

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA
INFORMÁTICA
INGENIERÍA INFORMÁTICA

Aplicación web para la gestión ágil de proyectos
Web application for agile project management

Realizado por
Manuel Valbuena Alba

Tutorizado por
Carlos Manuel Rossi Jiménez

Departamento
Lenguajes y Ciencias de la Computación

UNIVERSIDAD DE MÁLAGA

MÁLAGA, febrero 2021



UNIVERSIDAD
DE MÁLAGA



Resumen

Hoy en día las metodologías ágiles son muy usadas para el desarrollo de proyectos, independientemente del ámbito de estos. Uno de los motivos del éxito de estas metodologías es el mayor grado de satisfacción del cliente, debido a que ya que promueve una mayor implicación del mismo en el proceso del desarrollo. Una de las claves de esto es la estrategia incremental de dichas metodologías, de manera que el cliente recibe a lo largo del desarrollo varias demostraciones del trabajo realizado por los desarrolladores. Algunas de las metodologías ágiles más usadas son Scrum, XP (acrónimo de *extreme programming*), Kanban o Lean.

Otro de los motivos por el que las metodologías ágiles son tan usadas es el trabajo colaborativo dentro del equipo de desarrollo. Este trabajo colaborativo consiste en la división del equipo en distintos roles y en la realización de reuniones de forma frecuente con el fin de conseguir una mejor organización y seguimiento del proyecto. No obstante, es habitual que las herramientas de gestión de proyectos no faciliten una comunicación fluida entre los miembros del equipo de desarrollo y del equipo del cliente.

En este TFG se va a desarrollar una aplicación web para el desarrollo de proyecto ágiles que usen la metodología ágil Scrum. El principal objetivo de esta aplicación es proporcionar funcionalidades sencillas y eficientes de comunicación entre los participantes del proyecto, así como la validación de las tareas (conocidas como historias de usuario en Scrum) realizadas.

Palabras clave: Metodologías ágiles, desarrollo de proyectos, Scrum, equipo de desarrollo, equipo del cliente, Historias de usuario.

Abstract

Nowadays agile methodologies are very used for project development, regardless of the ambit of the project you are developing. One of the main motives of the success of these methodologies is the degree of satisfaction of the client, as he seems himself more involved in the development process. One of the keys of it it's the incremental strategy of these methodologies, so that the client receives some demos of the work realized by the development team throughout the development process. Some of the most used agile methodologies are Scrum, XP (acronym of extreme programming), Kanban or Lean.

Another of the motives of using agile methodologies is collaborative work inside of the development team. This collaborative work consists of dividing the team in various roles and having meetings frequently aiming for a better organization and follow-up of the project. However, it's usual that project management tools do not facilitate a fluid communication between the development team and the client team.

In this TFG, is going to be developed a web application for agile project management using the agile methodology Scrum. The main objective for this application is to give easy and efficient communication functionalities for the project participants, and validation for every task (known as user history in Scum) realized.

Keywords: Agile methodologies, project development, Scrum, development team, client team, User histories.

Índice

Resumen	1
Abstract	1
Índice	1
1 Introducción	1
1.1 Motivación	1
1.2 Solución propuesta	1
2 Estudio preliminar	3
2.1 Contexto	3
2.1.1 ¿Qué es SCRUM?	3
2.1.2 Análisis de aplicaciones similares	4
2.2 Catálogo de usuarios	8
2.3 Descripción general del sistema	8
3 Análisis	11
3.1 Catálogo de requisitos	11
3.2 Casos de uso	13
3.3 Especificación de los casos de uso	17
3.4 Diagramas de clases	30
4 Diseño	33
4.1 Diagramas de secuencia	33
4.2 Modelo físico de datos	38
4.3 Modelado de arquitectura	39
4.4 Maquetado de interfaz de usuario	40
5 Desarrollo de la aplicación	43
5.1 Entorno tecnológico	43
5.2 Implementación de la aplicación	44
5.3 Resultados. Casos de prueba	51
6 Conclusión	59
6.1 Conclusiones	59
6.2 Trabajos futuros	60

Bibliografía	61
Anexo. Manual de Implantación	62
Requisitos	62
Instalación	62
Anexo. Manual de usuario	64
Anexo. Maquetas de interfaz de usuario	74
Anexo. Diagramas de secuencia	103
CRUD Usuarios	103
CRUD Proyectos	108
CRUD Sprints	111
CRUD Historias de usuario	114
Adjuntar archivos a historia de usuario	117
Asignar responsable a historia de usuario	118
Cronograma	119
Log de historias de usuario	120
Validar acceso	121
Validar historias de usuario	122

1 Introducción

1.1 Motivación

Hoy en día hay múltiples aplicaciones y herramientas para el desarrollo y gestión ágil de proyectos. Cada una de estas herramientas con sus funcionalidades y particularidades que las diferencian unas de las otras. Sin embargo, podemos ver como en la gran mayoría de estas aplicaciones falla principalmente la comunicación entre los miembros participantes del proyecto.

Esta comunicación falla ya que en algunas ocasiones el proceso para poder contactar con otros usuarios dentro de la aplicación es muy difícil de ejecutar y en otras simplemente solo se puede establecer comunicación con los miembros de tu equipo de desarrollo.

Lo que se pretende en este TFG, es crear una aplicación que registre las principales funcionalidades que debe tener una herramienta de gestión ágil de proyectos, y que, a su vez, integre un proceso fácil y sencillo de comunicación entre los usuarios de la aplicación.

1.2 Solución propuesta

Para solucionar el problema planteado en el apartado anterior, como se ha comentado, la idea es crear una aplicación que facilite la comunicación entre los usuarios, sin dejar de lado el resto de las funcionalidades de una herramienta de gestión ágil de proyectos.

Para ello la aplicación contará, entre otras, con las siguientes funcionalidades:

- Gestión de usuarios.
- Gestión de proyectos.
- Gestión de sprints.
- Gestión de historias de usuario.
- Validación de historias de usuario.
- Adjuntar archivos a historias de usuario.
- Log de historias de usuario.

Para el desarrollo de la aplicación web, vamos a hacer uso del lenguaje de programación PHP, así como de HTML, CSS, JQuery, JWWidget y Bootstrap para tanto la creación de la aplicación, como para aplicar diseño a la web.

En cuanto a la metodología a usar para desarrollar la aplicación, nos vamos a basar en la metodología ágil Scrum. Para el modelado software se usará UML como referencia.

La memoria de este TFG se estructura en los siguientes capítulos:

- Estudio preliminar: en este apartado se describe el contexto y se evalúan algunas aplicaciones similares a la propuesta. Además, se establece un catálogo con los distintos usuarios que puede haber en la aplicación y se aporta una descripción general del sistema a desarrollar.
- Análisis: en este apartado vamos a recoger y clasificar los distintos requisitos del sistema, así como establecer y detallar los casos de uso. Para finalizar detallaremos los modelos de clases usados, que más adelante servirán para establecer el modelo físico de la base de datos.
- Diseño: aquí, se estructura la implementación de las funcionalidades que debe tener el sistema. Para ello nos ayudaremos de diversos diagramas, como los diagramas de secuencia, el modelado de arquitectura y el modelo físico de datos. Por último, veremos también, el maquetado de interfaz de usuario, que servirá como base para el diseño de la aplicación web.
- Desarrollo de la aplicación: entraremos en detalle sobre cómo se ha implementado la aplicación web, así como la base de datos. También expondremos una serie de casos de prueba que servirán para verificar el correcto funcionamiento de la aplicación.
- Conclusión: expondremos las conclusiones obtenidas una vez finalizado el proceso de desarrollo de la aplicación, a la vez, que se aportará la opinión que nos ha dejado el desarrollar este proyecto.
- Anexo: aportaremos un manual de usuario sobre como instalar y usar la aplicación web, así como todos y cada uno de los diagramas de secuencia y las maquetas de interfaz de usuario.

2 Estudio preliminar

2.1 Contexto

Este proyecto surgió en colaboración con mi tutor, con la idea de crear una aplicación para el desarrollo ágil de proyectos. En la actualidad existen varias aplicaciones con funcionalidades similares a las que se van a desarrollar en este proyecto, sin embargo, la gran mayoría de estas aplicaciones son de pago, ofreciendo pruebas limitadas de forma gratuita. Además, muchas de ellas son excesivamente complejas en lo que se refiere a los procesos de comunicación y validación por parte de clientes externos. Lo que se pretende es crear una aplicación sencilla que ofrezca las funcionalidades básicas para el desarrollo correcto de un proyecto mediante métodos ágiles y que facilite los procesos con los clientes externos mencionados anteriormente.

2.1.1 ¿Qué es SCRUM?

A continuación, se procede a explicar cómo funciona un proyecto desarrollado mediante la metodología ágil SCRUM, que será el método para el cual está pensado el uso de la aplicación a desarrollar en este TFG. SCRUM se basa en la división del proyecto en fases o etapas, conocidas como *sprints*, con una duración corta (no suelen superar las tres o cuatro semanas). En cada una de esas fases se deberán completar todas y cada una de las tareas (historias de usuario) asignadas a dicha etapa. Dentro de SCRUM encontramos varios tipos de perfiles participantes: el Equipo, formado por el jefe de proyecto o *product owner* y los desarrolladores; y el Cliente, formado por el responsable y el equipo del cliente.

Las principales actividades llevadas a cabo en SCRUM son las que se detallan a continuación:

1. Planificación de la iteración: la cual tiene como objetivo seleccionar y establecer los requisitos que tiene que satisfacer el proyecto a desarrollar, y a partir de los requisitos, planificar las tareas que se tienen que realizar.
2. Ejecución de la iteración: el equipo de desarrollo va realizando las tareas pendientes. Cada día se tiene una reunión (daily meeting) a la que cada miembro del equipo responde a las siguientes tres preguntas:
 - ¿Qué se ha hecho desde la última reunión?
 - ¿Qué voy a hacer a partir de que acabe esta reunión?
 - ¿Qué impedimentos tengo para resolver las tareas que tengo que hacer?Durante esta etapa, también se mantienen reuniones con el cliente para tratar de refinar los requisitos.

- Inspección y adaptación: el último día del sprint se hace una revisión de todo el trabajo realizado y se analiza la manera de trabajar del equipo, así como los problemas que se han encontrado durante el desarrollo.

2.1.2 Análisis de aplicaciones similares

Como se ha mencionado anteriormente, en la actualidad existen otras herramientas con funcionalidades similares a las que se plantean en este TFG, con la principal diferencia que, la gran mayoría de estas son herramientas que complican demasiado la comunicación y la validación por parte del cliente, como es el caso de Asana, Taiga o Redmine, y otras no se adaptan del todo a las metodologías ágiles ya que son muy genéricas, como es el caso de Trello. A continuación, se va a realizar un estudio comparativo entre las herramientas mencionadas y la aplicación web de este TFG. Para ello se usará un modelo de calificación con criterios tanto funcionales como no funcionales, el cual se presenta a continuación.

	Ponderación	Valoración Taiga	Calificación Taiga
Criterios funcionales			
Proyectos			
Crear un proyecto	0,8	80	64
Visualizar un proyecto	0,6	80	48
Añadir miembro a un proyecto	0,7	70	49
Formas de ordenar los proyectos	0,6	60	36
Usuario			
Funcionalidades ofrecidas en el perfil	0,8	90	72
Facilidad para contactar con otros usuarios	0,9	30	27
Subir archivos a la web	0,6	70	42
Búsquedas	0,7	80	56
Servicios			
Variedad de los servicios	0,8	50	40
Utilidad de los servicios	0,8	70	56
Criterios no funcionales			
Interfaz amigable	0,9	60	54
Coste	0,8	100	80
Tráfico de usuarios	0,7	80	56
Velocidad de navegación	0,8	80	64
Confianza	1	80	80
TOTAL			824

Tabla 2.1: Modelo de calificación de Taiga.

En la tabla 2.1 se ha evaluado el portal web Taiga, siendo este uno de los más usado hoy en día para la gestión ágil de proyectos. Se puede observar que es un portal bastante completo en lo que a la gestión de proyectos y usuarios se refiere, sin embargo, no ofrece una gran variedad de servicios. En relación con los criterios no funcionales, vemos que en

general, también es bastante buena, en especial el coste, ya que es gratis, a pesar de que su interfaz no es del todo amigable.

	Ponderación	Valoración Trello	Calificación Trello
Criterios funcionales			
Proyectos			
Crear un proyecto	0,8	80	64
Visualizar un proyecto	0,6	90	54
Añadir miembro a un proyecto	0,7	90	63
Formas de ordenar los proyectos	0,6	80	48
Usuario			
Funcionalidades ofrecidas en el perfil	0,8	70	56
Facilidad para contactar con otros usuarios	0,9	60	54
Subir archivos a la web	0,6	70	42
Búsquedas	0,7	90	63
Servicios			
Variedad de los servicios	0,8	90	72
Utilidad de los servicios	0,8	90	72
Criterios no funcionales			
Interfaz amigable	0,9	70	63
Coste	0,8	60	48
Tráfico de usuarios	0,7	80	56
Velocidad de navegación	0,8	80	64
Confianza	1	80	80
TOTAL			899

Tabla 2.2: Modelo de calificación de Trello.

En la tabla 2.2 se ha evaluado el portal Trello. En cuanto a los criterios funcionales, observamos que destaca en la gestión de proyectos, así como en los servicios que ofrecen, sin embargo, la gestión de usuarios no es tan buena, especialmente en que no es una aplicación muy especializada en el uso de metodologías ágiles. La evaluación de los criterios no funcionales en general es bastante parecida a la de la tabla anterior, aunque en este caso la peor valoración se la lleva el coste, ya que las tarifas que ofrece Trello son caras. A pesar del precio a pagar para tener acceso a la versión premium, Trello es el portal con la mejor valoración total entre los portales estudiados.

	Ponderación	Valoración Asana	Calificación Asana
Criterios funcionales			
Proyectos			
Crear un proyecto	0,8	80	64
Visualizar un proyecto	0,6	90	54
Añadir miembro a un proyecto	0,7	90	63
Formas de ordenar los proyectos	0,6	70	42
Usuario			
Funcionalidades ofrecidas en el perfil	0,8	70	56
Facilidad para contactar con otros usuarios	0,9	80	72
Subir archivos a la web	0,6	70	42
Búsquedas	0,7	90	63
Servicios			
Variedad de los servicios	0,8	90	72
Utilidad de los servicios	0,8	70	56
Criterios no funcionales			
Interfaz amigable	0,9	70	63
Coste	0,8	70	56
Tráfico de usuarios	0,7	70	49
Velocidad de navegación	0,8	80	64
Confianza	1	80	80
TOTAL			896

Tabla 2.3: Modelo de calificación de Asana.

En la tabla 2.3 se ha evaluado el portal Asana. Empezando por los criterios funcionales, podemos observar que Asana es el portal que mejor valoración tiene de todos los portales evaluados, dónde destaca principalmente la gestión de los proyectos, así como la variedad de servicios que ofrecen. Con relación a los criterios no funcionales observamos unas valoraciones bastante estándar sin que haya ningún criterio que destaque demasiado. La calificación total de Asana es de 896, que es solo 3 puntos menos que la calificación de Trello, por lo que Asana es otra muy buena opción para la gestión ágil de proyectos.

