre los números, en vez de pulsaciones sobre los botones de incremento o decremento debería generar un mensaje de advertencia que sugiriese cortésmente que las flechas se pueden usar para generar el número requerido. Además, se debería introducir un orden en la secuencia de presentación de los elementos principales de la interfaz. En primer lugar, aparecería el asistente, después aparecería su burbuja

informativa con la información e instrucciones asociadas al paso a realizar y, finalmente, se mostrarían los elementos del *área interactiva* principal. Por lo tanto, se incrementaría la importancia del *asistente* y del *mensaje con las instrucciones* para el paso actual. Finalmente, la introducción del sonido se implementaría cuidadosamente, atendiendo a las pautas de accesibilidad al respecto [171].

5.7. Conclusiones

Sin afirmar qué tipo de enfoque es mejor para el diseño de la interfaz (1. diseñar para un grupo uniforme de usuarios; 2. diseñar diferentes interfaces para diferentes grupos de usuarios; o 3. diseñar una interfaz adaptativa [3]) hemos realizado una adaptación de la interfaz, pero aplicándola de forma estática en la propia etapa de diseño, y considerando al usuario y sus necesidades como un componente esencial en dicho proceso. Se ha desarrollado y evaluado una forma de interacción, la *interacción AGILE* (Figura 5.12), que se adapta especialmente bien a la idiosincrasia de un usuario no arquetípico, el *novice elderly user*. Este tipo de usuario cumple con algunas peculiaridades, como falta de motivación para aprender, o una reducción de la memoria, incluso en la misma sesión, y todo agravado por la falta de un enfoque didáctico en la mayoría de las interfaces que utilizan o van a utilizar. Con este trabajo hemos cubierto dichos problemas mediante el uso de un eficaz sistema de guía y asistencia para cada paso del proceso de interacción. La asistencia se ha implementado utilizando un *asistente virtual*, encargado de guiar al usuario para la consecución su objetivo actual y de enseñarle cómo realizar operaciones con la interfaz, mediante simples y efectivas señales o animaciones. Dichas señales y gestos físicos deberán orientar la atención del usuario hacia las áreas relevantes de la interfaz. En estas interfaces se hace esencial la simplicidad del diseño, la efectividad de la guía, y el cuidado y conciso uso de la metáfora. La reducción del número de elementos de las pantallas tiene el beneficio de minimizar el riesgo de la *ceguera a los cambios* por parte de estos usuarios de la interfaz (por ejemplo, [138]). Además, la presencia de demasiados elementos de forma simultánea en la pantalla puede hacerles más difícil el segmentar y analizar dichos elementos [145]. Al minimizar el número de ítems y de comparaciones, facilitamos al máximo dichas detecciones y decisiones, respectivamente. Por ejemplo, para mejorar la experiencia táctil [162] y visual, se propone seleccionar iconos claros y de gran tamaño, acompañando a las descripciones textuales (como se recomienda en [60]).



Figura 5.12 Una de las pantallas resultantes de la segunda iteración de la *interfaz AGILE* para la transacción consistente en la compra de billetes de tren. En particular, para elegir las características adicionales del *asiento*.

Como soporte a dicha propuesta, se realizó una investigación basada en *user-testing* transnacional con *novice elderly users*, evaluando la interfaz en un dispositivo móvil y táctil (un *Ipad*), con grabación mediante cámara de video, y uso de *eye-tracker*, realizando un *test retrospectivo*, y complementado con el uso de un *cuestionario* cualitativo. El proceso de evaluación demostró la idoneidad de la *interfaz AGILE* para la realización de transacciones digitales ocasionales, como la adquisición un billete de tren o la compra de libros, por parte de dicho tipo de usuarios.

Entre las características más valiosas que demuestran su idoneidad, destacamos la consistencia, la simplicidad, la claridad, el continuo proceso de guía, y la minimización de errores. Todas estas características podrían adaptarse en el mismo proceso de diseño si varía el grupo de usuarios objetivo. Sería necesaria una tercera iteración de la interfaz para evitar los problemas de efectividad detectados en el uso de ciertos elementos, como el reloj o el contador (en concreto para las operaciones de incremento y decremento asociadas a dichos elementos). También deberían incluirse animaciones de la figura del asistente virtual, tutoriales rápidos para los controles más difíciles o desconocidos, y valorar la secuenciación y el orden de presentación de los elementos.

Terminaremos recalcando que esta *interacción AGILE* está íntimamente relacionada con la *Interacción Guiada por Objetivos* presentada en capítulos anteriores, pudiéndose

considerar una extensión y adaptación de la misma. De ella hereda tanto la filosofía de interacción como la estructura básica de la interfaz, pero estando más enfocada a mejorar la usabilidad y accesibilidad para los *novice elderly users*.

Capítulo 6. Conclusiones y trabajos futuros

Tres categorías de usuarios (novatos, intermedios y expertos) han dominado durante décadas el trabajo de diseñadores, investigadores y educadores de la interfaz de usuario (IU). Estas categorías fueron creadas para conceptualizar las necesidades, estrategias y objetivos de los usuarios alrededor de los años 80. Sin embargo, desde entonces, los grandes avances tecnológicos y de los paradigmas de interacción han generado nuevos contextos de uso y dado acceso a la tecnología a nuevos estereotipos de usuarios que ahora están intentando hacer uso de ella, y que tradicionalmente no lo hacían, entre los que se encuentran personas mayores con requisitos especiales, usuarios con pocos conocimientos informáticos y tecnológicos, o aquellos que necesitan realizar un uso ocasional de un sistema, produciéndose una diversificación de las tradicionales categorías de usuarios.

La investigación realizada con esta tesis se ha centrado, inicialmente, en la última de estas categorías, la del *usuario ocasional*, un usuario cada vez más frecuente y extendido, pero apenas contemplado ni definido, al menos de manera rigurosa, ni en la literatura científica ni académica, hasta ahora. Por ello, se ha realizado un minucioso proceso de búsqueda, revisión y análisis de las distintas categorías de usuarios ya definidas, prestando especial atención a aquellas (infrecuentes, intermitentes, inexpertos, etc.) que comparten características con nuestro usuario en estudio. Los resultados de este trabajo nos han permitido identificar las variables fundamentales que caracterizan y diferencian al UO del resto de categorías. En concreto, los valores de estas variables, *conocimiento* insuficiente *de la interfaz*, y *perspectiva de uso futuro* desconocida, lo distinguen de otros usuarios “non-knowledgeable”, como el *novato* y el *first-time user*, cuya adquisición del conocimiento suficiente de la interfaz se espera conseguir mediante el uso repetido de la misma. Se trata, por tanto, de a una categoría de usuario ortogonal con las ya establecidas, al deberse colocar fuera de la tradicional curva de aprendizaje correspondiente a los *novatos*, *intermedios* y *expertos*.

A partir del análisis de las implicaciones de la definición del UO y de algunas recomendaciones recopiladas de las clasificaciones de usuarios analizadas se han podido establecer unos *principios* fundamentales a tener en cuenta de cara al diseño de las interfaces para estos UO, complementados y avalados por los resultados de los estudios

empíricos realizados. Estos *principios* generales pueden ayudar a la comunidad IPO a mejorar la efectividad, eficiencia y satisfacción en el uso de la interfaz para dichos usuarios: *Guía y asistencia (Guidance and Assistance)*, *Facilidad de aprendizaje (Learnability)*, *Consecución de objetivos (Goalability)*, ajuste del *Tiempo transcurrido (Elapsed Time)*, y *Recuperación y manejo de errores (Recoverability and Error handling)*. Estas pautas reflejan el hecho de que el diseño de la interfaz para el UO no debe esperar ni basarse en siguientes usos del sistema. Requisitos relacionados con la “expertise” de estos usuarios deben excluirse de las condiciones previas al desarrollo de la interfaz. Como el UO utilizará el sistema sin el *conocimiento suficiente de la interfaz*, apenas se le debe exigir conocimiento previo sobre la misma. Y como *la probabilidad de uso futuro* también será desconocida (pero menor que 1), tampoco se debe presuponer la existencia de un periodo de aprendizaje y de familiarización con dicha interfaz.

Al tratarse de *principios* generales no especifican cómo conseguir alcanzarlos. Sin embargo, en los Capítulos 3 y 4 sí que se concreta, implementa y evalúa un estilo de interacción, la *Interacción Guiada por Objetivos*, que aplica de forma concreta estos principios con el objetivo de reducir drásticamente la curva de aprendizaje, si es que la hay. Se trata de una alternativa a las tradicionales *interfaces de manipulación directa*, con un enfoque diferente a la solución habitual para reducir al máximo la curva de aprendizaje, y que resultará especialmente adecuada para estos *usuarios ocasionales*. En vez de añadir elementos extra a la interfaz y de esperar que el usuario comprenda esas metáforas y sepa usar esos elementos, hacemos que sea la propia interfaz la que lo guíe continuamente. En concreto, al considerar la interacción como una tarea de resolución de problemas y al usuario como un sistema procesador de información, podemos describir, tanto la jerarquía de objetivos que se pueden abordar (el “qué hacer”), como las actividades y operaciones concretas a realizar para conseguir dichos objetivos (el “cómo hacerlo”), de forma parecida a como lo hacen las técnicas GOMS. Sin embargo, la idea central y la novedad que aporta este tipo de interacción consiste en que el usuario no tiene que dedicar tiempo a adquirir dicho conocimiento, ni buscarlo en un manual de usuario, ni en ningún otro sistema de ayuda, sino en que está integrado en la propia interfaz, en la denominada *Ventana de Objetivos (VO)*. Este elemento es el encargado de ir proporcionando dicho conocimiento al usuario, a medida que éste va interactuando con el sistema y avanzando por la jerarquía de posibles objetivos. Por tanto, es el que actúa como mecanismo de guía, indicando, continuamente y de manera secuencial y progresiva, la forma de alcanzar el *objetivo* actual.