	Ponderación	Valoración Redmine	Calificación Redmine
Criterios funcionales			
Proyectos			
Crear un proyecto	0,8	70	56
Visualizar un proyecto	0,6	60	36
Añadir miembro a un proyecto	0,7	80	56
Formas de ordenar los proyectos	0,6	100	60
Usuario			
Funcionalidades ofrecidas en el perfil	0,8	80	64
Facilidad para contactar con otros usuarios	0,9	60	54
Subir archivos a la web	0,6	80	48
Búsquedas	0,7	90	63
Servicios			
Variedad de los servicios	0,8	100	80
Utilidad de los servicios	0,8	70	56
Criterios no funcionales			
Interfaz amigable	0,9	50	45
Coste	0,8	80	64
Tráfico de usuarios	0,7	70	49
Velocidad de navegación	0,8	80	64
Confianza	1	80	80
TOTAL			875

Tabla 1.4: Modelo de calificación de Redmine.

En la tabla 2.4 se ha evaluado el portal Redmine, el cuál es que mayor variedad de servicios ofrece de los cuatro portales evaluados, también observamos que, en términos generales, tanto la gestión de proyectos como de usuarios son buenas, aunque flaquean un poco en la visualización de los proyectos como en la facilidad para contactar con otros usuarios. Algo parecido pasa con los criterios no funcionales ya que son buenos en general, sin embargo, Redmine nos ofrece una interfaz bastante compleja. Debido a esto Redmine es el tercer portal peor valorado del estudio.

Como se ha podido observar vemos que tanto Trello como Asana serían dos grandes competidores respecto a la aplicación que se está desarrollando en este TFG, y que Redmine y sobre todo Taiga tienen más carencias. Por lo que se va a poner especial hincapié en fortalecer los puntos en los que Trello y Asana tienen menor valoración para que la aplicación de este TFG sea la primera elección para los usuarios que necesiten el uso de una herramienta de gestión ágil de proyectos.

2.2 Catálogo de usuarios

Para la realización de este proyecto se decidió que la aplicación debe contemplar tres grupos de usuarios, cada uno de ellos con varios perfiles de usuario con una serie de privilegios y funciones claramente diferenciadas:

- **Equipo:** dentro del equipo vamos a encontrar dos tipos de perfiles:
 - **Jefe de proyecto (o Scrum Master):** este usuario se encargará de la gestión de los proyectos, de los sprints y de las historias de usuario. Otra de sus funciones será de la de asignar el responsable o responsables a cada una de las tareas que conforman el proyecto, así como ser el encargado de comunicarse con el responsable del cliente. Por último, el jefe de proyecto también tendrá que validar la finalización de cada una de las historias de usuario.
 - **Desarrolladores:** estos usuarios serán los encargados de llevar a cabo todas y cada una de las tareas que les han sido asignadas, así como actualizar el estado de estas según avance el proyecto (algunos de estos estados pueden ser “tarea pendiente”, “tarea en progreso” o “tarea finalizada”).
- **Cliente:** dentro de la organización del cliente encontramos dos tipos de perfiles:
 - **Responsable:** este usuario se considerará como el “representante” del cliente (se puede considerar un rol equivalente al del Product Owner en Scrum), por lo que será el encargado de resolver cualquier duda planteada por los miembros del Equipo. Al igual que el jefe de proyecto, el responsable también deberá validar (aceptar) todas y cada una de las historias de usuario desde el punto de vista del cliente del proyecto.
 - **Equipo del cliente:** estos usuarios se encargarán principalmente de dar soporte y ayuda al responsable.
- **Administrador:** este usuario, además de poder realizar todo lo mencionado anteriormente, también se encargará de la gestión de los usuarios.

Cada vez que se hace referencia al término “gestión” nos referiremos a la creación, consulta, modificación y eliminación de un elemento. A continuación, se describirá las principales funcionalidades de la aplicación.

2.3 Descripción general del sistema

Una vez comentado el motivo que da origen a este TFG y el funcionamiento del método ágil SCRUM, se va a detallar el funcionamiento de la aplicación web. La idea principal es dar la oportunidad a los clientes a encontrar un equipo de desarrollo que pueda llevar a cabo un proyecto que necesiten, y que el equipo de desarrollo tenga un lugar donde poder realizar y hacer el seguimiento de dicho proyecto.

La aplicación tendrá un acceso exclusivo, es decir, solo se podrá acceder si has recibido una invitación por parte del administrador de la aplicación. Una vez dentro de la web, los miembros del equipo podrán acceder a todos los proyectos a los que han sido asignados, tanto proyectos que ya hayan finalizado como proyectos que aún estén en curso. Dentro de cada proyecto podrá consultar los sprints en los que ha sido dividido, así como, cada una de las tareas a realizar en cada sprint. Por otro lado, el cliente podrá acceder a una funcionalidad en la que podrá exponer en detalle un proyecto que desea que sea realizado por un equipo de desarrollo. Tanto los miembros del equipo de desarrollo como los clientes tendrán acceso a su perfil dónde se les permitirá consultar o modificar cualquiera de sus datos personales, al igual que modificar su usuario y contraseña de acceso a la web. Por último, todos los usuarios de la aplicación contarán con un servicio de mensajería y notificaciones con el que se podrán poner en contacto con cualquier otro usuario de la aplicación.

Funcionalidades propias del administrador:

Como se ha indicado anteriormente, la aplicación contará con un usuario con perfil de administrador, que será el encargado de llevar a cabo toda la gestión de los usuarios, por lo que podrá crear, consultar, modificar o eliminar un usuario. Una vez haya creado un usuario, el administrador también podrá enviar una invitación a este usuario para que pueda acceder a la aplicación. Esta invitación contendrá un nombre de usuario y una contraseña con la que acceder.

Funcionalidades propias de los desarrolladores:

Una vez hayan recibido la invitación del administrador, los usuarios podrán acceder a la aplicación. Dentro de la aplicación, un desarrollador puede ver sus datos de perfil, así como modificarlos o cambiar su contraseña de acceso.

Además, podrá acceder al listado de proyectos de los que ha formado parte, de igual modo podrá acceder a los proyectos que estén en activo, donde podrá ver toda la información respecto a este proyecto, los sprints que lo componen o cada una de las tareas que conforman cada sprint. Dentro de las historias de usuario, los desarrolladores podrán modificar el estado en el que se encuentra la historia de usuario, realizar cualquier consulta necesaria sobre la historia de usuario, así como la subida de archivos y maquetas a cada una de las historias de usuario, a la vez que podrá consultar un log donde se mantendrá un historial de todos los cambios que se han producido sobre una historia de usuario.

Por último, un desarrollador tendrá acceso a un panel de notificaciones y mensajería en el que podrá, entre otras cosas, estar al corriente de cualquier modificación que haya sufrido un proyecto, sprint o historia de usuario en el que esté involucrado, o cualquier mensaje que haya recibido, el cuál desde este panel, lo podrá responder.

Funcionalidades referidas al jefe del equipo:

El jefe del equipo tendrá acceso a toda la funcionalidad descrita en el apartado anterior. Además, también se encargará de la gestión de los proyectos, sprints e historias de usuario, por lo que podrá crear, consultar, modificar o eliminar cualquiera de estos tres elementos. Una vez creada una historia de usuario, el jefe de proyecto deberá asignar uno o varios responsables a esta historia de usuario. Junto a cada una de las historias de usuario aparecerá un botón de validación, el cual podrá pulsar si la historia de usuario se ha finalizado correctamente, desde el punto de vista del equipo de desarrollo.

El jefe de proyecto también podrá dar por finalizado un proyecto una vez se hayan completado todas y cada una de las historias de usuario que formaban parte del proyecto.

Funcionalidades referidas al equipo del cliente:

Estos usuarios accederán a la aplicación con funcionalidades únicamente de consulta de información, es decir, solo podrán acceder a los proyectos que tienen asignados y ver cómo está siendo el desarrollo de estos, pero no podrá modificar nada.

Al igual, que los desarrolladores, los miembros del equipo también podrán acceder a los datos de su perfil, así como modificarlos o cambiar su contraseña de acceso. De igual modo, también tendrán acceso al panel de notificaciones y mensajería.

Funcionalidades referidas al responsable del cliente:

El responsable del cliente tendrá acceso a las mismas funcionalidades que los miembros del equipo del cliente, con la principal diferencia, que este usuario también tendrá la posibilidad de validar las historias de usuario, como lo hace el jefe de proyecto. La idea es que mediante la validación del responsable del cliente dé el visto bueno a la finalización de cada una de las tareas (historias de usuario) que componen el proyecto.

3 Análisis

3.1 Catálogo de requisitos

Podemos definir el catálogo de requisitos como un listado de las necesidades sobre el contenido o funcionalidad de un sistema. [2]

Dentro de los requisitos podemos encontrar tres tipos bien definidos. En primer lugar, tenemos los requisitos funcionales, que se entienden cada una de las acciones o tareas que debe implementar un sistema. [3]

A continuación, se presentan los requisitos funcionales de este TFG:

Identificador	Título	Descripción
RF-01	CRUD Usuarios	A los usuarios de la web se les asignará un rol (Jefe de proyecto o Desarrollador si pertenece a un Equipo o bien Responsable o miembro del equipo del cliente si es un Cliente). Los usuarios con el rol de jefe de proyecto podrán agregar a nuevos participantes a sus proyectos mediante el envío de invitaciones a través de su aplicación.
RF-02	CRUD Proyectos	Los jefes de proyectos podrán crear proyectos y gestionar la información de estos.
RF-03	CRUD Sprints	Cada proyecto se dividirá en etapas, dentro de las cuales estarán asignadas una serie de tareas que habrá que realizar dentro del tiempo que dure el sprint.
RF-04	CRUD Historias de Usuario	Todos los proyectos creados contarán con una serie de historias de usuario.
RF-05	Validación de Historias de Usuario	Las historias de usuario deberán ser validadas por el jefe de proyecto, así como por el responsable del cliente.
RF-06	Asignación de Historias de Usuario	El jefe de proyecto asignará cada una de las historias de usuario a uno o varios miembros del equipo de desarrolladores.

RF-07	Adjuntar archivo a Historia de Usuario	Los usuarios podrán subir un archivo (documentación o funcionalidades concretas) que se asociará a una historia de usuario.
RF-08	Adjuntar maquetas a Historia de Usuario	Al igual que en el punto anterior, con la diferencia que esta se centrará exclusivamente en la subida de maquetas. Los archivos se subirán en formato .jpg o .pdf para que pueda ser visualizados en la web.
RF-9	Cronograma	Se creará un cronograma donde se podrá visualizar el tiempo restante para finalizar una tarea.
RF-10	Chat de mensajería	Se implementará un chat donde se podrán poner en contacto los desarrolladores con los clientes.
RF-11	Log de Historia de Usuario	Se mantendrá un registro de cada una de las actualizaciones realizadas sobre cada una de las historias de usuario. Dicho registro podrá ser consultado por los usuarios participantes en cada historia de usuario.

También encontramos los requisitos no funcionales, los cuales los podemos definir como aquellos requisitos que sirven para especificar condiciones, características o capacidades que debe satisfacer un sistema. [4].

Los requisitos no funcionales de este TFG son los siguientes:

Identificador	Título	Descripción
RNF-01	Velocidad de la web	La página web estará implementada de forma que ofrezca el mejor rendimiento posible ante cualquier tipo de situación. Por ejemplo, todos los dispositivos podrán cargar cualquier página de la aplicación en menos de 10 segundos.
RNF-02	Diseño adaptativo	La aplicación web estará implementada de forma que se adapte correctamente a las dimensiones de la pantalla del dispositivo desde el que se acceda.
RNF-03	Compatibilidad	La web funcionará en un amplio catálogo de navegadores web. Algunos de ellos son Google Chrome, Mozilla Firefox, Opera o Edge.
RNF-04	Número de usuarios	La aplicación web estará diseñada para soportar al menos 200 usuarios de forma simultánea.

RNF-05	Interfaz amigable	La interfaz de usuario será muy fácil de usar para así mejorar la experiencia de uso de los usuarios. Por ejemplo, se podrá acceder a cualquier página de la aplicación en menos de cuatro clicks, así mismo, todas las páginas de la aplicación contarán con el mismo diseño.
--------	-------------------	--

Por último, encontramos los requisitos de información, que describen las principales entidades de información que gestiona y almacena un sistema para dar soporte a las operaciones realizadas por el propio sistema. [5]

Los requisitos de información de este TFG se presentan a continuación:

Identificador	Título	Descripción
RI-01	Usuarios	Nombre, Apellidos, Contraseña, Email, Rol, Fecha_alta, Fecha_baja
RI-02	Proyectos	Nombre, Descripción, Usuarios_participantes
RI-03	Sprints	Id_sprint, Proyecto
RI-04	Historias de Usuarios	Nombres, Descripción, Estado, Tiempo_estimado, Tiempo_employado, Responsable, Adjuntos, Comentarios, Usuarios_participantes
RI-05	Comentarios	Título, Texto, Autor, Fecha_envío, Historias_de_usuario

3.2 Casos de uso

Podríamos definir los casos de uso como una secuencia de acciones realizada por uno o varios actores que dan como resultado un valor observable [6].

En nuestro proyecto se han englobado cada uno de los casos de uso según el tipo de usuario, siendo los tipos: Administrador, Jefe de proyecto, Desarrolladores, Responsable del cliente y Equipo del cliente.

Antes de pasar a describir los casos de uso relativos a cada uno de los actores, cabe destacar que previamente cada usuario deberá haber iniciado la sesión.

En primer lugar, pasaremos a describir los casos de uso relacionados con el Jefe de proyecto. Este actor es el que más casos de uso tiene, ya que entre otros destaca el CRUD Proyectos, CRUD Sprint, CRUD Historias de usuario o Validación de historia de usuario.

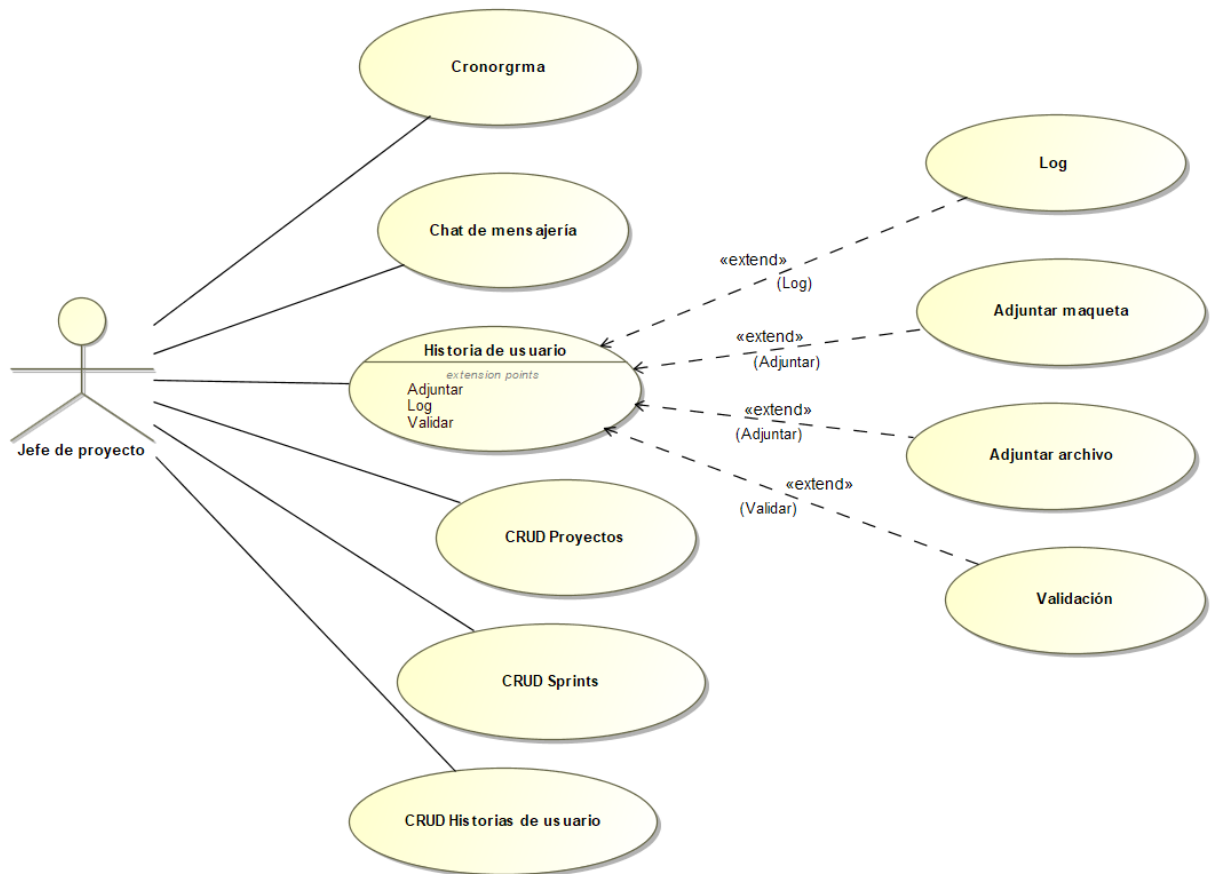


Figura 3.1: Diagrama de casos de uso para el usuario con rol Jefe de proyecto.

A continuación, describiremos los casos de uso de los Desarrolladores, los cuales tienen menos casos que el Jefe de proyecto. Los casos de uso que engloba son Adjuntar archivo a historia de usuario, Adjuntar maqueta a historia de usuario y Log de historia de usuario. El jefe de proyecto, como se ha visto en la Figura 3.1, también está relacionado con estos casos de uso.

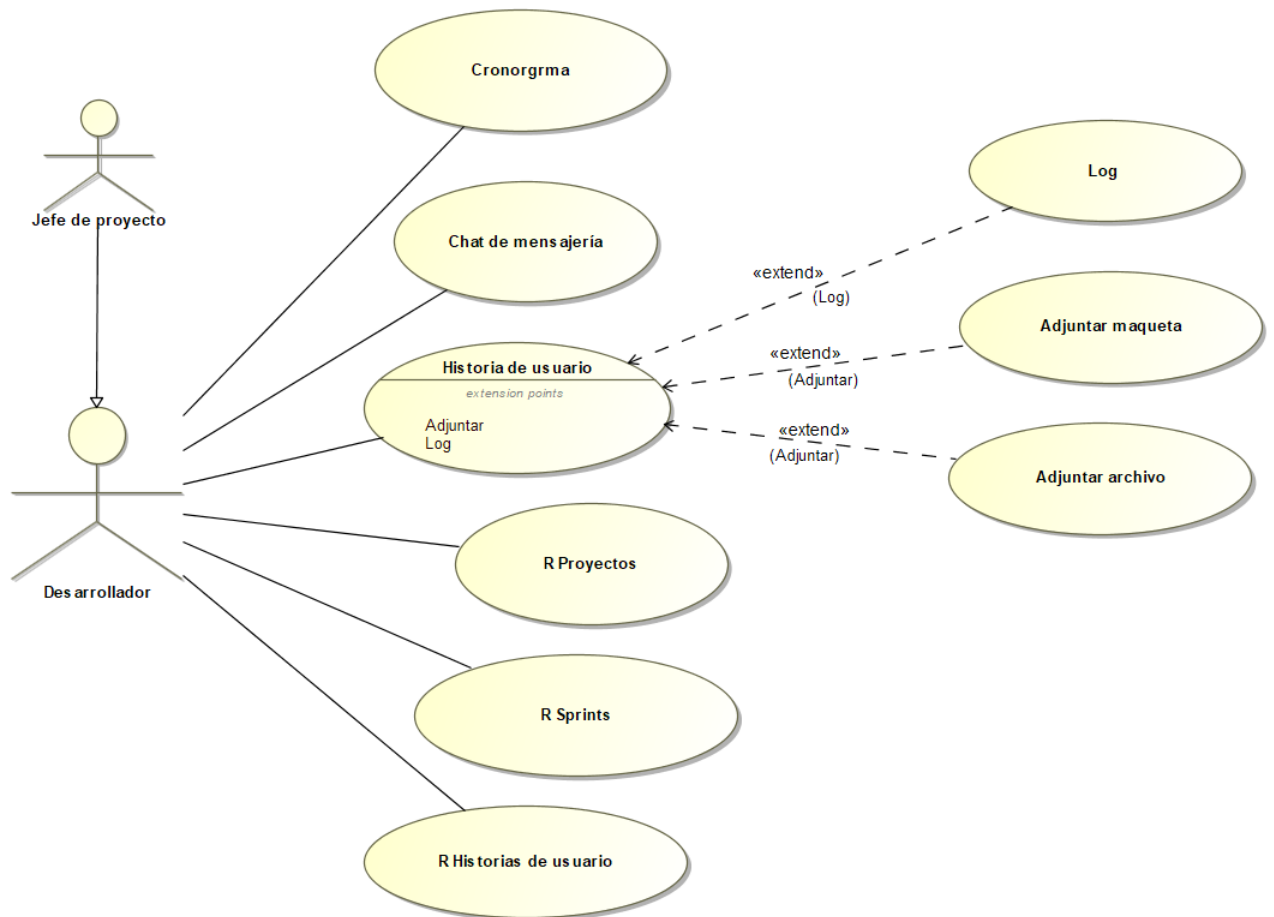


Figura 3.2: Diagrama de casos de uso para el usuario con rol Desarrollador.

Por último, los actores del lado del cliente tienen prácticamente los mismos casos de uso con la diferencia que el Responsable del cliente se relaciona con el caso de uso Validación de historia de usuario, además de los casos de uso Chat de mensajería y Cronograma que son los que relacionados con el Equipo del cliente.

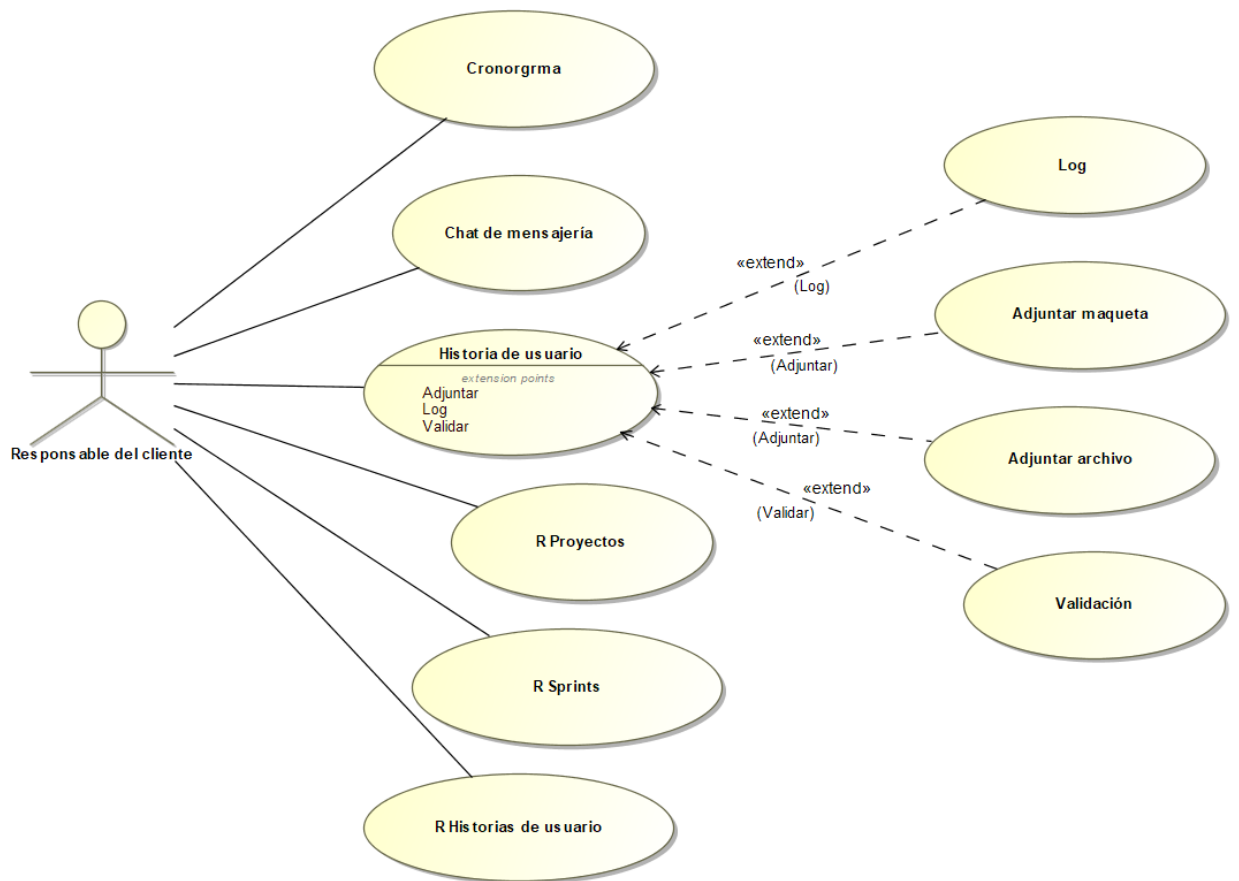


Figura 3.3: Diagrama de casos de uso para el usuario con rol Responsable del cliente.

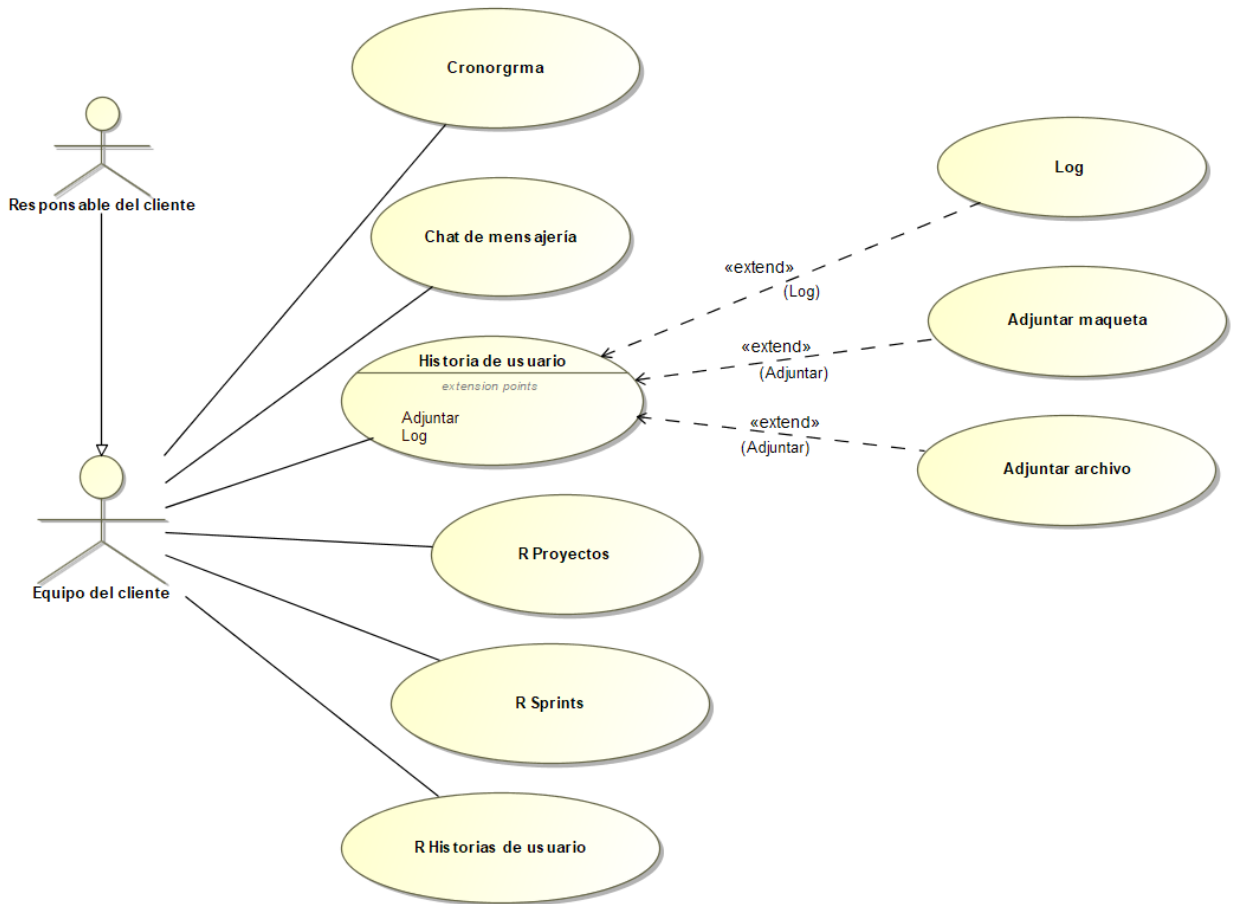


Figura 3.4: Diagrama de casos de uso para el usuario con rol Equipo del cliente.

El actor Administrador engloba todos y cada uno de los casos de uso mostrados anteriormente, a la vez que tiene como caso de uso exclusivo CRUD Usuarios

3.3 Especificación de los casos de uso

Respecto a la especificación de los casos de uso, en cada caso se va a estudiar una descripción resumida del caso de uso, una precondición que debe ser cumplida para poder ejecutar el caso de uso, unos escenarios base y alternativos y una postcondición, que reflejará el resultado obtenido después de ejecutar con éxito el escenario base.