Y éste, siempre se alcanzará siguiendo los pasos de un *método*, o realizando una *selección* entre un grupo de alternativas, y siempre de acuerdo con el árbol jerárquico de objetivos y tareas resultante de un previo y muy recomendable proceso de análisis y especificación de estas interfaces. Aunque es cierto que el usuario tendrá menos libertad de acción, esta limitación suele ser liberadora para usuarios novatos, inexpertos y, especialmente, para los ocasionales, ya que, como compensación, no necesitan estar familiarizados con los widgets gráficos ni pensar cuál debe ser el siguiente comando o paso a realizar. Y lo más importante para ellos, prácticamente no se les requiere conocimiento previo, reduciéndose drásticamente la curva de aprendizaje, si es que la hay.

Por otro lado, para realizar la importante tarea de especificación o modelado de estas interfaces (especialmente del sistema de guía), nos hemos basado en una de las variantes que existen dentro de la familia de técnicas GOMS: NGOMSL. Sin embargo, para cubrir los distintos aspectos de esta interacción ha sido necesario realizar una adaptación y extensión de la misma. Además, de esta manera, a partir de la especificación o modelado de la interfaz, y tras un fácil proceso de “compilación”, será casi directo obtener el mecanismo de guía, la *Ventana de Objetivos*, de la interfaz.

Posteriormente, y con la finalidad de demostrar que este tipo de interacción guiada cumplía realmente su propósito y poder realizar ajustes si eran necesarios, se ha realizado un estudio empírico comparativo con las actuales y tan extendidas interfaces de *Manipulación Directa* (MD), presentes en casi todo tipo de situaciones y para todo tipo de usuarios, realizando test con usuarios, y recogiendo datos tanto objetivos (tiempos y errores) como subjetivos (valoraciones y preferencias). A pesar de que las aplicaciones empleadas para las pruebas requieren un alto grado de manipulación de objetos y existen en el mercado sólo bajo el estilo de MD, el análisis de los datos resultantes muestra que los UO tienen una clara preferencia por esta interacción guiada (mayor satisfacción), consiguen mejores tiempos (mayor eficiencia), y tienen menos incidencias, errores y confusión (mayor eficacia). Evidentemente, con este trabajo no tratamos de minimizar la gran importancia que las *interfaces de manipulación directa* convencionales tienen, han tenido durante décadas, y seguirán teniendo en el futuro. Lo que mostramos es que un sistema de guía integrado, como el de la IGO, es generalmente una mejor opción para aquellos usuarios que no conocen la interfaz y que no desean o no pueden permitirse una curva de aprendizaje, ofreciendo una alternativa a diseñadores y desarrolladores de sistemas

interactivos que tienen a este tipo de usuarios como objetivo, y especialmente, cuando dichos usuarios tengan poca experiencia con la informática o la tecnología.

Una interesante línea de trabajo futuro podría ser la integración de la IGO dentro de las interfaces adaptativas. En este sentido, las tecnologías recientes de Deep Learning son bastante prometedoras, ya que se podrían utilizar para generar cambios automatizados en la guía, según el perfil y características de los usuarios.

Un beneficio adicional del trabajo realizado sobre esta *Interacción Guiada por Objetivos* y sobre el *Usuario Ocasional* es que es útil y aplicable, al menos como punto de partida, a otros tipos de usuarios con requisitos similares. Por ejemplo, aquellos que se sienten cómodos con la idea de no tener que recordar cómo utilizar la interfaz, o que tengan poca motivación y/o dificultad para aprender a usarla, como puede ocurrir con las personas mayores sin experiencia con las nuevas tecnologías, los *novice elderly users*. Teniéndolos como objetivo principal, se ha desarrollado un estudio similar al anterior, pero enfocado y adaptado a estos usuarios y a dispositivos táctiles.

En concreto, se han analizado en profundidad las características y necesidades de estos *novices elderly users*, se han determinado las variables que los definen, y se ha propuesto, bajo un enfoque inclusivo, una metodología de diseño de un tipo de interfaces especialmente apropiadas para ellos. El resultado es la *interacción AGILE*. Este tipo de usuario cumple con algunas peculiaridades, como falta de motivación para aprender, o una reducción de la memoria, incluso en la misma sesión, y todo agravado por la falta de un enfoque didáctico en la mayoría de las interfaces que utilizan o van a utilizar. Con este trabajo hemos cubierto dichos problemas mediante el uso de un eficaz sistema de guía y asistencia para cada paso del proceso de interacción. En estas interfaces se hace esencial la simplicidad del diseño, asistir y guiar continuamente al usuario, simplificar su toma de decisiones, demostrarle cómo realizar el uso de los elementos de la interfaz, y un cuidado y conciso uso de la metáfora. La asistencia se ha implementado utilizando la figura de un *asistente virtual*, encargado de guiar paso a paso al usuario para la consecución sus objetivos, de enseñarle cómo realizar operaciones con la interfaz, y de orientar su atención hacia las áreas relevantes de la interfaz en cada momento.

Como soporte a dicha propuesta, se realizó una investigación basada en un proceso de *user-testing* transnacional (realizado en Escocia y en España), evaluando la interfaz en un

dispositivo móvil y táctil (*Ipad*), con grabación mediante cámara de video, y uso de *eye-tracker*, realizando un *test retrospectivo*, y complementado con el uso de un *cuestionario* cualitativo. El proceso de evaluación demostró la idoneidad de la *interfaz AGILE* para la realización de transacciones digitales ocasionales, como la adquisición un billete de tren o la compra de libros, para estos *novice elderly users*.

Entre las características más valiosas que demuestran su idoneidad, destacamos la consistencia, la simplicidad, la claridad, el continuo proceso de guía, y la minimización de errores. *Consistencia* gracias a la estructura uniforme de la interfaz. *Simplicidad* en la toma de decisiones sin esfuerzo que los usuarios exhibieron en cada paso, al requerirles una mínima carga cognitiva. *Claridad* al mostrar los elementos estrictamente indispensables para poder realizar la transacción, con botones grandes, tipo y tamaño de fuente legible, y mensajes concisos. *Guía*, gracias a las sucintas instrucciones dadas en cada paso por la figura del asistente virtual, y ubicadas de manera consistente en una amplia y visible región de la pantalla. Y *minimización de errores*, al restringir las posibles opciones que el usuario tiene en cada paso, sin afectar por ello a la efectividad en la consecución del objetivo, y mejorando la satisfacción del usuario. Todas estas características podrían adaptarse en el mismo proceso de diseño si varía el grupo de usuarios objetivo; por ejemplo, maximizando el uso del canal del sonido para usuarios ciegos, o incluir en el sistema de guía una demostración que enseñe al usuario cómo hacer gestos que puede desconocer, etc. Sin embargo, estas revisiones o adaptaciones de la interfaz deben aplicarse en el proceso de diseño, y deben ser cuidadosamente probadas e iteradas.

Por otro lado, sería necesaria una tercera iteración del modelo de *interfaz AGILE* para evitar los problemas de efectividad detectados en el uso de ciertos elementos. También deberían incluirse animaciones de la figura del asistente virtual, tutoriales rápidos para los controles más difíciles o desconocidos, y valorar la secuenciación del orden de presentación de los elementos (aspectos sugeridos al final de la sección 5.6).

Finalmente, y como trabajo futuro, podría realizarse la aplicación y evaluación de este tipo de interacción con otros tipos de tareas o transacciones con mayor variedad y complejidad, al mismo tiempo que presenten nuevos retos para los diseñadores, modificando o actualizando cualquiera de los principios aquí expuestos. Y también se podría probar este estilo de interacción con usuarios con limitaciones más profundas, como

la demencia o el Parkinson, intentando encontrar nuevos y más apropiados elementos interactivos para este tipo de interfaces y este tipo de usuarios.

Anexo I. Especificación NGOMSL-IGO de la *interfaz* *IGO* evaluada en el Capítulo 4.

En este anexo contiene la especificación NGOMSL-IGO completa correspondiente a la versión IGO de la aplicación para el diseño y amueblado de cocinas utilizada en el proceso de evaluación comparativa con usuarios, entre la Interacción Guiada por Objetivos y la Manipulación Directa convencional, descrito en el Capítulo 4.