Título	Validar acceso
Descripción	El usuario introduce un nombre de usuario y una contraseña para acceder a la aplicación web.
Precondición	El usuario deberá haber sido invitado previamente a la aplicación web.

Escenario base	<ol style="list-style-type: none"> 1. El usuario accede a la aplicación web. 2. El sistema muestra en pantalla la venta de "Login". 3. El usuario introduce el nombre de usuario y contraseña. 4. El sistema valida los datos introducidos por el usuario. 5. Se cierra la ventana de Login y se muestra la página principal.
Escenario alternativo	<p>Escenario alternativo 1:</p> <ol style="list-style-type: none"> 1. El nombre de usuario y/o contraseña introducidos son incorrectos. 2. El sistema envía un mensaje de error. <p>Escenario alternativo 2:</p> <ol style="list-style-type: none"> 1. El usuario pulsa el botón "Iniciar sesión" sin haber rellenado alguno de los campos. 2. El sistema envía un mensaje de error. <p>Escenario alternativo 3:</p> <ol style="list-style-type: none"> 1. El usuario intenta iniciar sesión sin éxito más veces de las permitidas. 2. El sistema bloquea el inicio de sesión durante 15 minutos.
Postcondición	En el escenario base el usuario habrá podido acceder al sistema, mientras que en el sistema habrá quedado registrado el usuario que iniciado sesión en la aplicación web.

Título	CRUD Usuarios
Descripción	Cualquier usuario dado de alta en la aplicación podrá acceder al botón del perfil situado en la esquina superior derecha, mientras que la creación de usuarios correrá a cargo del administrador del sistema.
Precondición	Solo el administrador del sistema podrá crear y dar de alta a nuevos usuarios (C).
Escenario base	<p>Escenario base (Create):</p> <ol style="list-style-type: none"> 1. El administrador pulsa el botón "Nuevo usuario". 2. El sistema muestra la vista correspondiente. 3. El administrador rellena los campos, marca la casilla invitar usuario y pulsa el botón "Crear".

	<p>4. El sistema envía la invitación al correo introducido por el administrador y registra en la base de datos al usuario que se acaba de crear.</p> <p>Escenario base (Read):</p> <ol style="list-style-type: none"> 1. Un usuario pulsa el botón “Perfil”. 2. El sistema muestra la venta asociada al perfil del usuario. 3. El usuario puede consultar todos sus datos personales como nombre, apellidos o correo electrónico. <p>Escenario base (Update):</p> <ol style="list-style-type: none"> 1. El usuario, una vez dentro del perfil, pulsa el botón “Actualizar perfil”. 2. El sistema muestra la ventana donde el usuario podrá modificar sus datos. 3. El usuario modifica los datos que desee cambiar y pulsa el botón “Guardar”. 4. El sistema muestra un mensaje de confirmación y registra los cambios en la base de datos. <p>Escenario base (Delete):</p> <ol style="list-style-type: none"> 1. El usuario, una vez dentro del perfil, pulsa el botón de “Eliminar usuario”. 2. El sistema muestra un mensaje de confirmación 3. El usuario pulsa el botón “Aceptar”. 4. El sistema eliminar el usuario de la base de datos y muestra la ventana de Login.
<p>Escenario alternativo</p>	<p>Escenario alternativo (Create):</p> <ol style="list-style-type: none"> 1. El administrador introduce un correo existente dentro de la aplicación. 2. El sistema envía un mensaje informando de dicho conflicto. <p>Escenario alternativo (Update):</p> <ol style="list-style-type: none"> 1. El usuario cambia su correo electrónico y el nuevo que introduce coincide con uno ya registrado. 2. El sistema envía un mensaje informando de dicho conflicto. <p>Escenario alternativo (Delete):</p> <ol style="list-style-type: none"> 1. El usuario, una vez el sistema muestra el mensaje de confirmación para el eliminar el usuario, pulsa el botón “Cancelar”. 2. El sistema muestra la pantalla del perfil.

Postcondición	El administrador habrá podido crear e invitar a la aplicación a un nuevo usuario, mientras que cualquier usuario habrá accedido a su perfil para consultar o modificar sus datos o eliminar su usuario.
----------------------	---

Título	CRUD Proyectos
Descripción	Cualquier usuario registrado en la aplicación web podrá acceder a la información relacionada con un proyecto del que el usuario forme parte, mientras que las funciones de crear, modificar y eliminar proyectos estarán reservadas para el jefe de proyectos.
Precondición	El usuario deberá haber sido invitado previamente a la aplicación web
Escenario base	<p>Escenario base (Create):</p> <ol style="list-style-type: none"> 1. El jefe de proyectos pulsa el botón “Nuevo proyecto”. 2. El sistema muestra la ventana de creación de proyectos. 3. El jefe de proyecto rellena los campos solicitados. 4. El sistema crea el proyecto en la base de datos y muestra la ventana asociada al proyecto que se acaba de crear. <p>Escenario base (Read):</p> <ol style="list-style-type: none"> 1. El usuario pulsa el botón “Mis proyectos”. 2. El sistema muestra una ventana donde aparecen todos los proyectos de los que forma parte el usuario. 3. El usuario pulsa sobre uno de esos proyectos. 4. El sistema muestra toda la información de ese proyecto. <p>Escenario base (Update):</p> <ol style="list-style-type: none"> 1. El jefe de proyecto pulsa el botón “Editar proyecto”. 2. El sistema muestra la ventana de editar un proyecto. 3. El jefe de proyecto modifica los datos necesarios. 4. El sistema guarda los cambios y muestra la ventana Mis proyectos. <p>Escenario base (Delete):</p> <ol style="list-style-type: none"> 1. El jefe de proyecto pulsa el botón “Eliminar proyecto”. 2. El sistema muestra un mensaje de confirmación. 3. El jefe de proyecto pulsa el botón “Aceptar”.

	4. El sistema eliminar el proyecto de la base de datos y muestra la ventana Mis proyectos.
Escenario alternativo	<p>Escenario alternativo (Create y Update):</p> <ol style="list-style-type: none"> 1. El jefe de proyecto pulsa el botón "X". 2. El sistema muestra la ventana Mis proyectos. <p>Escenario alternativo (Delete):</p> <ol style="list-style-type: none"> 1. El jefe de proyecto pulsa el botón "Cancelar". 2. El sistema muestra la ventana Mis proyectos.
Postcondición	El jefe de proyectos habrá podido crear, modificar o eliminar cualquier proyecto, a la vez que cualquier usuario habrá podido acceder a la información asociada a cualquier proyecto del que forme parte.

Título	CRUD Sprints
Descripción	El jefe de proyecto divide un proyecto en varias etapas o fases, conocidas como sprints. La creación, modificación y eliminación de los sprints correrán a cargo del jefe de proyecto, mientras que la visualización la podrá realizar cualquier usuario de la aplicación.
Precondición	El usuario deberá haber sido invitado previamente a la aplicación web, además se deberá haber creado un proyecto previamente.
Escenario base	<p>Escenario base (Create):</p> <ol style="list-style-type: none"> 1. El jefe de proyectos pulsa el botón "Gestionar sprints". 2. El sistema muestra la ventana. 3. El jefe de proyecto pulsa el botón "Nuevo sprint". 4. El sistema muestra la ventana asociada a la creación de sprints. 5. El jefe de proyecto rellena los campos necesarios y pulsa el botón "Crear". 6. El sistema guarda en la base de datos el sprint creado y vuelve a la ventana Gestionar sprints. <p>Escenario base (Read):</p> <ol style="list-style-type: none"> 1. Un usuario pulsa el botón "Gestionar sprints". 2. El sistema muestra la ventana de los sprints. 3. El usuario pulsa sobre uno de los sprints que aparecen. 4. El sistema muestra la información asociada al sprint pulsado.

	<p>Escenario base (Update):</p> <ol style="list-style-type: none"> 1. El jefe de proyecto pulsa el botón "Gestionar sprints". 2. El sistema muestra la ventana de sprints. 3. El jefe de proyecto pulsa sobre uno de los sprints que aparecen. 4. El sistema muestra el sprint seleccionado. 5. El jefe de proyecto pulsa el botón "Editar sprint". 6. El sistema muestra una ventana con los datos del sprint. 7. El jefe de proyecto modifica los datos del sprint. 8. El sistema guarda los cambios realizados y vuelve a la ventana Gestionar sprints. <p>Escenario base (Delete):</p> <ol style="list-style-type: none"> 1. El jefe de proyecto pulsa el botón "Gestionar sprints". 2. El sistema muestra la ventana de sprints. 3. El jefe de proyecto pulsa sobre uno de los sprints que aparecen. 4. El sistema muestra el sprint seleccionado. 5. El jefe de proyecto pulsa el botón "Eliminar sprint". 6. El sistema muestra un mensaje de confirmación. 7. El jefe de proyecto pulsa el botón "Aceptar". 8. El sistema elimina el sprint de la base de datos y vuelve a la ventana Gestionar sprints.
Escenario alternativo	<p>Escenario alternativo (Create y Update):</p> <ol style="list-style-type: none"> 1. El jefe de proyecto pulsa el botón "X". 2. El sistema muestra la ventana Gestionar sprints. <p>Escenario alternativo (Delete):</p> <ol style="list-style-type: none"> 1. El jefe de proyecto pulsa el botón "Cancelar". 2. El sistema muestra la ventana Gestionar sprints.
Postcondición	<p>El jefe de proyectos habrá podido crear, modificar o eliminar sprints asociados a un proyecto, a la vez que cualquier usuario habrá podido acceder a la información asociada a cualquier sprint de un proyecto del que forme parte.</p>

Título	CRUD Historias de usuario
Descripción	<p>El jefe de proyecto crea historias de usuario o tareas repartidas en los diferentes sprints que componen un proyecto. La creación, modificación y eliminación de estas historias de usuario correrán a cargo del jefe de proyectos, mientras que la visualización la podrá realizar cualquier usuario que forme parte del proyecto.</p>

Precondición	El usuario deberá haber sido invitado previamente a la aplicación web y deberá existir tanto un proyecto como sprints creados anteriormente.
Escenario base	<p>Escenario base (Create):</p> <ol style="list-style-type: none"> 1. El jefe de proyecto accede a la ventana Gestionar sprints. 2. El sistema muestra la ventana solicitada. 3. El jefe de proyectos accede a uno de los sprints mostrados. 4. El sistema muestra el sprint seleccionado. 5. El jefe de proyecto pulsa el botón "Historia de usuario". 6. El sistema muestra la ventana para crear una nueva historia de usuario. 7. El jefe de proyecto rellena los datos para crear la historia de usuario y la asigna un/os responsable/s. 8. El sistema guarda los cambios y vuelve a la ventana Gestionar sprints. <p>Escenario base (Read):</p> <ol style="list-style-type: none"> 1. Un usuario accede a la ventana Gestionar sprints. 2. El sistema muestra la ventana solicitada. 3. Un usuario accede a uno de los sprints mostrados. 4. El sistema muestra el sprint seleccionado. 5. Un usuario comprueba toda la información referida a las historias de usuario que conforman el sprint seleccionado. <p>Escenario base (Update):</p> <ol style="list-style-type: none"> 1. El jefe de proyecto accede a la ventana Gestionar sprints. 2. El sistema muestra la ventana solicitada. 3. El jefe de proyectos accede a uno de los sprints mostrados. 4. El sistema muestra el sprint seleccionado. 5. El jefe de proyecto pulsa el botón "Historia de usuario". 6. El sistema muestra la ventana de las historias de usuario. 7. El jefe de proyecto modifica los datos necesarios. 8. El sistema guarda los cambios realizados y muestra la ventana Gestionar sprints. <p>Escenario base (Delete):</p> <ol style="list-style-type: none"> 1. El jefe de proyecto accede a la ventana Gestionar sprints. 2. El sistema muestra la ventana solicitada. 3. El jefe de proyectos accede a uno de los sprints mostrados. 4. El sistema muestra el sprint seleccionado.

	<ol style="list-style-type: none"> 5. El jefe de proyecto pulsa el botón "Eliminar". 6. El sistema muestra un mensaje de confirmación. 7. El jefe de proyecto pulsa el botón "Aceptar". 8. El sistema elimina la historia de usuario y muestra la ventana Gestionar sprints.
Escenario alternativo	<p>Escenario alternativo (Create y Update):</p> <ol style="list-style-type: none"> 1. El jefe de proyecto pulsa el botón "X". 2. El sistema muestra la ventana Gestionar sprints. <p>Escenario alternativo (Delete):</p> <ol style="list-style-type: none"> 1. El jefe de proyectos pulsa el botón "Cancelar". 2. El sistema muestra la ventana Historia de usuario.
Postcondición	<p>El jefe de proyectos habrá podido crear, modificar o eliminar historias de usuario asociadas a un sprint, a la vez que cualquier usuario habrá podido acceder a la información asociada a cualquier historia de un sprint de un proyecto del que forme parte.</p>

Título	Validación de historias de usuario
Descripción	El jefe de proyecto y el responsable del cliente deberán validar las historias de usuario del proyecto para que así quede validado el proyecto.
Precondición	Tanto el jefe de proyectos como el responsable del cliente deberán tener sus sesiones iniciadas.
Escenario base	<ol style="list-style-type: none"> 1. El jefe de proyecto y el responsable del cliente pulsa el botón "Mis proyectos". 2. El sistema muestra la ventana. 3. El jefe de proyecto y el responsable del cliente pulsan sobre uno de los proyectos. 4. El sistema muestra el proyecto seleccionado. 5. El jefe de proyecto y el responsable del cliente pulsan el botón "Validar historia de usuario". 6. El sistema marca la historia de usuario como validada y muestra la ventana Mis proyectos.
Escenario alternativo	Para este caso de uso no se ha considerado un escenario alternativo.

Postcondición	En el escenario base tanto el jefe de proyectos como el responsable del cliente habrán podido validar una historia de usuario perteneciente a un proyecto, de forma que en el log de historias de usuario quedará reflejada esta validación.
Título	Asignación de historias de usuario.
Descripción	El jefe de proyecto se encargará de asignar uno o varios miembros del equipo de desarrollo como responsables de realizar una historia de usuario concreta.
Precondición	El jefe de proyectos deberá tener la sesión iniciada. Así mismo, la asignación de responsables la realizará el jefe de proyectos.
Escenario base	<ol style="list-style-type: none"> 1. El jefe de proyecto pulsa el botón "Gestionar sprints". 2. El sistema muestra la ventana. 3. El jefe de proyecto pulsa sobre uno de los sprints mostrados. 4. El sistema muestra el sprint seleccionado. 5. El jefe de proyecto pulsa el botón "Asignar responsable". 6. El sistema muestra la ventana. 7. El jefe de proyecto selecciona los miembros del equipo y pulsa el botón "Aceptar". 8. El sistema guarda los cambios y muestra la ventana Gestionar sprints.
Escenario alternativo	<p>Escenario alternativo 1:</p> <ol style="list-style-type: none"> 1. El jefe de proyectos, en la ventana "Asignar responsable", pulsa el botón "Cancelar". 2. El sistema vuelve a la ventana Gestionar sprints. <p>Escenario alternativo 2:</p> <ol style="list-style-type: none"> 1. El jefe de proyectos asigna como responsable de la historia de usuario a un desarrollador que no forma parte del proyecto. 2. El sistema muestra un mensaje de error.
Postcondición	En el escenario base, el jefe de proyectos habrá podido asignar responsables una historia de usuario perteneciente a un proyecto, de forma que la asignación quedará reflejada en el log de historias de usuario.

Título	Adjuntar archivo a historias de usuario.
Descripción	Cualquier miembro del equipo de desarrollo que sea responsable de una historia de usuario podrá subir archivos a la historia de usuario de la cuál es responsable.
Precondición	El miembro del equipo de desarrollo deberá tener la sesión iniciada, además de estar asignado como responsable de una historia de usuario.
Escenario base	<ol style="list-style-type: none"> 1. El miembro del equipo de desarrollo pulsa el botón “Mis proyectos”. 2. El sistema muestra los proyectos de los que forma parte. 3. El miembro del equipo de desarrollo pulsa el botón “Gestionar sprints”. 4. El miembro del equipo de desarrollo selecciona unos de los sprints. 5. El sistema muestra el sprint. 6. El miembro del equipo de desarrollo selecciona una de las historias de usuario. 7. El sistema muestra la historia de usuario. 8. El miembro del equipo de desarrollo pulsa el botón “Adjuntar archivos”. 9. El sistema abre una ventana emergente para seleccionar un archivo. 10. El miembro del equipo de desarrollo selecciona los archivos y pulsa el botón “Subir”. 11. El sistema adjunta los archivos seleccionados y muestra la ventana de la historia de usuario.
Escenario alternativo	<p>Escenario alternativo 1:</p> <ol style="list-style-type: none"> 1. El miembro del equipo de desarrollo selecciona un archivo con un formato no permitido. 2. El sistema muestra un mensaje de error. <p>Escenario alternativo 2:</p> <ol style="list-style-type: none"> 1. El miembro del equipo de desarrollo pulsa el botón “Cancelar”, en el proceso de selección de archivos. 2. El sistema muestra la ventana de la historia de usuario.
Postcondición	En el escenario base, el miembro del equipo de desarrollo habrá podido subir a la aplicación web una serie de archivos a una historia de usuario de la que él es el responsable. El archivo subido podrá ser visto desde el log de historias de usuario.

Título	Adjuntar maquetas a historias de usuario.
Descripción	Cualquier miembro del equipo de desarrollo que sea responsable de una historia de usuario podrá subir maquetas a la historia de usuario de la cuál es responsable.
Precondición	El miembro del equipo de desarrollo deberá tener la sesión iniciada, además de estar asignado como responsable de una historia de usuario.
Escenario base	<ol style="list-style-type: none"> 1. El miembro del equipo de desarrollo pulsa el botón “Mis proyectos”. 2. El sistema muestra los proyectos de los que forma parte. 3. El miembro del equipo de desarrollo pulsa el botón “Gestionar sprints”. 4. El miembro del equipo de desarrollo selecciona unos de los sprints. 5. El sistema muestra el sprint. 6. El miembro del equipo de desarrollo selecciona una de las historias de usuario. 7. El sistema muestra la historia de usuario. 8. El miembro del equipo de desarrollo pulsa el botón “Adjuntar maquetas”. 9. El sistema abre una ventana emergente para seleccionar una maqueta. 10. El miembro del equipo de desarrollo selecciona las maquetas y pulsa el botón “Subir”. 11. El sistema adjunta las maquetas seleccionadas y muestra la ventana de la historia de usuario.
Escenario alternativo	<p>Escenario alternativo 1:</p> <ol style="list-style-type: none"> 3. El miembro del equipo de desarrollo selecciona una maqueta con un formato no permitido. 4. El sistema muestra un mensaje de error. <p>Escenario alternativo 2:</p> <ol style="list-style-type: none"> 3. El miembro del equipo de desarrollo pulsa el botón “Cancelar”, en el proceso de selección de maquetas. 4. El sistema muestra la ventana de la historia de usuario.
Postcondición	En el escenario base, el miembro del equipo de desarrollo habrá podido subir a la aplicación web una serie de maquetas a una historia de usuario de la que él es el responsable. La maqueta subida se podrá ver desde el log de historias de usuario.

Título	Log de historias de usuario.
Descripción	Cualquier cambio realizado sobre una historia de usuario quedará guardado en la aplicación web y todos los usuarios asociados a la historia de usuario podrán ver dichos cambios.
Precondición	Deberá existir una historia de usuario perteneciente a un sprint dentro de un proyecto.
Escenario base	<ol style="list-style-type: none"> 1. Un usuario accede al log para ver los cambios realizados sobre las historias de usuario de un proyecto. 2. Un usuario realiza una acción sobre una historia de usuario perteneciente a un proyecto. 3. El sistema registra esta modificación y la presenta en forma de mensaje dentro del log. 4. Las acciones realizadas sobre una tarea se presentarán en el log en orden descendiente, de forma que la primera notificación que aparece sea la más reciente. 5. Cualquier usuario que forme parte del proyecto podrá acceder al log de historias de usuario.
Escenario alternativo	Para este caso no se han considerado escenarios alternativos.
Postcondición	Los cambios realizados sobre una historia de usuario quedarán reflejados en la aplicación web y cualquier usuario responsable de la historia de usuario podrá ver esos cambios.

Título	Cronograma
Descripción	Para cada proyecto existirá un cronograma que será de ayuda para que los usuarios que formen parte del proyecto tengan mayor facilidad para ver los plazos de realización de cada historia de usuario, así como la duración de cada uno de los sprints.
Precondición	Deberá existir un proyecto creado.
Escenario base	<ol style="list-style-type: none"> 1. El usuario pulsa el botón “Mis proyectos”. 2. El sistema muestra la ventana. 3. El usuario selecciona uno de los proyectos.

	<ol style="list-style-type: none"> 4. El sistema muestra el proyecto seleccionado. 5. El usuario pulsa el botón "Cronograma". 6. El sistema muestra la ventana del cronograma seleccionado. 7. El usuario pulsa sobre un día que tenga asignada una historia de usuario. 8. El sistema abre una ventana con toda la información relacionada sobre esa historia de usuario.
Escenario alternativo	Para este caso no se han considerado escenarios alternativos.
Postcondición	Los usuarios que forman parte de un proyecto podrán ver un cronograma asociado a dicho proyecto.

Título	Chat de mensajería
Descripción	Los usuarios tendrán a su disposición un chat de mensajería con el que se podrán comunicar con cualquier otro usuario registrado en la aplicación web.
Precondición	El usuario deberá tener la sesión iniciada.
Escenario base	<ol style="list-style-type: none"> 1. El usuario, estando en la página principal, pulsa el botón "Chat de mensajería". 2. El sistema muestra la ventana del chat. 3. El usuario escribe un mensaje. 4. El sistema guarda dicho mensaje, así como el autor y la fecha de envío. 5. Otro usuario que reciba el mensaje podrá ver el contenido, autor y fecha del mensaje, así como responder.
Escenario alternativo	Para este caso no se han considerado escenarios alternativos.
Postcondición	Los usuarios acceden al chat de mensajería y podrán enviar mensajes a otros usuarios de la aplicación web.

3.4 Diagrama de clases

Se entiende por diagrama de clases, el modelo que recoge la estructura de las clases de objetos que forman el sistema y como interactúan entre ellos.

Dentro de cada clase podemos encontrar una serie de atributos y operaciones que definen a cada una de las clases. Los vínculos entre las clases, mencionados anteriormente, se conocen como relaciones, siendo las más comunes asociación, herencia, agregación, composición y dependencia. [7]

Para la identificación de las clases del proyecto se ha seguido el patrón Modelo-Vista-Controlador, y según este patrón hemos diferenciado entre clases de entidad, clases de control y clases de interfaz.

Para la elaboración de este TFG se ha contado con siete clases de entidad: Usuarios, Equipos, Proyectos, Notificaciones, Comentarios, Sprints e Historias de usuario. Cada una de las clases cuenta con una serie de atributos propios de cada clase, así como varias operaciones distintivas para cada clase. La gran mayoría de las relaciones entre las clases son del tipo asociación, excepto la relación entre las clases Proyectos-Sprints y Sprints-Historias de usuario que son relaciones de composición. A continuación, se muestra cómo queda el diagrama de clases del proyecto con todas sus clases, atributos, operaciones y relaciones, así como el grado de multiplicidad de estas.

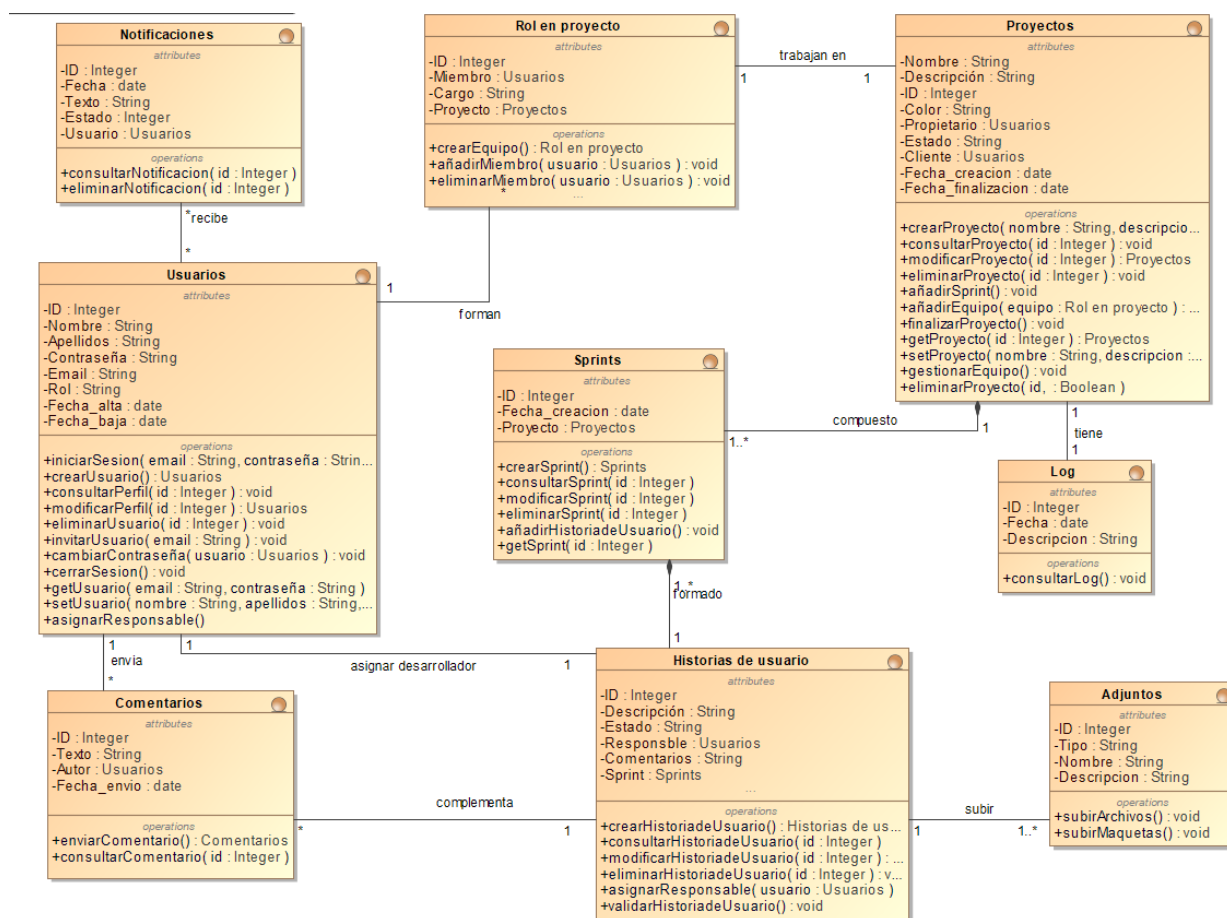


Figura 3.5: Diagrama de clases de entidad del proyecto.

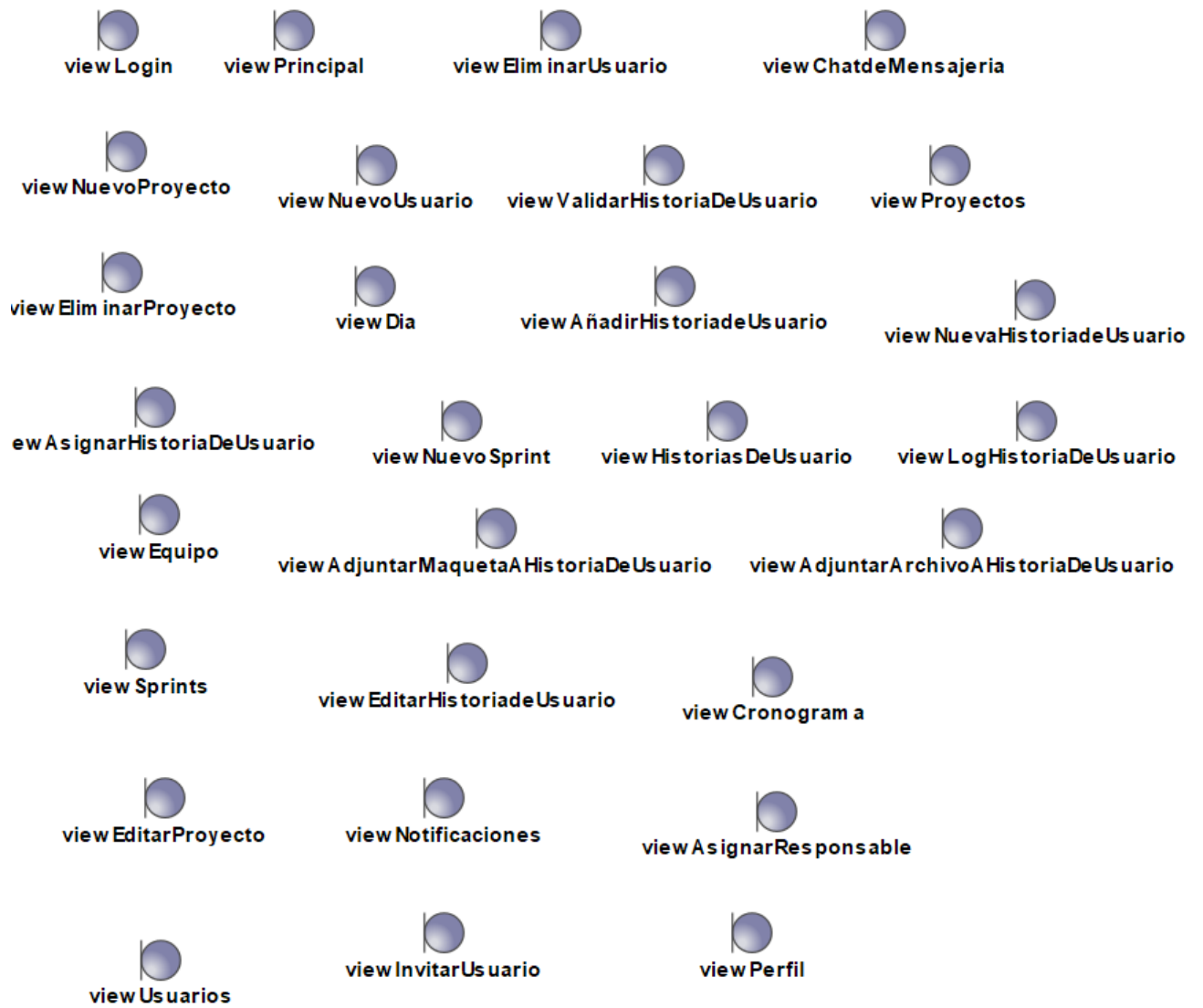


Figura 3.6: Diagrama de clases de interfaz del proyecto.