Method for: **Diseñar cocina**

```
step 1. Accomplish: Dibujar el plano de la cocina...
step 2. Accomplish: Amueblar la cocina...
step 3. Decide: If ¿comenzar de nuevo desde el principio?
           then goto 1
step 4. Accomplish: Encargar la cocina...
step 5. Return with goal accomplished
```

Method for: **Dibujar el plano de la cocina**

```
Cancelable
step 1. Accomplish: Elegir la forma de la cocina...
step 2. Accomplish: Introducir medidas de las paredes...
step 3. Accomplish: Colocar puertas o ventanas...
step 4. Return with goal accomplished
```

Method for: **Elegir la forma de la cocina**

```
step 1. Pinche la forma de la cocina a amueblar
step 2. Decide: If ¿cambiar forma elegida? then goto 1
step 3. Return with goal accomplished
```

For the system > Selection for: **Introducir medidas**

```
a) If (forma == 2_lados) then accomplish:
    Introducir medidas(2 lados)...
b) If (forma == 4_lados) then accomplish:
    Introducir medidas(4 lados)...
...
```

Return with goal accomplished

Method for: **Introducir medidas (2 lados)**

Cancelable

- step 1. Introduzca la medida del lado A
- step 2. Introduzca la medida del lado B
- step 3. Decide: If ¿desea corregir alguna medida? then goto 1
- step 4. Return with goal accomplished

Method for: **Introducir medidas (4 lados)**

Cancelable

- step 1. Introduzca medida del lado A
- step 2. Introduzca medida del lado B
- step 3. Introduzca medida del lado C
- step 4. Introduzca medida del lado D
- step 5. Decide: If ¿desea corregir alguna medida? then goto 1
- step 6. Return with goal accomplished

Method for: **Colocar puertas y ventanas**

Cancelable

- step 1. Accomplish: **Elegir una tarea con puerta o ventana...**
- step 2. Decide: If ¿más operaciones o cambios? then goto 1
- step 3. Terminar operaciones con puertas o ventanas
- step 4. Return with goal accomplished

Selection for: **Elegir una tarea con puerta o ventana**

Cancelable, disable if (num_puertas == 0)

- a) If (poner puerta o ventana) then
accomplish: **Poner puerta o ventana...**
- b) If (quitar puerta o ventana) then
accomplish: **Quitar puerta o ventana...**
disable if (num_puertas == 0)
- c) If (desplazar puerta o ventana) then
accomplish: **Desplazar puerta o ventana...**
disable if (num_puertas == 0)

Return with goal accomplished

Method for: **Poner una puerta o ventana**

Cancelable

- step 1. Pinche el objeto a poner
- step 2. Pinche el lugar donde colocarlo
- step 3. Return with goal accomplished
effect nun_puertas++

Method for: **Quitar una puerta o ventana**

Cancelable

- step 1. Pinche el objeto a quitar
- step 2. Confirmar el borrado (en caso contrario, pinche "cancelar")
- step 3. Return with goal accomplished
effect nun_puertas--

Method for: **Desplazar una puerta o ventana**

Cancelable

- step 1. Pinche el objeto a mover
- step 2. Pinche el nuevo lugar donde colocarlo
- step 3. Return with goal accomplished

```

Method for: Amueblar la cocina
  step 1. Accomplish: Elegir tarea de amueblado...
  step 2. Decide: If ¿seguir amueblando?
                then goto 1
  step 3. Terminar con las tareas de amueblado
  step 4. Accomplish: Elegir elementos comunes...
                disable if (materiales_elegidos == false)
  step 5. Decide: If ¿desea modificar presupuesto? then goto 1
  step 6. Confirmar el presupuesto (pero después ya
                no podrá hacer más cambios)
  step 7. Return with goal accomplished

```

```

Selection for: Elegir tarea de amueblado
Cancelable, disable if (num_muebles == 0)
a) If (poner) then accomplish Poner mueble...
b) If (desplazar) then accomplish Desplazar mueble...
    disable if (num_muebles == 0)
c) If (girar) then accomplish Girar mueble...
    disable if (num_muebles == 0)
d) If (quitar) then accomplish Quitar mueble...
    disable if (num_muebles == 0)
e) If (vaciar cocina) then accomplish Vaciar cocina...
    disable if (num_muebles == 0)
f) If (elegir materiales) then
    accomplish Elegir elementos comunes...
    disable if (num_muebles == 0)
g) If (ver cocina en 3D) then accomplish Ver en 3D...
h) If (consultar presupuesto) then
    accomplish Consultar presupuesto...
    disable if (num_muebles == 0)
Return with goal accomplished

```

```

Method for: Poner mueble
Cancelable
  step 1. Accomplish: Elegir tipo de mueble...
  step 2. Pinche el mueble que desea poner
  step 3. Pinche el lugar donde colocarlo
  step 4. Return with goal accomplished
          effect num_muebles++

```

```

Method for: Desplazar mueble
Cancelable
  step 1. Pinche el mueble a mover
  step 2. Pinche el nuevo lugar donde colocarlo
  step 3. Return with goal accomplished

```

```

Method for: Girar muebles
Cancelable
  step 1. Pinche el mueble que desea girar
  step 2. Girar el mueble 90 grados
  step 3. Decide: If ¿volver a girar?
                then goto 2
  step 4. Terminar de girar
  step 5. Pinche el nuevo lugar del mueble
  step 6. Return with goal accomplished

```

Method for: **Quitar mueble**

Cancelable
step 1. Pinche el mueble que desea quitar
step 2. Confirme el borrado del mueble;
 en caso contrario, pinche "cancelar"
step 3. Return with goal accomplished
 effect num_muebles--

Method for: **Vaciar cocina**

Cancelable
step 1. Confirme que desea quitar todos los muebles;
 en caso contrario, pinche "cancelar"
step 2. Return with goal accomplished
 effect num_muebles = 0

Method for: **Elegir elementos comunes...**

Cancelable
step 1. Pinche la encimera que prefiera
 disable if (no_es_posible_encimera == true)
step 2. Pinche la madera que prefiera
 disable if (no_es_posible_puertas == true)
step 3. Pinche el tipo de tirador que prefiera
 disable if (no_es_posible_tiradores == true)
step 4. Return with goal accomplished

Method for: **Ver en 3D**

step 1. Pinche para girar la cocina
step 2. Decide: If ¿otro giro? then goto 1
step 3. Return with goal accomplished

Method for: **Consultar presupuesto**

step 1. Return with goal accomplished

Selection for: **Elegir tipo de mueble**

Cancelable
a) If (es mueble bajo) then
 effect mostrar_muebles_bajos
b) If (es mueble alto, de pared) then
 effect mostrar_muebles_altos
c) If (es electrodoméstico) then
 effect mostrar_electrodomésticos
d) If (es otro mobiliario) then
 effect mostrar_otro_mobiliario
Return with goal accomplished

Method for: **Realizar pedido**

step 1. Introduzca su nombre
step 2. Decide: If ¿desea corregir su nombre? then goto 1
step 3. Return with goal accomplished

Anexo II. Publicaciones que avalan la Tesis

En este apéndice se relacionan las principales publicaciones derivadas del proceso de desarrollo de esta tesis.

1. **JCR Indexed Journal** (1º Tercil)

Carrillo, A.L., Falgueras, J.A. (2020). Proposal and testing Goals-Guided Interaction for occasional users. *Human-Centric Computing and Information Sciences*, 10, 4. <https://doi.org/10.1186/s13673-020-0209-2>

* **JCR**: **Computer Science**, Information Systems – **Q2** (#40 de 155), fi: **3.212**

2. **JCR Indexed Journal** (Q1)

Carrillo A.L., Martínez, S., Falgueras, J., Scott-Brown, K. (2017). A reflective characterisation of occasional user. *Computers in Human Behavior*, 70, 74–89 <https://doi.org/10.1016/j.chb.2016.12.027>

* **JCR**: Psychology, Multidisciplinary - **Q1** (#16 de 135), fi: **3.536**

Psychology, Experimental - **Q1** (#8 de 85), fi: **3.536**

* **SCImago Journal Rank**:

Computer Science, Human-Computer Interaction - **Q1** (#5), fi: **1.555**

* **Scopus**:

Computer Science, Human-Computer Interaction - **Q1** (#7), fi: **4.57**

3. **CORE A Ranked International Conference**

Carrillo A.L., Falgueras J.A. (2019). GDI as an Alternative Guiding Interaction Style for Occasional Users. In: *Lamas D., Loizides F., Nacke L., Petrie H., Winckler M., Zaphiris P. (eds) Human-Computer Interaction – INTERACT 2019. Lecture Notes in Computer Science, vol. 11749, pp 75–96. Springer.* https://doi.org/10.1007/978-3-030-29390-1_5

4. **Book Chapter** (*Human-Computer Interaction Series*. Springer)

Martínez S., Carrillo A.L., Scott-Brown K., Falgueras J. (2013). AGILE Interface for No-Learning nor Experience Required Interaction. In: *Martín E., Haya P., Carro R. (eds) User Modeling and Adaptation for Daily Routines, pp 119–151. Human-Computer Interaction Series. Springer.* https://doi.org/10.1007/978-1-4471-4778-7_5

5. **International Conference**

Carrillo A.L. Juan Falgueras (2015). Goal Driven Interaction vs. Direct Manipulation, an empirical comparison. In: *Proceedings of the XVI International Conference on Human Computer Interaction (Interacción'15)*. ACM, New York, NY, USA. ISBN 978-1-4503-3463-1 <https://doi.org/10.1145/2829875.2829892>

6. **Internacional Conference**

Carrillo A.L., Martínez, S., Falgueras, J., Scott-brown (2012). Diseño y evaluación de interfaces para la tercera edad. In: *Proceedings of the XIII International Conference on Human Computer Interaction (Interacción'12)*.

Listado de Figuras

- Figura 2.1** Evolución del conocimiento de la interfaz, adquirido por un usuario a lo largo de las repetidas sesiones de uso de la misma interfaz. 14
- Figura 2.2** El Usuario Ocasional se sitúa fuera de la tradicional curva de aprendizaje asociada a la típica clasificación de usuarios de 3 categorías (novatos, intermedios y expertos). El UO queda representado por un punto fuera de dicha curva..... 33
- Figura 2.3** Árbol de decisión para clasificar a los potenciales usuarios como *Usuarios Ocasionales*. 37
- Figura 2.4** Capturas de pantalla correspondientes a las dos interfaces comparadas para la adquisición de un billete de tren por *Usuarios Ocasionales*. (a) A la izquierda, la tradicional interfaz web. (b) A la derecha, la interfaz de usuario equivalente, construida aplicando explícitamente los principios de diseño dados para este tipo de usuario. 44
- Figura 2.5** Capturas de pantalla de las dos interfaces testeadas para el diseño y amueblado de cocinas (durante el proceso de mover o cambiar la posición un mueble): a la izquierda, la interfaz de manipulación directa (MD); a la derecha, la interfaz guiada (IGO). 46
- Figura 3.1** Principales Estilos y Paradigmas de interacción 59
- Figura 3.2** Estructura básica una *interfaz IGO*, en la que se puede apreciar sus tres componentes principales: la *Ventana de Trabajo* (VT), la *Barra con la jerarquía de Objetivos Activos* (BOA), y la *Ventana de Guía* (VG), ofreciendo esta última el *método* que permite alcanzar el *objetivo* actual. 64
- Figura 3.3** Ejemplo en el que el *paso* actual consiste en una *Iniciación de objetivo*. Al pulsar el usuario sobre dicho *paso*, la VO deberá actualizar su contenido, y pasar a mostrar el *método* (o la *selección*) que permita alcanzar el *objetivo* recién iniciado. 65
- Figura 3.4** Ejemplo en el que el *paso* actual consiste en la *Realización de una acción*. En este caso, se trata de una acción a realizar en la VT (sin interacción física directa con la VO) cuya realización puede detectar el sistema, momento en el que éste pasará a habilitar el paso siguiente en la VO. 66
- Figura 3.5** Ejemplo en el que el *paso* actual consiste en la *Realización de una acción*, en este caso, a realizar directamente sobre la propia VO, con la intención de activar un comando del sistema (eliminar un mueble). Para avanzar, el usuario deberá pulsar sobre dicho *paso* en la VO. En este caso, si se realiza el borrado del mueble seleccionado, todavía existiría la opción, ya en el siguiente paso, de cancelar dicha operación (el borrado del mueble). 66

- Figura 3.6** Ejemplo en el que el *paso* actual consiste en un *Salto condicional*. El usuario deberá decidir qué *paso* desea realizar, de los dos que se le ofrecen (continuar por el siguiente, o saltar al paso indicado). 67
- Figura 3.7** Ejemplo en el que se muestran 3 estados distintos de una interfaz IGO. En la imagen central, el *paso* a realizar implicará la *Finalización del método* actual. También se puede observar que se retornaría al siguiente paso del *método* padre (el siguiente *paso* al que desvió la interacción hacia el *método* que acaba de terminar). 68
- Figura 3.8** Dos ejemplos en los que para alcanzar el *objetivo* actual se debe realizar una *selección* (en la VO). En el primer caso (el superior), no todas las *alternativas* están habilitadas en ese momento. 68
- Figura 3.9** Ejemplo en el que el *objetivo* actual no se alcanza realizando la típica *selección* IGO (en la Ventana de Objetivos), sino siguiendo los *pasos* de un *método*, y dónde el *paso* actual implica realizar la elección de una de las diversas opciones presentadas en la VT. 69
- Figura 3.10** Ejemplo que muestra una interfaz IGO en 3 momentos distintos de uso de un sistema (los mismos que en la *Figura 3.6*). En la imagen central, la VO está ofreciendo el *método* que permite alcanzar un *objetivo cancelable*. En esta ocasión se puede apreciar que, si el usuario no realiza el *paso* actual, sino que elige *cancelar* el *objetivo*, se retornaría al mismo punto y al mismo estado del *objetivo* padre (que desvió la interacción hacia el *método* que acaba de cancelar). 70
- Figura 4.1** Capturas de pantalla de las interfaces testeadas, relacionadas con el proceso de introducción de las medidas de las paredes de la cocina: a la izquierda, la *interfaz de MD*; a la derecha, la *interfaz IGO*. 93
- Figura 4.2** Capturas de pantalla de las interfaces testeadas, relacionadas con el proceso de selección de un tipo de ventana y de su colocación sobre la forma o contorno de la cocina: a la izquierda, la interfaz de MD; a la derecha, la interfaz IGO. 93
- Figura 4.3** Capturas de pantalla de las interfaces testeadas, relacionadas con el proceso de mover o cambiar un mueble de posición: a la izquierda, la *interfaz de MD*; a la derecha, la *interfaz IGO*. 94
- Figura 4.4** Hoja de tareas entregada a los participantes, en la que se describe tanto el escenario de uso de la aplicación como los tres grupos de tareas a realizar. 95
- Figura 4.5** Diagramas de cajas y bigotes correspondientes a los *tiempos invertidos en completar de las tareas* (T_{task1} , T_{task2} , T_{task3} , y T_{total}). 106
- Figura 4.6** Diagramas de cajas y bigotes correspondientes al *número de errores e incidencias leves, moderadas, severas y totales* detectadas (E_{slight} , $E_{moderate}$, E_{severe} and E_{Total}) 106

- Figura 4.7** Diagramas de cajas y bigotes correspondientes a las 7 preguntas (Q_i) del *questionario Post-interfaz*. La escala para Q_1 , Q_2 y Q_7 va de 1 (nada) a 7 (completamente); para Q_3 va de 0 a 3; para Q_4 y Q_6 , de 1 a 5. Y para Q_5 se muestra el porcentaje, para cada interfaz, de la opción elegida con más frecuencia..... 108
- Figura 4.8** Áreas calientes y clústeres asociados a las interfaces de MD (izquierda) e IGO (derecha). Cada fila corresponde a un participante..... 115
- Figura 5.1** Gráficas que muestran, para unas condiciones cognitivas normales, cómo suele evolucionar el conocimiento de un sistema y de su interfaz con el uso (la de la izquierda), y la progresión de dicho conocimiento (gráfica de la derecha). 123
- Figura 5.2** Enfoques del diseño de la IU que ignoran los factores ambientales, dando lugar a rendimientos diferentes para la misma capacidad del usuario. (b) Enfoque de interfaz AGILE, que encapsula el factor ambiental para ofrecer un rendimiento unificado y aumentado para el mismo usuario y capacidad..... 126
- Figura 5.3** Estructura del prototipo inicial de *interfaz AGILE*, aplicado a una tableta (*IPad*). Consiste, de izquierda a derecha y de arriba a abajo, en un *mapa de objetivos*, un *mapa de pasos*, un *área de asistencia* y un *área de interacción*. 136
- Figura 5.4** Primer prototipo de *interfaz AGILE*, mostrado en un PC con MS Power Point. Se puede apreciar, de izquierda a derecha y de arriba a abajo, nubes con los objetivos iniciados (*mapa de objetivos*); globos de texto explicando cada paso (*mapa de pasos*); un asistente virtual (*área de asistencia*); y el *área de trabajo o de interacción*..... 139
- Figura 5.5** Ejemplo de pantalla correspondiente a la segunda iteración del prototipo de *interfaz AGILE* para la compra de billetes de tren, mostrado en un iPad2. Tenga en cuenta la ausencia de los mapas de objetivos y de pasos. En cambio, permanece la figura del asistente virtual (*área de asistencia*) y la zona de trabajo (*área interactiva*). 140
- Figura 5.6** Ejemplo de pantalla correspondiente al sitio web ferroviario que los participantes tuvieron que utilizar para realizar la compra de billetes de tren. En particular, se corresponde con la pantalla en que el usuario tenía que seleccionar el horario, el tipo de billete y el precio..... 141
- Figura 5.7** Pantalla correspondiente al sitio web utilizado para la transacción de compra de libros. En particular, cuando el título del libro ya había sido encontrado y había que indicar la cantidad de ejemplares deseados..... 142
- Figura 5.8** Una de las pantallas de la segunda iteración de la *interfaz AGILE* para la transacción para la compra de billetes de tren. En particular, para seleccionar la opción de orientación del asiento..... 145

- Figura 5.9** (a) Árbol de tareas correspondientes al sitio web ferroviario (con 12 tareas: 7 obligatorias con 5 con valores predeterminados y 5 opcionales). (b) Árbol de tareas correspondiente a la interfaz AGILE, y para la misma transacción digital (con 14 tareas, de las cuales 14 eran obligatorias y ninguna con valor predeterminado). Nótese la posibilidad de avanzar en el paso 6, y de volver atrás desde el paso 11. 146
- Figura 5.10** Una de las pantallas resultantes de la segunda iteración de *la interfaz AGILE* para la transacción para la compra de billetes de tren. En particular, para elegir la hora del viaje..... 147
- Figura 5.11** Detalle de la interfaz AGILE para la compra de libros. En particular, para indicar el número de libros a comprar. La línea representa el scanpath del participante (74 años), es decir, la trayectoria que la mirada del participante dibuja sobre la grabación de la escena. Los círculos representan las fijaciones del participante, y el tamaño de dichos círculos significa la duración de la fijación 148
- Figura 5.12** Una de las pantallas resultantes de la segunda iteración de *la interfaz AGILE* para la transacción consistente en la compra de billetes de tren. En particular, para elegir las características adicionales del *asiento*. 151