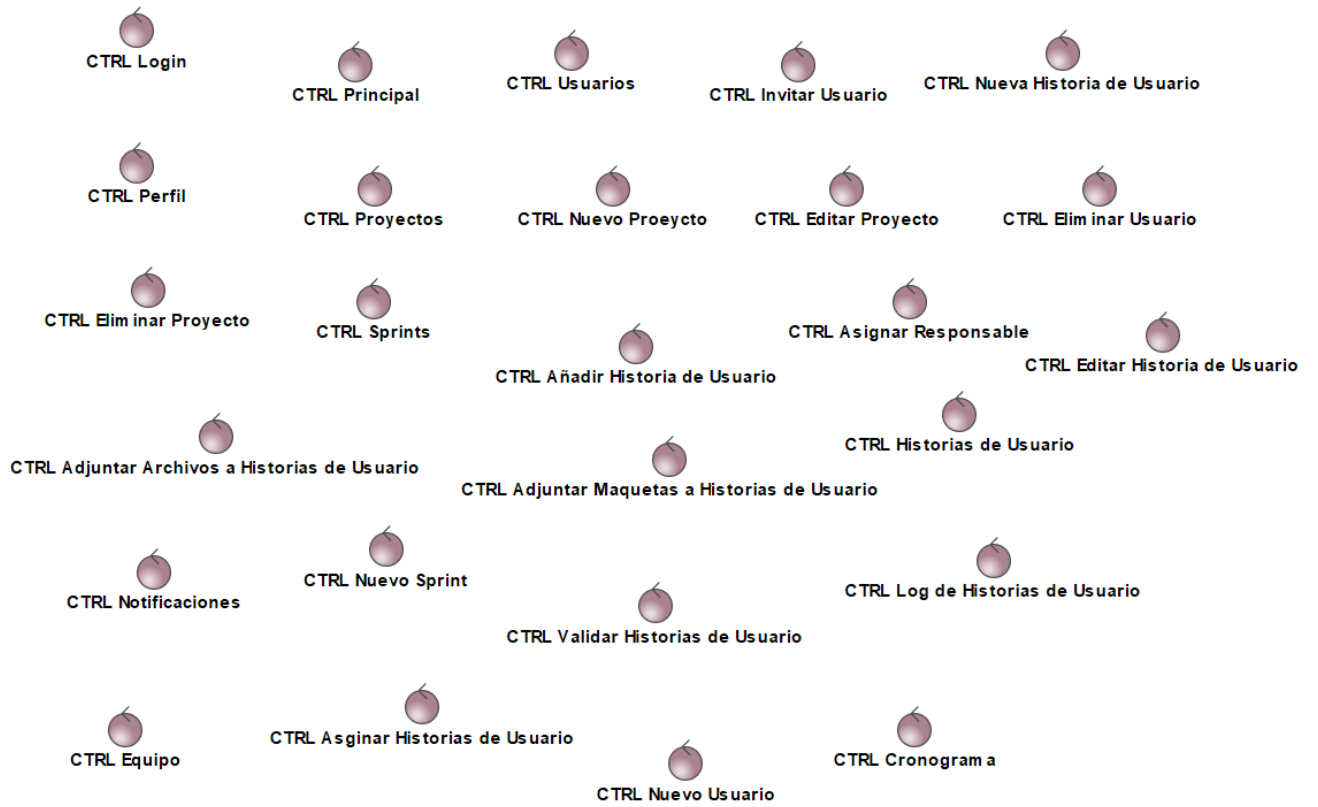


Figura 3.7: Diagrama de clases de control del proyecto.

4 Diseño

4.1 Diagramas de secuencia

Se podría definir los diagramas de secuencia como un tipo de diagrama cuyo principal objetivo es la descripción del funcionamiento de un sistema de información, enfocándose en el intercambio de mensajes por cada uno de los objetos que forman parte del sistema.

Los diagramas de secuencia se componen de dos dimensiones: el eje vertical y el eje horizontal. El eje vertical representa el tiempo mientras que el horizontal los objetos que conforman el sistema. El funcionamiento modelado consiste en que los objetos envían mensajes de los unos a los otros ordenados de arriba abajo, representando esta ordenación el tiempo. Cada uno de los objetos tiene una línea de vida y unos focos de control. La línea de vida representa el tiempo que dicho objeto existe, mientras que los focos de control representan los momentos en el que los objetos se encuentran realizando alguna acción. [8]

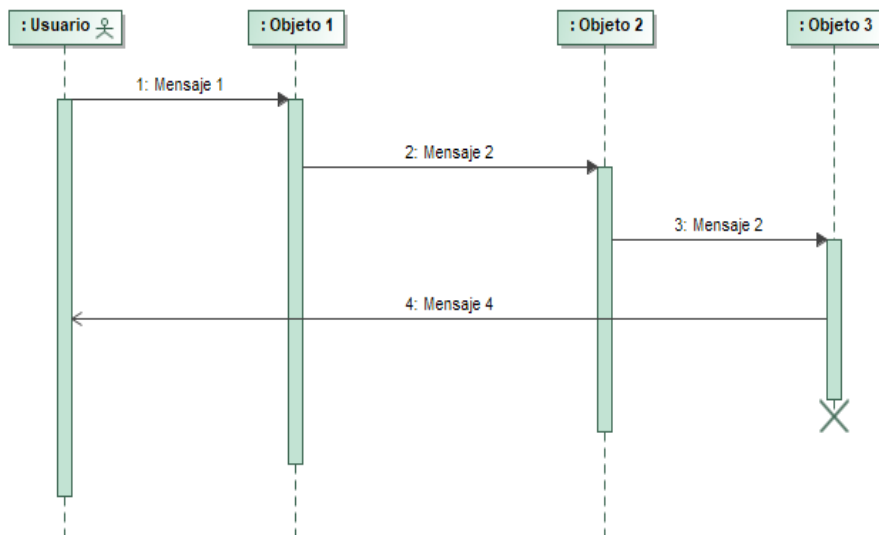


Figura 4.1: Ejemplo de cómo se estructura un diagrama de secuencia

En la Figura 4.1, podemos observar lo que se ha comentado anteriormente. Arriba del todo, en el eje horizontal, aparecen todos los objetos que forman parte del diagrama, en este caso el usuario y los tres objetos, mientras que el eje vertical podemos apreciar el tiempo de los objetos. Cada uno de los mensajes son lo que hemos mencionado antes como focos de control mientras que las barras verticales verdes representan las líneas de

vida de cada objeto. Como podemos ver en la imagen, la línea de vida del objeto 3 se cierra una vez este envía el mensaje 4.

A continuación, se presentan los diagramas de secuencia relacionados con el caso de uso CRUD Usuarios.

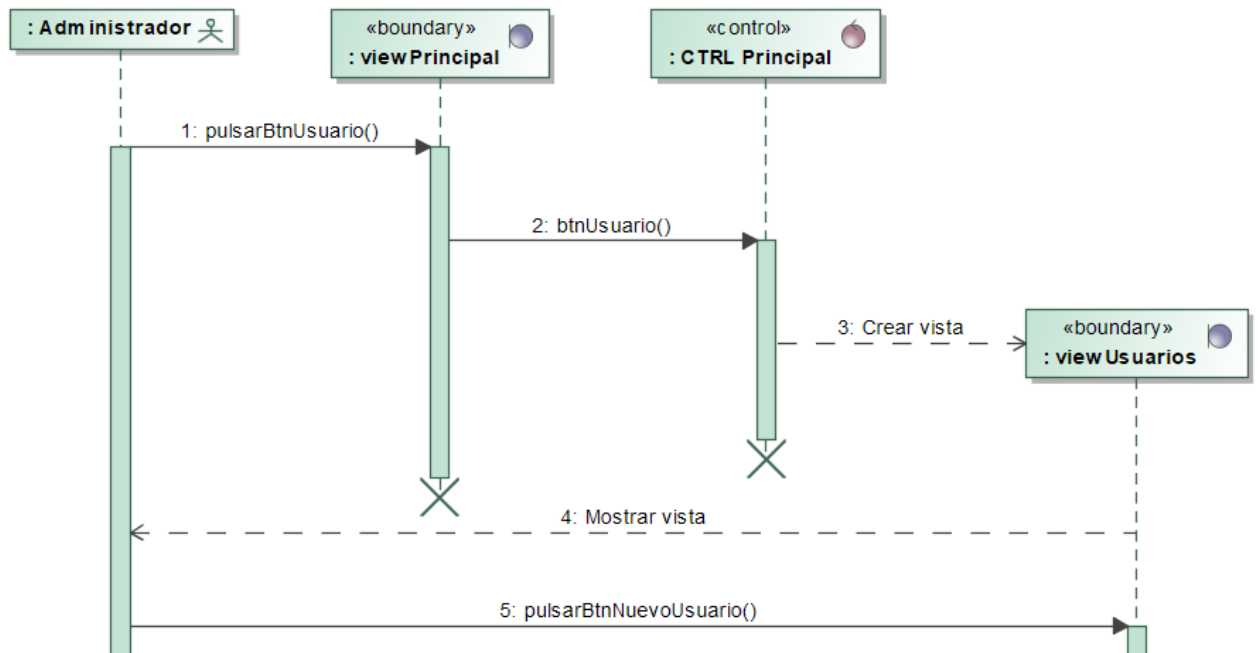


Figura 4.2: Diagrama de secuencia. Crear usuario (Parte 1)

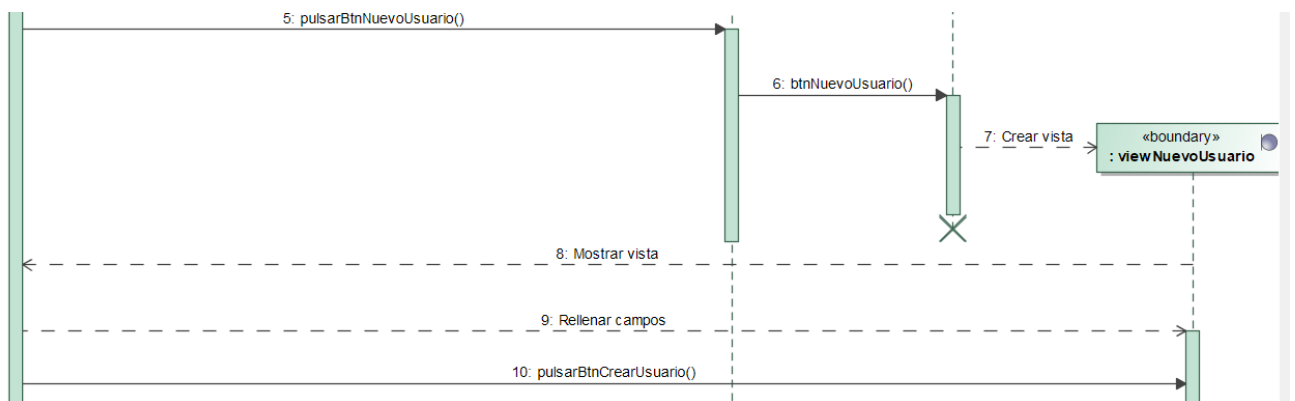


Figura 4.3: Diagrama de secuencia. Crear usuario (Parte 2)

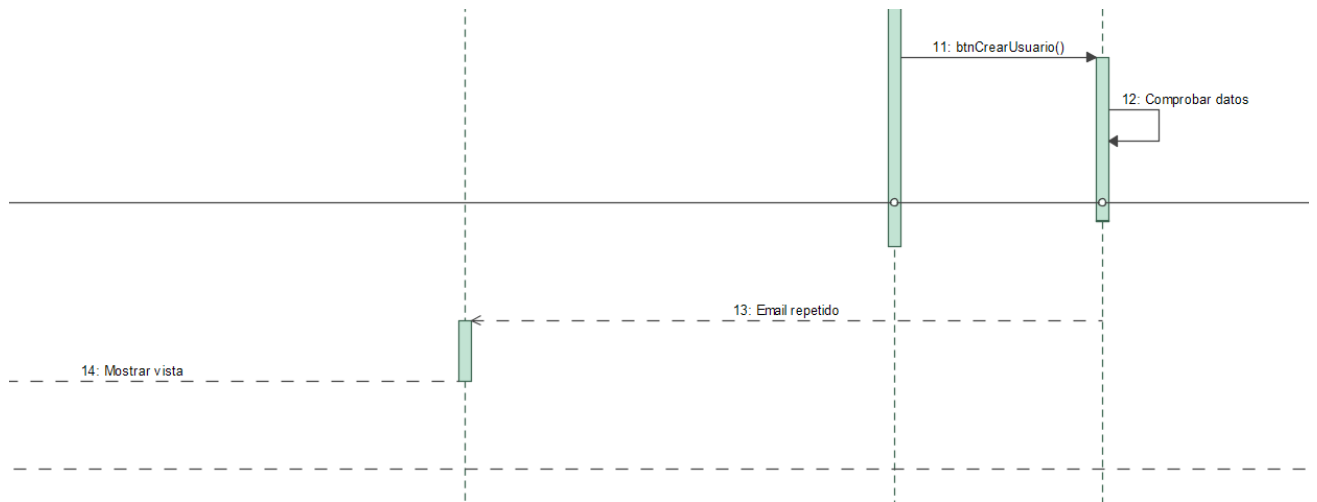


Figura 4.4: Diagrama de secuencia. Crear usuario (Parte 3)

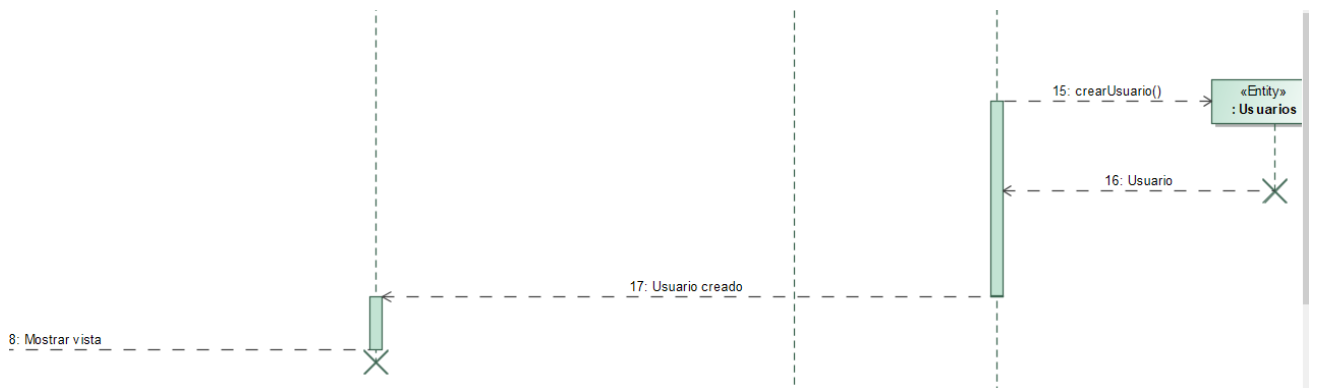


Figura 4.5: Diagrama de secuencia. Crear usuario (Parte 4)

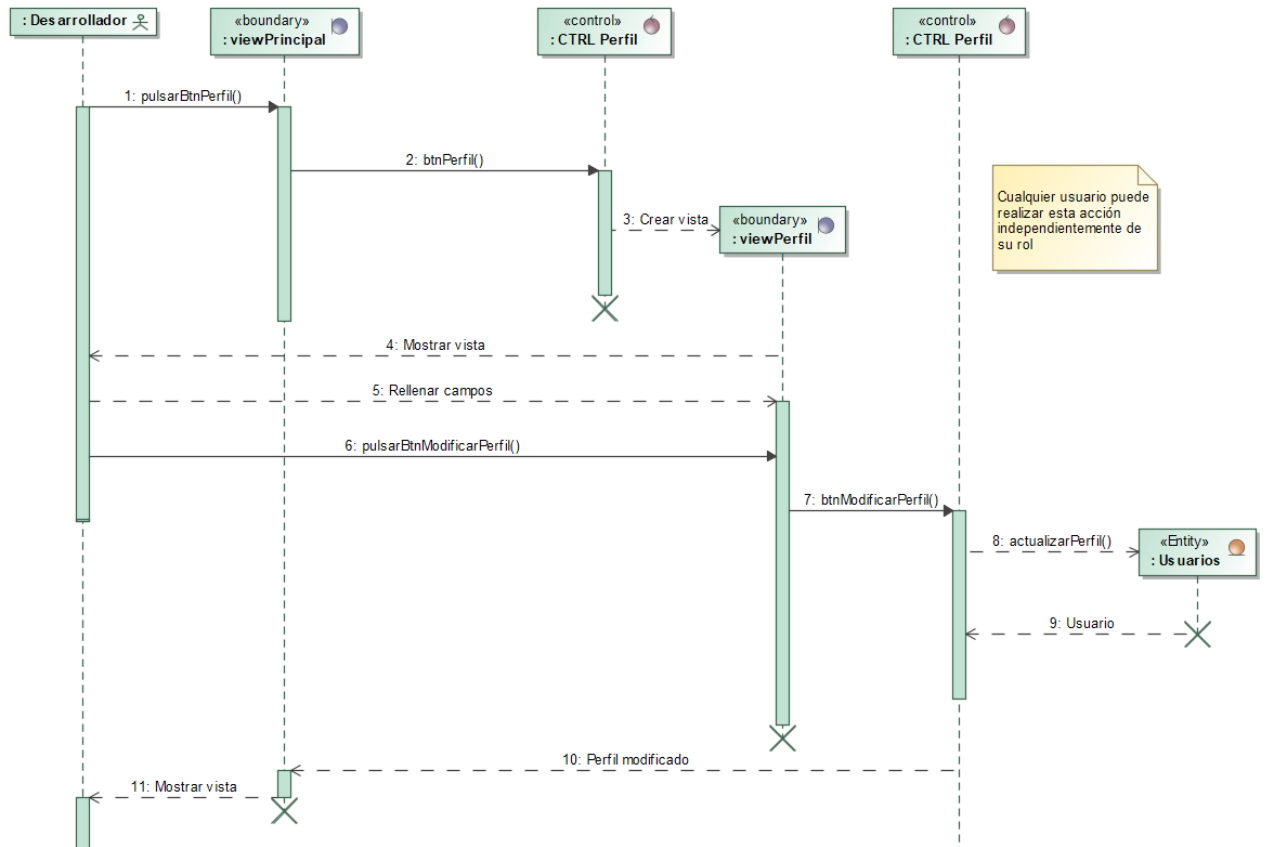


Figura 4.6: Diagrama de secuencia. Leer y Actualizar usuario.

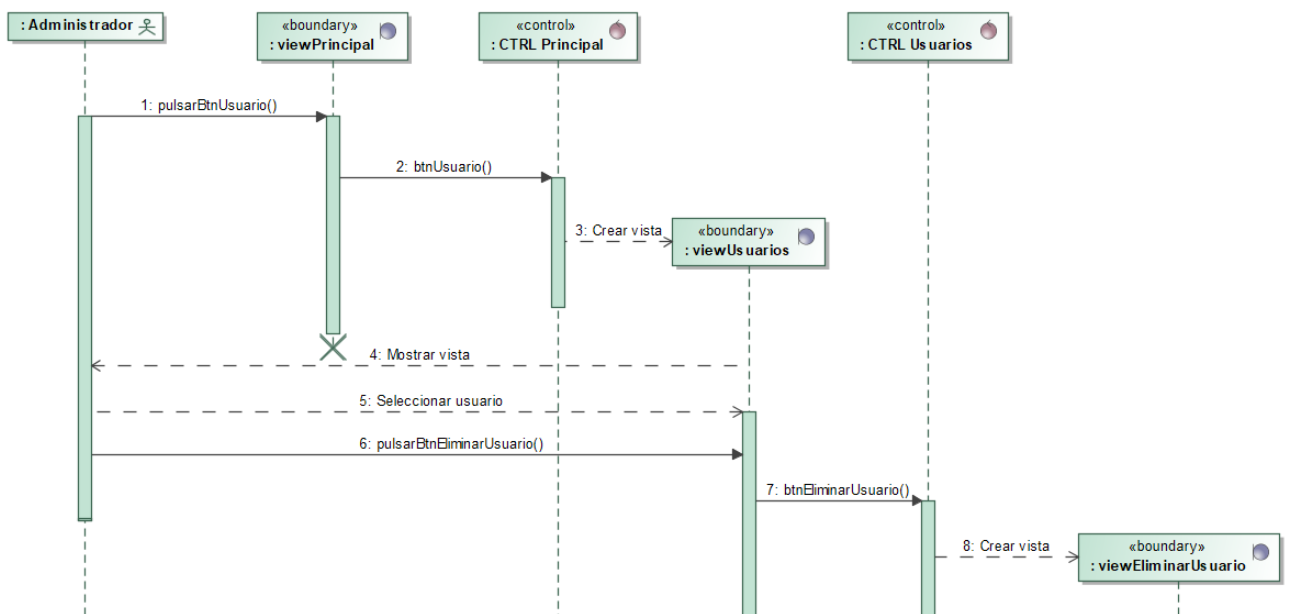


Figura 4.7: Diagrama de secuencia. Eliminar usuario (Parte 1).

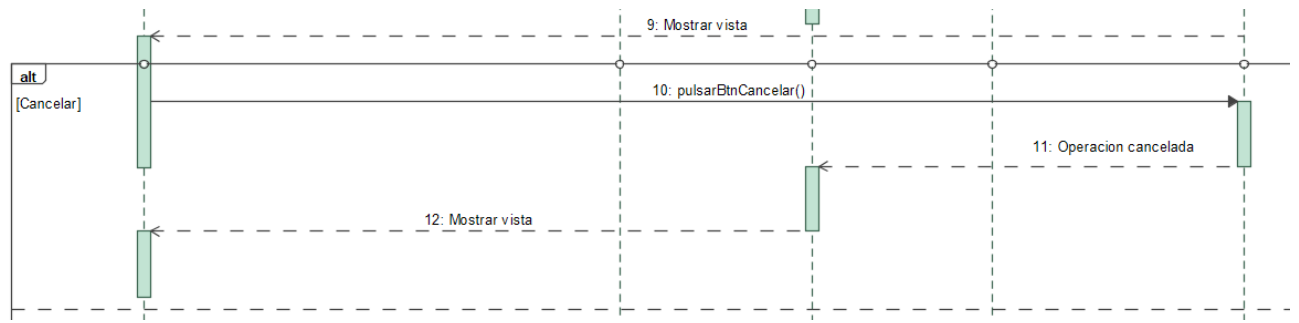


Figura 4.8: Diagrama de secuencia. Eliminar usuario (Parte 2)



Figura 4.9: Diagrama de secuencia. Eliminar usuario (Parte 3)

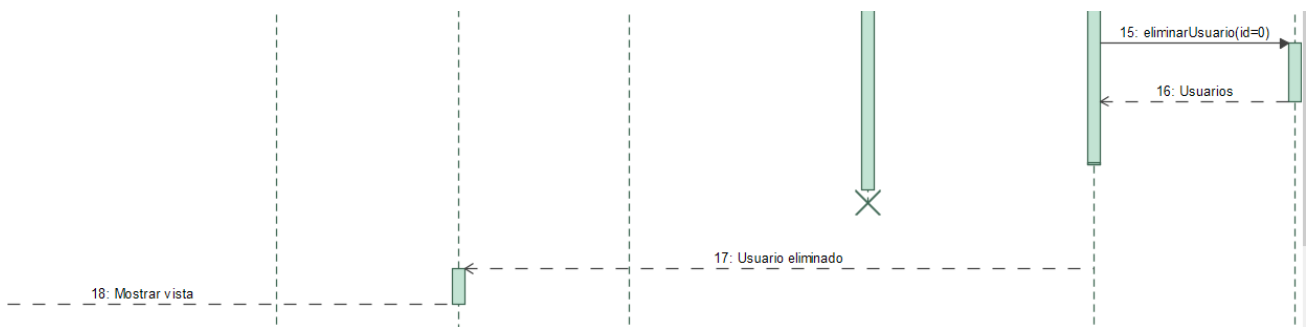


Figura 4.10: Diagrama de secuencia. Eliminar usuario (Parte 4)

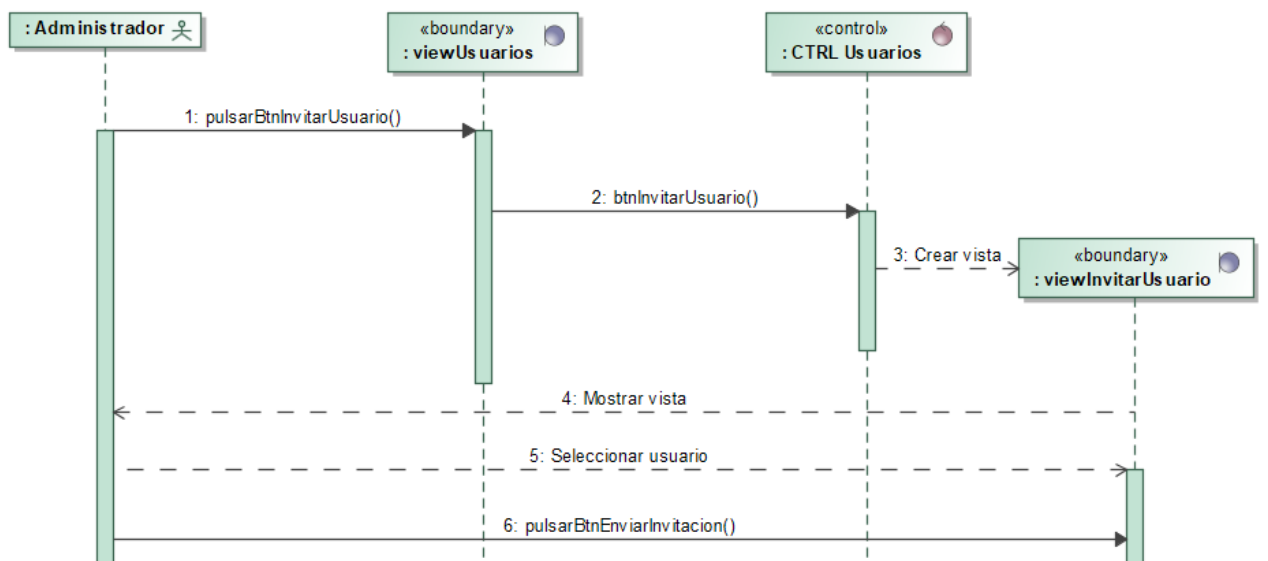


Figura 4.11: Diagrama de secuencia. Invitar usuario (Parte 1)

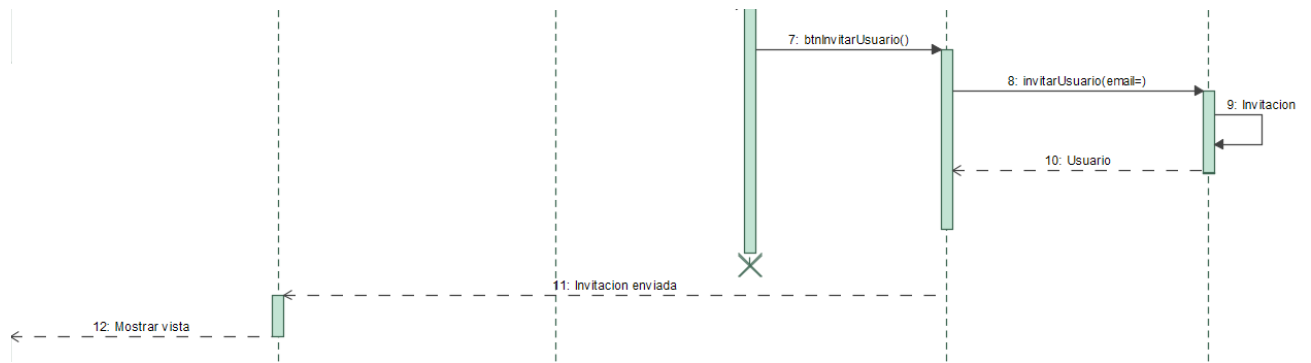


Figura 4.12: Diagrama de secuencia. Invitar usuario (Parte 2)

El resto de los diagramas de secuencia relacionados con los demás casos de uso, se pueden ver tanto en el anexo de esta memoria como dentro del proyecto de MagicDraw, el cual se encuentra subido en el Campus Virtual.