Listado de Tablas

Tabla 2.1 Variables utilizadas por los distintos autores para establecer sus categorías de usuarios.	26
Tabla 2.2 Revisión del concepto de usuario inexperto e infrecuente en las distintas clasificaciones de usuario.....	26
Tabla 3.1 Descripción, características, y ventajas e inconvenientes de los principales Estilos de Interacción de cara a un uso ocasional.	60
Tabla 4.1 Resumen de los principales datos y características de los participantes, recopilados mediante el Cuestionario Pre-test (Background Questionnaire)	91
Tabla 4.2 <i>Tiempos</i> (en segundos) invertidos por cada participante <i>en completar las tareas</i>	96
Tabla 4.3 Resumen de los errores (Ligeros/Moderados/Severos) detectados durante las pruebas, incluyendo el tipo de interfaz involucrada (MD o IGO) y las tareas en que sucedieron (1, 2, y/o 3).....	98
Tabla 4.4 Cantidad de <i>errores</i> por participante y tipo de interfaz.	99
Tabla 4.5 Respuestas de cada participante al <i>cuestionario Post-interfaz</i>	101
Tabla 4.6 Respuestas de cada participante al <i>cuestionario Post-interfaz</i>	102
Tabla 4.7 Estadísticas correspondientes a los <i>tiempos invertidos en completar de las tareas</i> (T_{task1} , T_{task2} , T_{task3} , y T_{total}).	105
Tabla 4.8 Estadísticas correspondientes al número de errores e incidencias leves, moderadas, severas y totales detectadas (E_{slight} , $E_{moderate}$, E_{severe} and E_{total}).	106
Tabla 4.9 Resultados de los tests de Wilcoxon sobre los tiempos invertidos en completar de las tareas y sobre el número de errores.....	107
Tabla 4.10 Algunos resultados interesantes relacionados con los tiempos invertidos en completar las tareas y sobre el número de errores.	107

Tabla 4.11 Estadísticas correspondientes a las respuestas (Q_i) al cuestionario Post-interfaz.
..... 109

Tabla 4.12 Resultados de las respuestas al Cuestionario Comparativo: porcentaje de
participantes que prefieren un estilo de interacción al otro. 110

Bibliografía

- [1] Americans with Disabilities. Act of 1990 incorporating the changes made by the ADA Amendments Act of 2008. (2008) <http://www.ada.gov/pubs/adastatute08.pdf> Visitado: 01/08/2016
- [2] Annett, J., Duncan, K.D. (1967) Task Analysis and Training Design. *Occupational Psychology*, 41, 211–221.
- [3] Aykin, N.M., Aykin, T. (1991) Individual differences in human-computer interaction. *Comp. and Ind. Eng.*, 20(3): 73–379. [http://dx.doi.org/10.1016/0360-8352\(91\)90009-U](http://dx.doi.org/10.1016/0360-8352(91)90009-U)
- [4] Baecker, R. M. & Buxton, W. A. S. (eds.) (1987). *Readings in human-computer interaction: A multidisciplinary approach*. San Mateo, CA: Morgan Kaufmann Publishers.
- [5] Baker, M. (2012). The Difference between Novices and Casual Users. Every Page is Page One. <http://everypageispageone.com/2012/05/25/the-difference-between-novices-and-casual-users/> Visitado: 15/01/2015
- [6] Bargas-Avila, J., Brenzikofer, O., Roth, S., Tuch, A., Orsini, S., Opwis, K. (2010). Simple but crucial user interfaces in the world wide web: introducing 20 guidelines for usable web form design. In *Matrai, R. (ed.), User Interfaces*. INTECH, 1–10.
- [7] Barnum, C.M. (2011). *Usability Testing Essentials: Ready, Set...Test!* Morgan Kaufmann Publishers, Burlington, MA, USA.
- [8] Bitner, M. J., Ostrom, A. L., Meuter, M. L., 2002. Implementing Successful Self-Service Technologies. *Academy of Management Executive*, 16(4), 96-108.
- [9] Bjerre, T. (1977). Construction and implementation of an information query language on the basis of the relational database theory. In *Proceedings of an International Seminar on Intelligent Question-Answering and Data Base Systems, 21-30 June, 1977, Bonas, France* (pp. 191–203). Rocquencourt: Institut de Recherche d'Informatique et d'Automatique.
- [10] Bollaert, J. (2015). Crafting a wizard. In *IBM developerWorks*. <http://www.ibm.com/developerworks/library/us-wizard/> Visitado: 30/05/2015
- [11] Bouzid, A., Ma, W. (2013) Don't make me tap: a common sense approach to voice usability. ISBN 1-492195-1-7
- [12] Brooke, J. 1996. SUS: A quick and dirty usability scale. *Usability Evaluation in Industry*, 189(194), 4–10.
- [13] Brooke, J. 2013. SUS: A Retrospective. *Journal of Usability Studies*, 8(2), 29–40.
- [14] Brusilovsky P., Millán E. (2007) User Models for Adaptive Hypermedia and Adaptive Educational Systems. In: *Brusilovsky P., Kobsa A., Nejdl W. (eds) The Adaptive Web. Lecture Notes in Computer Science, vol 4321*. Springer, Berlin, Heidelberg

- [15] Budi, R. (2017). Wizards: Definition and Design Recommendations. <https://www.nngroup.com/articles/wizards/> Visitado: 26/05/2018.
- [16] Buxton, B. (2007). Multi-touch systems that I have known and loved. *Microsoft Research*, 56, 1–11.
- [17] Card, S., Moran, T., & Newell, A. (1980). Computer text-editing: An information processing analysis for a routine cognitive skill. *Cognitive Psychology*, 12, 32–74.
- [18] Card, S., Moran, T., & Newell, A. (1983). *The Psychology of Human-Computer Interaction*. Hillsdale, New Jersey: Erlbaum.
- [19] Carrillo, A., Falgueras, J.A. (2020). Proposal and testing Goals-Guided Interaction for occasional users. *Human-Centric Computing and Information Sciences*, 10, 4. <https://doi.org/10.1186/s13673-020-0209-2>
- [20] Carrillo, A.L., Falgueras, J. (2019). GDI as an Alternative Guiding Interaction Style for Occasional Users. In: *Lamas D., Loizides F., Nacke L., Petrie H., Winckler M., Zaphiris P. (eds) Human-Computer Interaction – INTERACT 2019*. INTERACT 2019. Lecture Notes in Computer Science, vol 11749. Springer, Cham. https://doi.org/10.1007/978-3-030-29390-1_5
- [21] Carrillo A.L., Martínez, S., Falgueras, J., Scott-Brown, K. (2017). A reflective characterisation of occasional user. *Computers in Human Behavior*, 70, 74–89. <https://doi.org/10.1016/j.chb.2016.12.027>
- [22] Carrillo, A.L., Falgueras, J. (2015). Goal Driven Interaction vs. Direct Manipulation, an empirical comparison. In: *Proceedings of the XVI International Conference on Human Computer Interaction (Interacción'15)*. ACM, New York, NY, USA. <https://doi.org/10.1145/2829875.2829892>
- [23] Carrillo A.L., Martínez, S., Falgueras, J., Scott-brown (2012). Diseño y evaluación de interfaces para la tercera edad. In: *Proceedings of the XIII International Conference on Human Computer Interaction (Interacción'12)*.
- [24] Carrillo, A.L., Falgueras, J., Guevara, A. (2006). A notation for Goal Driven Interfaces Specification. In: *Navarro-Prieto R., Vidal J. (eds.) HCI related papers of Interacción 2004*. Springer, Dordrecht
- [25] Carrillo, A.L., Falgueras, J., Guevara, A. (2006b). GDIT - Tool for the Design, Specification and Generation of Goals Driven User Interfaces. In: *Manolopoulos, Y., Filipe, J., Constantopoulos, P., Cordeiro, J. (eds.) ICEIS 2006 - Proceedings of the Eighth International Conference on Enterprise Information Systems - HCI*. <https://doi.org/10.5220/0002496601350138>
- [26] Carroll, J. M., Thomas, J.C. (1982). Metaphor and the cognitive representation of computing systems. *Systems, Man and Cybernetics, IEEE Transactions on*, 12, 107–116.
- [27] Charland, A., & Leroux, B. (2011). Mobile application development: web vs. native. *Communications of the ACM*, 54(5), 49–53.
- [28] Chisnell, D. (2007) Where technology meets green bananas. *Interact.*, 14(2):10–11. <http://dx.doi.org/10.1145/1229863.1229876>

- [29] Clarkson, J., Coleman R. (eds). (2015). Special Issue: Inclusive Design. *Applied Ergonomics*, 46, part B, January, 233–324.
- [30] Codd, E. F. (1974). Seven Steps to Rendezvous with the Casual User. In *Proceedings IFIP TC-2 Working Conference on Data Base Management Systems*, 1-5 April, 1974, Cargese, Corsica; Published In J.W. Klimbie & K.I. Koffeman (eds.), *Data Base Management*. North-Holland.
- [31] Coe, M. (1996). *Human Factors for Technical Communicators*. New York: John Wiley & Sons.
- [32] Cooper, A. (1999). *The inmates are running the asylum: Why high-tech products drive us crazy and how to restore the sanity*. Indianapolis: Sams.
- [33] Cooper, A. (2007). *About face 3.0: The Essentials of Interaction Design*. Wiley Publishing, Inc.
- [34] Cuff, R. (1980). On casual users. *International Journal of Man-Machine Studies*, 12, 163–187.
- [35] Disneyland Paris official app. (2016). <http://www.disneylandparis.co.uk/guest-services/mobile-app/> Visitado: 21/04/2016
- [36] Dix, A., Finlay, J., Abowd, G., Beale, R. (2004). *Human-Computer Interaction*. (3th ed.). (Chapter 3). Pearson Education. Prentice Hall.
- [37] Eason, K. D. (1976). Understanding the naive computer user. *The Computer Journal*, 19(1), 3–7.
- [38] Eurostat, 2017. “A look at the lives of the elderly in the EU today”. A web tool released by Eurostat, the statistical office of the European Union. <https://ec.europa.eu/eurostat/cache/infographs/elderly/index.html> Visitado: 10/08/2019
- [39] Fourney, D.W., Carter, J.A, (2006) A Standard Method Of Profiling The Accessibility Needs Of Computer Users With Vision And Hearing Impairments. In M.A. Hersh (ed.), *Procs of Conf. and Workshop on Assistive Technologies for People with Vision and Hearing Impairments, Technology for Inclusion CVHI Kufstein, Austria*
- [40] Fowler, M., Highsmith, J. (2001) The AGILE manifesto, *Software Development*, 9(8): 28–35
- [41] Gentner, D., Nielsen, J. (1996). The Anti-Mac interface, *Communications of the ACM* 39(8), 70–82.
- [42] Gillingham, P. (2014). Information systems and human service organizations: Managing and designing for the “occasional user”. *Human Service Organizations: Management, Leadership & Governance*, 38(2), pp. 169–177.
- [43] Goldsmith, S. (1976). *Design for the Disabled*. Royal Institute of British Architects.
- [44] Gong, J., & Tarasewich, P. (2004). Guidelines for handheld mobile device interface design. In *Proceedings of DSI 2004 Annual Meeting*, pp. 3751–3756.
- [45] Gorakala, S.K., Usulli, M. (2015). *Building a Recommendation System with R*. Packt Publishing Ltd. Birmingham, UK, 15–18.

- [46] Hansen, W. J. (1971). User engineering principles for interactive systems. In *Proceedings of the November 16-18, fall joint computer conference*, pp. 523–532.
- [47] Hanson, V.L. (2010) Influencing technology adoption by older adults. *Interacting with Computers*, 22(6): 502–509. <http://dx.doi.org/10.1016/j.intcom.2010.09.001>
- [48] Hargittai, E. (2010) Digital Na(t)ives? Variation in Internet Skills and Uses among Members of the “Net Generation”. *Sociological Inquiry*, 80(1): 92–113. <http://dx.doi.org/10.1111/j.1475-682X.2009.00317.x>
- [49] Hartson, H. (1998). Human-computer interaction: Interdisciplinary roots and trends. *The Journal of Systems and Software*, Elsevier, 43, 103–118.
- [50] Hartson, H. and Hix, D. (1989). Human-computer interface development: concepts and systems for its management. *ACM Computing Surveys*, 21(1), 5–92.
- [51] Hawthorn, D. (2000) Possible implications of aging for interface designers. *Interacting with computers*, 12(5), pp.507–528. [http://dx.doi.org/10.1016/S0953-5438\(99\)00021-1](http://dx.doi.org/10.1016/S0953-5438(99)00021-1)
- [52] Hewett, T.T., Baecker, R., Card, S., Carey, T., Gasen, J., Mantei, M., Perlman, G., Strong, G., Verplank, W. (1992). *ACM SIGCHI Curricula for Human-Computer Interaction*. ACM, New York.
- [53] Hix, D., Hartson, H. (1993) *Developing user interfaces: ensuring usability through product and process*. New York, New York, United States of America. John Wiley and Sons, 471-57813-4
- [54] Hollnagel, E., Woods, D. (2005). *Joint Cognitive Systems: Foundations of Cognitive Systems Engineering*. Boca Raton, FL: CRC Press.
- [55] Hutchins, E.L., Hollan, J.D., Norman, D.A. (1986). Direct manipulation interfaces. In *D.A. Norman & S.W. Draper (eds.), User-centered system design*. Hillsdale, NJ: Erlbaum, 87–124.
- [56] Hyppönen, H. (2001) Industry awareness and transfer in Europe and the USA. In: *Roe, Patrick R.W. (ed). Bridging the Gap? Access to telecommunications for all people. European Commission.*
- [57] IKEA Home Planner. (2015). <http://homeplanner.ikea.com/US/UI/Pages/VPUI.htm> Visitado: 06/10/2015
- [58] IKEA kitchen planning tool (2012) http://www.ikea.com/ms/en_US/rooms_ideas/kitchen/download1.html. Visitado: 01/04/2012
- [59] ISO 9241-11: Ergonomics of human-system interaction — Part 11: Usability: Definitions and concepts (2015)
- [60] ISO 9241-171:2008: Ergonomics of human-system interaction — Part 171: Guidance on software accessibility. http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=39080. Visitado: 4/05/2012
- [61] ISO 9241-210:2010. Ergonomics of human-system interaction — Part 210: Human-centred design for interactive systems. (2010).

http://www.iso.org/iso/catalogue_detail.htm?csnumber=52075
Visitado: 02/12/2013

- [62] Jacko, J.A., Vitense, H. (2001) A review and reappraisal of information technologies within a conceptual framework for individuals with disabilities. *Universal Access in the information Society*, 1(1): 56–76
- [63] Jette, A.M. (2006) Toward a common language for function, disability, and health. *Physical Therapy*, 86(5): 726–734
- [64] John, B.E., Kieras, D.E. (1994). The GOMS family of analysis techniques: Tools for design and evaluation. Carnegie Mellon University School of Computer Science Technical Report No. CMU-CS-94-181. Also appears as the Human-Computer Interaction Institute Technical Report No. CMU-HCII-94-106.
- [65] John, B.E., Kieras, D.E. (1996). The GOMS family of user interface analysis techniques: Comparison and contrast. *ACM Transactions on Computer-Human Interaction*, 3(4), December 1996, 320–351. <http://dx.doi.org/10.1145/235833.236054>
- [66] John, B.E., Kieras, D.E. (1996b). Using GOMS for User Interface Design and Analysis: Wich Technique? *ACM Transactions on Computer-Human Interaction*, 3(4), December 1996, 287–319.
- [67] Johnson, J., Roberts, T.L., Verplank, W., Smith, D.C., Irby, C.H., Beard, M., Mackey, K. (1989) The xerox star: A retrospective. *Computer*, 22, 11–26, 28–29.
- [68] Johnson, M. (2007). Unscrambling the Average User of Habbo Hotel. *Human Technology*, 3(2), 127–153.
- [69] Johnson, P. (1989). Task Knowledge Structures. In Diaper D. (ed.) *Task analysis in Human Computer Interaction*. Ellis Horwood.
- [70] Karat, C.M., Lai, J., Yankelovich, S.N. (2012). In book: *The Human Computer Interaction Handbook*. Third Edition. Chapter: *Speech and Language Interfaces, Applications, and Technologies*. CRC, Taylor and Francis Group.
- [71] Kennedy, T. C. S. (1975). Some behavioural factors affecting the training of naïve users of an interactive computer system. *International Journal of Man-Machine Studies*, 7, 817–834.
- [72] Kieras, D. (1997). A Guide to GOMS Model Usability. Evaluation using NGOMSL. In M. Helander & T. Landauer (Eds.), *The handbook of human-computer interaction*. (2th ed.). Amsterdam: North-Holland, 733–766.
- [73] Kieras, D. (1997b). Task Analysis and the Design of Functionality. In A. Tucher (ed.) *The computer Science and Engineering Handbook*. Boca Raton, CRC Inc, 1401–1423.
- [74] Kimani S. (2009). WIMP Interfaces. In Liu L., Özsu M.T. (eds.) *Encyclopedia of Database Systems*. Springer, Boston, MA.
- [75] Kinsella, K., He, W. (2009) An Aging World: 2008. US Census Bureau, Int. Population Reports

- [76] Lane, D. M., Napier, H.A., Peres, S. C., Sándor, A. (2005). The Hidden Costs of Graphical User Interfaces: Users' Failures to Make the Transition from Menus and Icon Tool Bars to Keyboard Shortcuts. *International Journal of Human Computer Interaction*, 18(2), 133-144.
- [77] Langdon, P., Thimbleby, H. (2010) Inclusion and interaction: Designing interaction for inclusive populations. *Interacting with Computers*, 22(6): 439–448. <http://dx.doi.org/10.1016/j.intcom.2010.08.007>
- [78] Langdon, P., Persad, U., John Clarkson, P. (2010) Developing a model of cognitive interaction for analytical inclusive design evaluation. *Interacting with Computers*, 22(6): 510–529. <http://dx.doi.org/10.1016/j.intcom.2010.08.008>
- [79] Laubheimer, P., Budi, R. (2018). Intelligent Assistants Have Poor Usability: A User Study of Alexa, Google Assistant, and Siri. <https://www.nngroup.com/articles/intelligent-assistant-usability/> Visitado: 23/03/2019.
- [80] Lieberman, H. (1998). Integrating user interface agents with conventional applications. In *Proceedings of the 3rd international conference on Intelligent user interfaces*. ACM, New York, NY, USA, 39–46.
- [81] Ling, R. (2008). Exclusion or self-isolation? Texting and the elderly users. *The information society*, 24(5).
- [82] Loranger, H. (2016). Checklist for Planning Usability Studies. NN/g Nielsen Norman Group. www.nngroup.com/articles/usability-test-checklist
- [83] Lorés, J. (ed.). Abascal, J., Cañas, J.J., Gea, M., Gil, A.B., Lorés, J., Martínez, A.B., Ortega, M., Valero, P., Vélez, M. (2001) *La Interacción persona ordenador*. Libro electrónico editado por AIPO. Versiones disponibles en la red (<https://aipo.es/content/el-libro-electronico>) y CD-ROM (ISBN 84-607-2255-4).
- [84] Lough, D. E., Burns, A. D. (1977). *An Analysis of Data Base Query Languages*. Naval Postgraduate School Monterey California.
- [85] Louvre audio guide. (2016). <http://www.louvre.fr/en/audio-guide> Visitado: 06.04.2016
- [86] Madden, R., Hogan, T. (1997). *The definition of disability in Australia: moving towards national consistency*. Canberra. AIHW.
- [87] Maes P. (1994). Agents that reduce work and information overload. *Communications of the ACM*, 37(7).
- [88] Mann, W. C. (1975). Why things are so bad for the computer-naïve user. In: *Proceedings of the AFIPS National Computer Conference 44, May, 1975* (pp. 785-787).
- [89] Marschollek, M., Mix, S., Wolf, K. H., Effertz, B., Haux, R., Steinhagen-Thiessen, E. (2007). ICT-based health information services for elderly people: past experiences, current trends, and future strategies. *Informatics for Health and Social Care*, 32(4), 251–261. <http://dx.doi.org/10.1080/14639230701692736>

- [90] Marsden, P., Hollnagel, E. (1996). Human interaction with technology: The accidental user. Usage of Modern Technology by Experts and Non-professionals. *Acta Psychologica*, 91(3), 345–358.
- [91] Martin, J. (1973). *Design of Man-Computer Dialogues*. Prentice-Hall.
- [92] Martinez, S., Sloan, R., Szymkowiak, A., Scott-Brown, K. 2010 Using Virtual Agents to Cue Observer Attention: Assessment of the impact of agent animation. The Second Int. Conf. on Creative Content Technologies CONTENT 2010: 7-12. November 21-26, Lisbon, Portugal. ISBN: 978-1-61208-001-7
- [93] Martinez, S. et al., 2012. Animated Virtual Agents to Cue User Attention. *International Journal On Advances In Intelligent Systems*, 4 (3 & 4), pp.299–308
- [94] Martinez, S., Carrillo, A. L., Scott-Brown, K. C., Falgueras, J. (2013). AGILE Interface for ‘No-Learning Nor Experience Required’ Interaction. In: *E. Martín et al. (eds.), User Modeling and Adaptation for Daily Routines: Providing Assistance to People with Special Needs* (pp. 119–151), Human-Computer Interaction Series, Springer-Verlag London. https://doi.org/10.1007/978-1-4471-4778-7_5
- [95] McDonough, F.A., Ladner, R.E., Roth, W., Scadden, L.A., Vanderheiden, G.C. (1998). Public Law: 99-506, “Section 508” Electronic Equipment Accessibility for Disabled Workers. In: O’Hare, J.J. (ed) *Proceeding of the SIGCHI Conference on Human Factors in Computing Systems (CHI’88)*, ACM, New York, pp 219–222. <https://doi.org/10.1145/57167.57204>
- [96] Meuter, M. L., Ostrom, A. L., Bitner, M. J., & Roundtree, R. (2003). The influence of technology anxiety on consumer use and experiences with self-service technologies. *Journal of Business Research*, 56(11), 899–906.
- [97] Meuter, M. L., Ostrom, A. L., Roundtree, R. I., & Bitner, M. J. (2000). Self-Service Technologies: Understanding Customer Satisfaction with Technology-Based Service Encounters, *Journal of Marketing*, 64(3), 50–64.
- [98] Microsoft (2015). Visitado: 28/05/2018 Wizards. <https://docs.microsoft.com/es-es/windows/win32/uxguide/win-wizards>
- [99] Middleton, S.E. (2002). Interface agents: A review of the field. ArXiv, cs.MA/0203012.
- [100] Miller, C.S., Remington, R.W. (2004). Modeling information navigation: implications for information architecture. *Human-Computer Interact.*19(3), September 2004, 225–271.
- [101] Moldt, D., & Scheve, C.V. (2001). Emotions and Multimodal Interface-Agents: A Sociological View. *Mensch & Computer*.
- [102] Moran, T.P. (1981). An Applied Psychology of the User. *Computing*, 13. <http://dx.doi.org/10.1145/356835.356836>
- [103] Mori, G., Paternò, F., Santoro, C. (2002). CTTE: Support for Developing and Analysing Task Models for Interactive System Design. *IEEE Transactions on Software Engineering*. 28(8), 797–813.
- [104] Mueller, J. P. (2003). *Accessibility for Everybody: Understanding the Section 508 Accessibility Requirements*. Apress.

- [105] Myers, B., Hudson, S. E., & Pausch, R. (2000). Past, present, and future of user interface software tools. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 7(1), 3–28.
- [106] Naur, P. et al. (Eds) (1960). Report on the Algorithmic Language ALGOL'60. *Communications of the ACM*, 3, 299–314.
- [107] Neustein, A., Markowitz, J.A. (2013). Mobile speech and advanced natural language solutions. Heidelberg, Germany: Springer.
- [108] Newell, A.F. (1993) Interfaces for the ordinary and beyond. *Software, IEEE*, 10(5): 76–78. <http://dx.doi.org/10.1109/52.232406>
- [109] Newell, A.F. (2006) Older people as a focus for inclusive design. *Gerontechnology* 4 (4): 190–199. <http://dx.doi.org/10.4017/gt.2006.04.04.003.00>
- [110] Newell, A.F. (2008) Accessible Computing—Past Trends and Future Suggestions: Commentary on “Computers and People with Disabilities.” *ACM Trans. on Accessible Comp. (TACCESS)*, 1(2): 9
- [111] Newell, A.F., Simon, H. (1972). *Human Problem Solving*. Prentice-Hall Englewood Cliffs, NJ
- [112] Nicolle, C., Abascal, J. (2001) Inclusive design guidelines for HCI. In *Procs of INCLUDE 2001*, 18-20 April, London
- [113] Nicolle, C., Peters, B. (1999) Elderly and Disabled Travelers: Intelligent Transp. Systems Designed for the 3rd Millennium. *Transp. Hum. Factors*, 1(2): 121–134. http://dx.doi.org/10.1207/sthf0102_1
- [114] Nielsen, J. (1993) Usability engineering. San Diego, California, United States of America. Academic Press. ISBN 0-12-518405-0
- [115] Nielsen, J. (1993b) Iterative user-interface design. *Comp.*, 26(11): 32–41. <http://dx.doi.org/10.1109/2.241424>
- [116] Nielsen, J. (1994). Heuristic evaluation. In: *Nielsen, J., and Mack, R.L. (eds.), Usability Inspection Methods*, John Wiley & Sons, New York, NY.
- [117] Nielsen, J. (1995) *Usability engineering*. Morgan Kaufmann
- [118] Nielsen, J., Budiu, R. (2012). *Mobile usability*. New Riders.
- [119] Nielsen, J. (2006). Quantitative Studies: How Many Users to Test? NN/g Nielsen Norman Group. www.nngroup.com/articles/quantitative-studies-how-many-users/ Visitado: 17/11/2018.
- [120] Norman, D.A. (1983). Some observations on mental models. *Mental models*, 7(112), 7–14.
- [121] Norman, D.A. (2002). *The Design of Everyday Things*. Basic Books, Inc., New York, NY, USA.
- [122] Norman, D.A. (2010). Natural user interfaces are not natural interactions. *Interactions*, 17(3), 6–10.
- [123] Norman, D.A., Draper, S.W. (eds) (1986). *User-centered system design*. Hillsdale, NJ: Erlbaum, 87–124.

- [124] Paap, K., Roske-Hofstrand, R. (1988). Design of menus. In: *Helander, M. (ed.), Handbook of Human-Computer Interaction*, 205–235.
- [125] Paap, K.R., Roske-Hofstrand, R.J. (1986). The Optimal Number of Menu Options per Panel. *Human Factors*, 28(4), 377–385.
- [126] Paternò, F. (2003). ConcurTaskTrees: An Engineered Notation for Task Models. In: *D. Diaper, & N. Stanton, (eds.), The Handbook of Task Analysis for Human-Computer Interaction*. Lawrence Erlbaum Associates, Mahwah, *ch. 24*, 483–503.
- [127] Payne, S.J., Green, T.R.G. (1990). Task Action Grammar: recent developments. In: *D. Diaper (Ed.) Approaches to Task Analysis*. Cambridge University Press.
- [128] Pieper, M., Morasch, H., Piéla, G. (2003) Bridging the educational divide. *Univ. Access in the Inf. Soc.*, 2(3): 243–254. doi: <http://dx.doi.org/10.1007/s10209-003-0061-y>
- [129] Pieraccini, R. (2012). *Voice in the Machine*. Cambridge, MA: MIT Press.
- [130] Pirsig, R.M. (1974). *Zen and the art of motorcycle maintenance*. New York: William Morrow and Company.
- [131] Porrero, I.P. (1998) *Improving the quality of life for the European citizen: technology for inclusive design and equality*, IOS Press
- [132] Potosnak, K., Hayes, P.J., Rosson, M.B., Schneider, M.L., Whiteside, J.A. (1986) Classifying users: a hard look at some controversial issues. *ACM SIGCHI Bulletin*, 17(4): 84–88. <http://dx.doi.org/10.1145/22339.22353>
- [133] Preece, J., Rogers, Y., Sharp, H., Benyon, D., Holland, S., Carey, T. (1994). *Human computer interaction*. Harlow, Essex, England. Addison Wesley Longman Limited.
- [134] Programa PADRE. Personal Income Tax Help programs. (2014). <http://www.agenciatributaria.es/AEAT.internet/renta/padre.shtml> Visitado: 10.11.2016
- [135] Pruitt, J., Adlin, T. (2010). *The persona lifecycle: keeping people in mind throughout product design*. Morgan Kaufmann.
- [136] Pullin, G., Newell, A. (2007) Focussing on extra-ordinary users. In: *Procs of the 4th Int. Conf. on Univ. access in hum. comp. interact.: coping with diversity*: 253–262
- [137] Reed, A. (2013). The importance of “Occasional User” requirements. TechWell Corp. <http://www.techwell.com/2013/06/importance-occasional-user-requirements> Visitado: 06/12/2014
- [138] Rensink, R.A. et al (2010) To see or not to see : the need for attention to perceive changes in scenes. *Soc.*, 8(5): 368-373
- [139] Rogers, Y., Sharp, H., Preece, J. (2011) *Interaction design: Beyond human-computer interaction*, Wiley
- [140] Rutkowski, C. (1982). An introduction to the human applications standard computer interface. *Byte*, 7, 291–310.

- [141] Santhanam, R., Wiedenbeck, S. (1993). Neither novice nor expert: The discretionary user of software. *International Journal of Man-Machine Studies*, 38, 201–229.
- [142] Savidis, A., Stephanidis, C. (2004) Unified user interface design: designing universally accessible interactions. *Interact. with Comp.*, 16(2), 243–270.
- [143] Schiaffino, S., Amandi, A. (2004). User-interface agent interaction: Personalization issues. *International Journal of Human-Computer Studies*, 60, 129–148.
- [144] Schneider, M. L. (1981). Models for the Design of Static, Software Systems. In: *Proceeding of the Workshop on Human Computer Interaction, March, 1981, Atlanta, Georgia* (pp. 65–75).
- [145] Scott-Brown, K.C., Baker, M.R., Orbach, H.S. (2000) Comparison Blindness. *Vis. Cogn.*, 7(1-3): 253-267
- [146] Seckler, M., Heinz, S., Bargas-Avila, J.A., Opwis, K., Tuch, A.N. (2014). Designing Usable Web Forms – Empirical Evaluation of Web Form Improvement Guidelines. *Proceedings of the 2014 annual conference on Human factors in computing systems*, 1275–1284.
- [147] Selker, T. (2008). Touching the future. *Communications of the ACM*, 51(12).
- [148] SensoMotric Instruments (2012a) Eye & Gaze Tracking Systems. <http://www.smivision.com/>. Accessed 16/10/2012
- [149] SensoMotric Instruments (2012b) SMI Eye Tracking Glasses. <http://eyetracking-glasses.com/>. Accessed 16/10/2012
- [150] Shapiro, S.C., Kwasny, S.C. (1975). Interactive consulting via natural language. *Communications of the ACM*, 18, 459–462.
- [151] Sharp, H., Preece, J., Rogers, Y. (2019). *Interaction Design: Beyond Human-Computer Interaction 5th Edition*. John Wiley & Sons, Indianapolis, Indiana.
- [152] Sharpless, M. (2000). The design of personal mobile technologies for lifelong learning. *Computers & Education*, 34(3), 177–193.
- [153] Shneiderman, B. (1980). *Software Psychology: Human Factors in Computer and Information Systems*. Cambridge, MA. Winthrop.
- [154] Shneiderman, B. (1983). Direct Manipulation: A Step Beyond Programming Languages. *IEEE Computer*, 16(8), 57–69. <http://dx.doi.org/10.1109/MC.1983.1654471>
- [155] Shneiderman, B. (1986). *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. Addison-Wesley Publishers, MA
- [156] Shneiderman, B., Plaisant C. (2005). *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. (4th ed.). Addison-Wesley Publishers.
- [157] Shneiderman, B., Plaisant C., Cohen, M., Jacobs, S., Elmqvist, N., Diakopoulos, N. (2016). *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. (6th ed.). Pearson.

- [158] Sloan, R.J.S. (2011) Emotional avatars: choreographing emotional facial expression animation. PhD thesis: 67, 73, 82, 86, 87. Univ. of Abertay Dundee
- [159] Smith, D.K., Alexander, R.C. 1999. Fumbling the Future: How Xerox Invented, Then Ignored, the First Personal Computer. Lincoln, NE. iUniverse.com, Inc.
- [160] Sova, D.H., Nielsen, J. How to Recruit Participants for Usability Studies. NN/g Nielsen Norman Group. www.nngroup.com/reports/how-to-recruit-participants-usability-studies Visitado: 28/10/2017.
- [161] Stasko, J. (1996). Future research directions in human-computer interaction. *ACM Comput. Surv.*, 28(4), December.
- [162] Stevens, J.C. (1992) Aging and spatial acuity of touch. *J. of Gerontology*, 47(1), 35–40
- [163] Stevens, J.C. et al (1998) A multimodal assessment of sensory thresholds in aging. *The J.s of Gerontol. Series B: Psychological Sciences and Social Sciences*, 53(4), 263
- [164] Swiffin, A.L. et al (1985) Pal: An effort efficient portable communication aid and keyboard emulator. In: *Procs of the 8th Annual Conf. on Rehabil. Technol.:* 197–199
- [165] Tidwell, J. (2011). *Designing Interfaces: Patterns for Effective Interaction Design*. O'Reilly Media Inc. Gravenstein Highway North, Sebastopol, CA 95472
- [166] Turoff, M. (1997). The design and evaluation of interactive systems. Section 1.4.3: “User Roles and Types”. <http://web.njit.edu/~turoff/coursenotes/IS732/book/tablecon.htm> Visitado: 23.05.2014
- [167] US Rehabilitation Act Amendments Section 508. (1998). <https://www.section508.gov/> Visitado: 01/10/2016
- [168] van Dam, A. (1997). Post-WIMP User Interfaces. *Communications of the ACM*, 40(2),63–67. <http://dx.doi.org/10.1145/253671.253708>
- [169] Vassiliou, Y., Jarke, M. (1982). Query Languages-A Taxonomy. In: *Human Factors and Interactive Computer Systems: Proceedings of the Nyu Symposium on User Interfaces, 26-28 May, 1982, New York*, pp. 47–82.
- [170] WAI-W3C. Web Accesibilit Initiative. Strategies, standards, resources to make the Web accessible to people with disabilities. <https://www.w3.org/WAI/> Visitado: 15/01/2019.
- [171] WCAG. Web Content Accessibility Guidelines. <https://www.w3.org/TR/WCAG21/> Visitado: 20/03/2019.
- [172] Whintont, K. and Budiu, R. (2018b). The Paradox of Intelligent Assistants: Poor Usability, High Adoption. <https://www.nngroup.com/articles/intelligent-assistants-poor-usability-high-adoption/> Visitado: 20/03/2019.
- [173] Whiteside, J., Bennett, J., Holzblatt, K. (1988) Usability engineering: Our experience and evolution. In: *M. Helander ed. Handbook of Human-Computer Interaction: 791-817*, Amsterdam: North Holland

- [174] Whiteside, J., Jones, S., Levy, P., Wixon, D. (1985). User performance with command, menu, and iconic interfaces. In: *Proceedings CHI'85 Human Factors in Computing Systems, 14–18 April, 1985, San Francisco, New York, United States of America* (pp. 185–191). ACM.
- [175] OMS (2001). ICF, International Classification of Functioning, Disability and Health, World Health Organization. ISBN 9241545429.
- [176] Wickham, D.P., Pupons, D., Mayhew, D.L., Stoll, T., Toley III, K.J., Rouiller, S. (2002). *Designing Effective Wizards: A Multidisciplinary Approach*. Upper Saddle River, NJ: Prentice Hall.
- [177] Wilcoxon, F. (1945). Individual Comparisons by Ranking Methods. *Biometrics*, 1, 80–83.
- [178] Williams, G., et al. (1984). The Apple Macintosh Computer. *Byte Magazine*, 9(2), 30–80.
- [179] Willis, L.M., and McDonald, S. (2016) Retrospective protocols in usability testing: a comparison of Post-session RTA versus Post-task RTA reports. *Behaviour & Information Technology*, 35:8, 628-643, <https://www.tandfonline.com/doi/full/10.1080/0144929X.2016.1175506>
- [180] Wilson, T. (1999). User modelling: A global perspective. *Anales De Documentación*, 2(2), 95-102.
- [181] Wintenton, 2015. Menu Design: Checklist of 15 UX Guidelines to Help Users. <https://www.nngroup.com/articles/menu-design/> Visitado: 10/02/2019.
- [182] Zajicek, M. (2006) Aspects of HCI research for older people. *Univers.* Access in the *Inf. Soc.*, 5(3): 279–286. doi: <http://dx.doi.org/10.1007/s10209-006-0046-8>
- [183] Ziegler, J. (1996). Interactive techniques. *ACM Computing Surveys*, Vol. 28(1), March.
- [184] Zloof, M.M. (1975). Query-by-Example. In: *Proceedings of the AFIPS National Computer Conference 44, May, 1975, New Jersey* (pp. 431–438). AFIPS press.
- [185] Zloof, M.M. (1978): Design aspects of the Query-by-Example database management language. In: *Shneiderman, B. (Ed.), Databases: Improving Usability and Responsiveness, London and New York* (pp. 29–55). Academic Press.