4.4 Modelo físico de datos

El modelo físico de datos está basado en el modelo de clases de entidad visto en la Figura 2.5, con la particularidad que en las relaciones entre las entidades se detalla las claves primarias (o PK) y las claves foráneas (o FK) de cada una de las entidades. [9]

A continuación, vamos a presentar como quedaría el modelo físico de datos de la aplicación, tomando como referencia el modelo de clases de entidad:

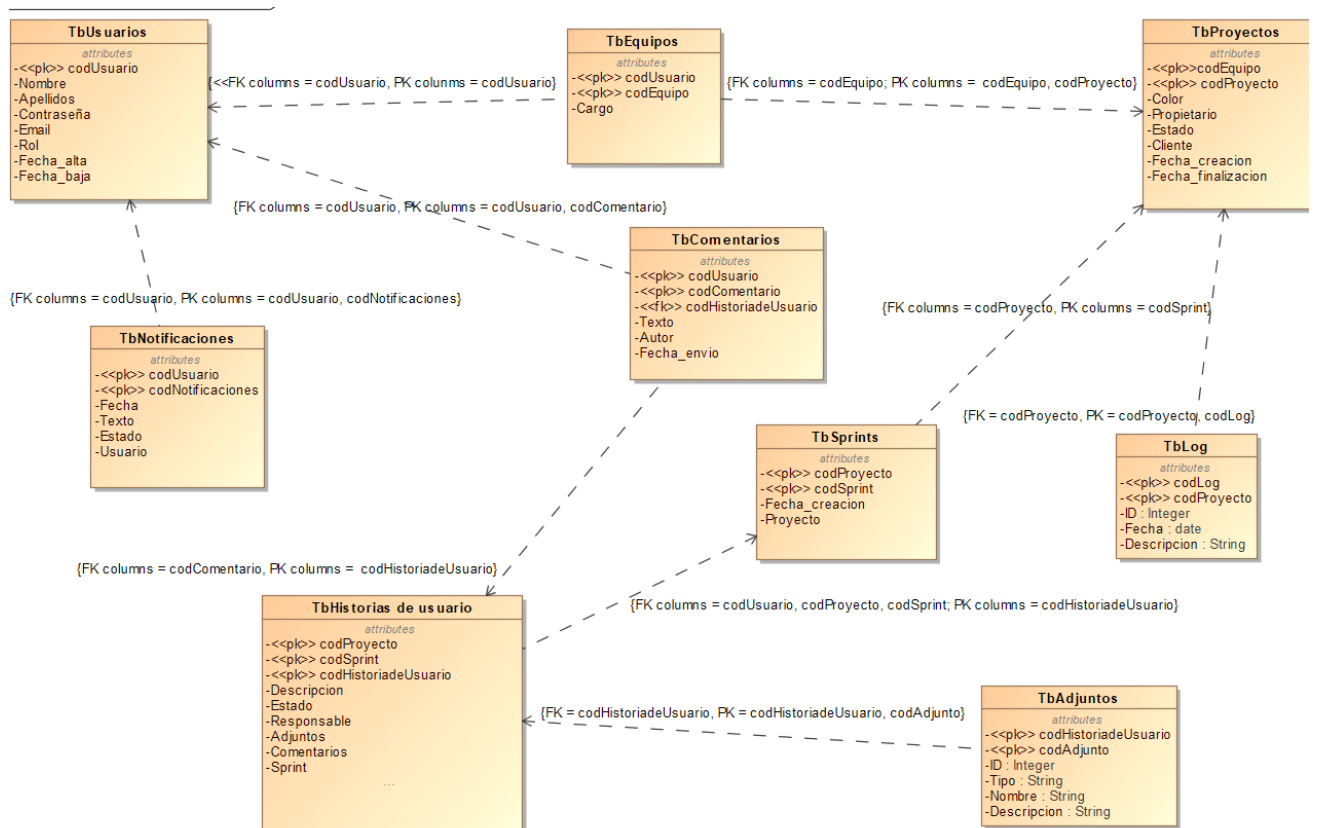


Figura 4.10: Modelo físico de datos de la aplicación

4.5 Modelado de arquitectura

El modelado de arquitectura es un tipo de diagrama que tiene dos claros objetivos:

- Mostrar la disposición de las particiones físicas del sistema de información y la asignación de los componentes software a estas particiones.
- Mostrar las relaciones físicas entre los componentes software y hardware del sistema. [10]

El modelado de arquitectura de este proyecto queda de la siguiente manera: un usuario se conecta a internet a través de dos posibles nodos, un PC o un portátil, a través de Internet se establece una conexión con el servidor de aplicación el cual se conecta con el servidor de base de datos. El diagrama queda estructurado en 3 capas. Una primera capa, llamada presentación, con los nodos de conexión a Internet, una segunda capa, llamada modelo, con el servidor de aplicación y una tercera capa, llamada base de datos, con el servidor de base de datos.

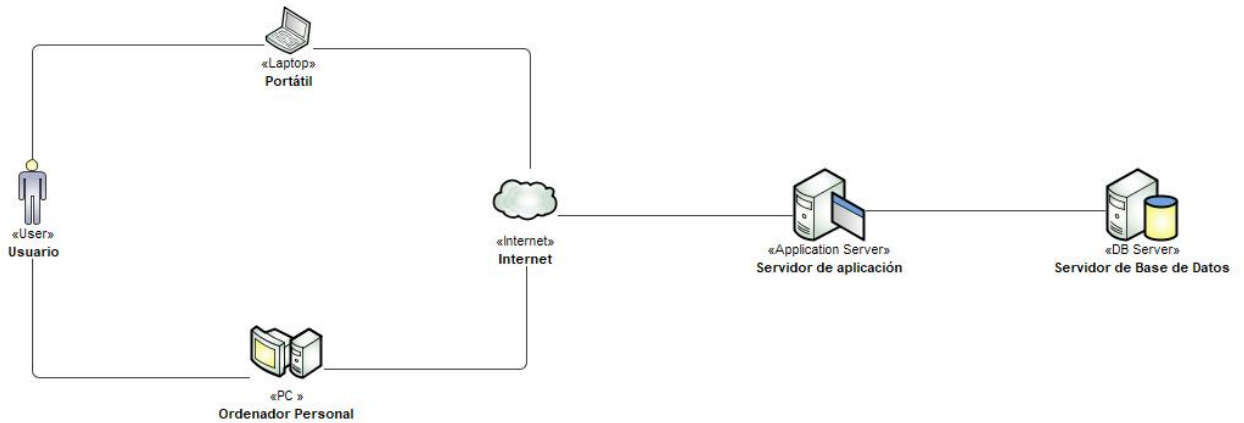


Figura 4.11: Modelado de arquitectura de la aplicación.

4.6 Maquetado de interfaz de usuario

El maquetado de interfaces es la parte de la ingeniería del software encargada del diseño de una aplicación web, relacionada con la interfaz de usuario, los componentes de la web y las vistas, sin ofrecer ningún tipo de funcionalidad. [11]

Para la creación de las maquetas hemos usado la herramienta Balsamiq Wireframes, la cual es una herramienta que proporciona todos los elementos necesarios para la creación de las interfaces de una manera muy sencilla. [12]

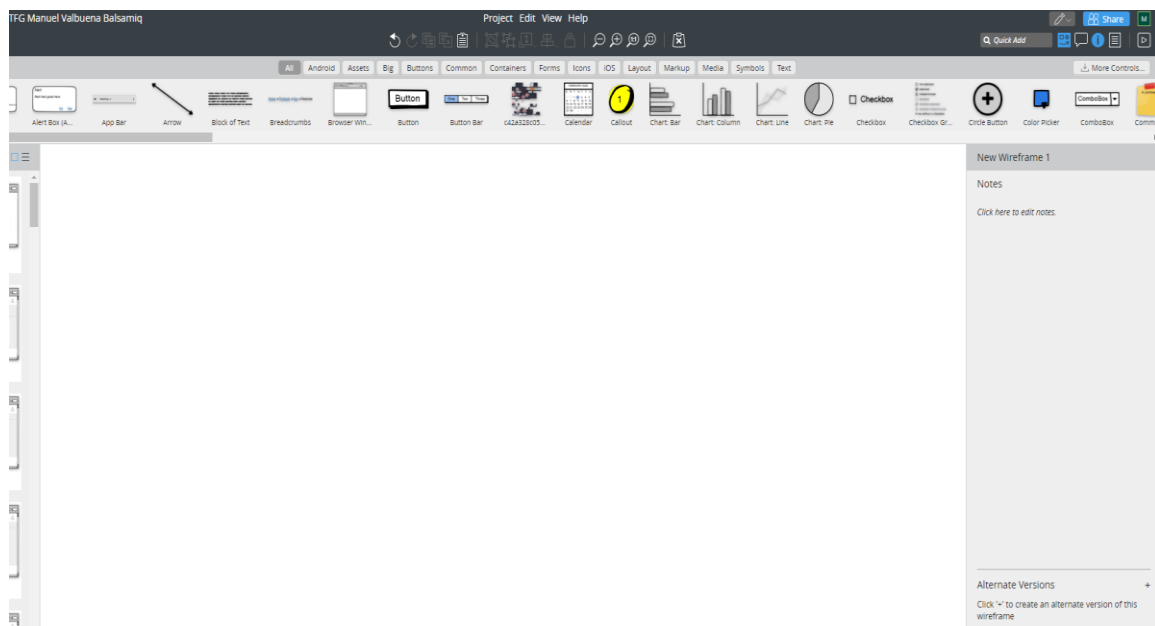


Figura 4.12: Interfaz de la herramienta Balsamiq Wireframes.

A continuación, se presentan algunas de las maquetas de interfaz de usuario que sirvieron a posteriori para la creación del diseño de la aplicación web. El resto de las maquetas se encuentran disponibles tanto en el anexo de esta memoria como en el documento PDF “TFG Manuel Valbuena Alba Balsamiq”, subido en el Campus Virtual.



Figura 4.13: Maqueta de interfaz de usuario. Inicio de sesión.

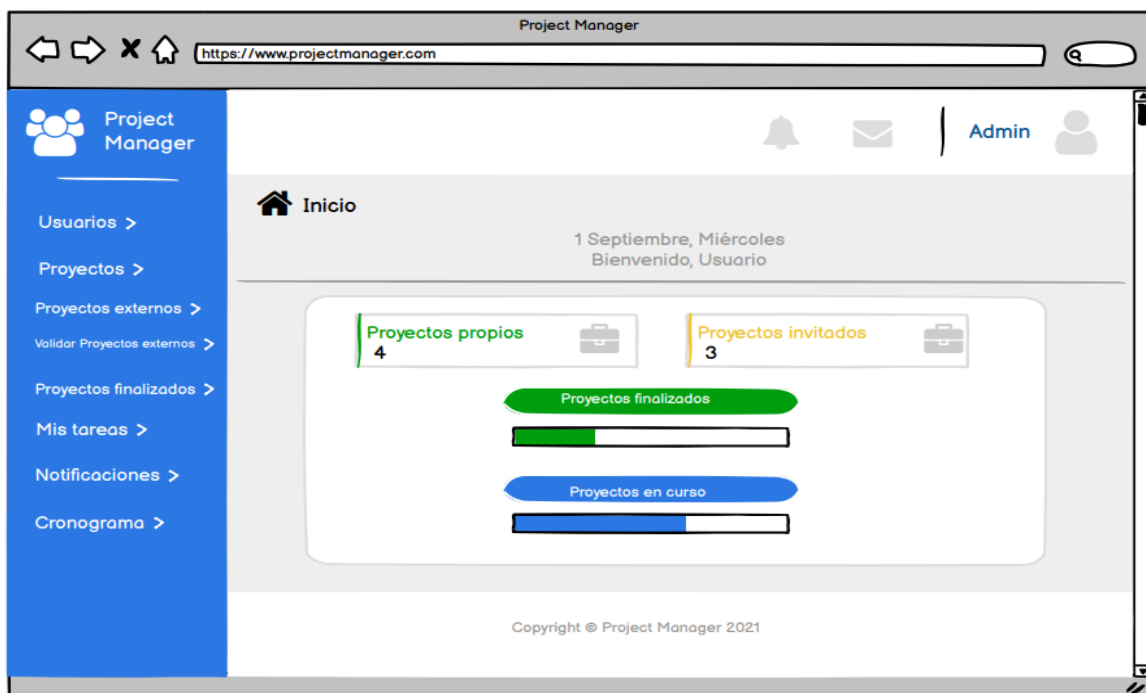


Figura 4.14: Maqueta de interfaz de usuario. Vista principal.

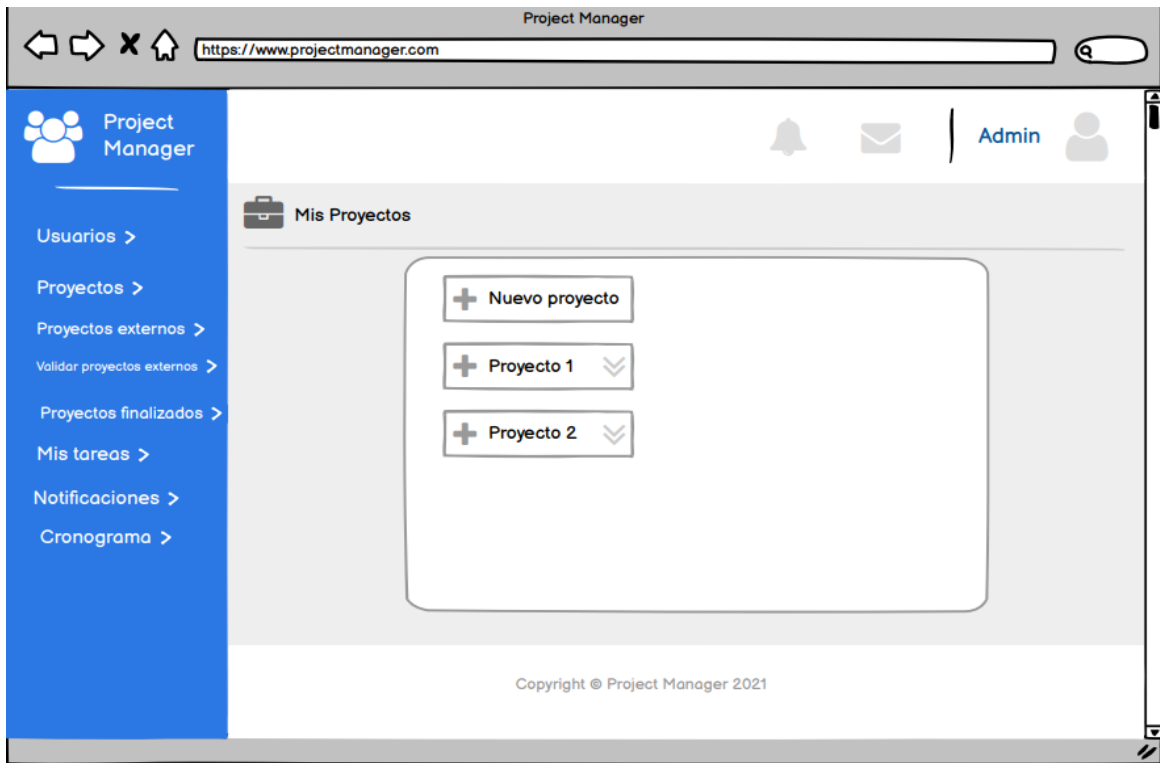


Figura 4.15: Maqueta de interfaz de usuario. Mis proyectos.

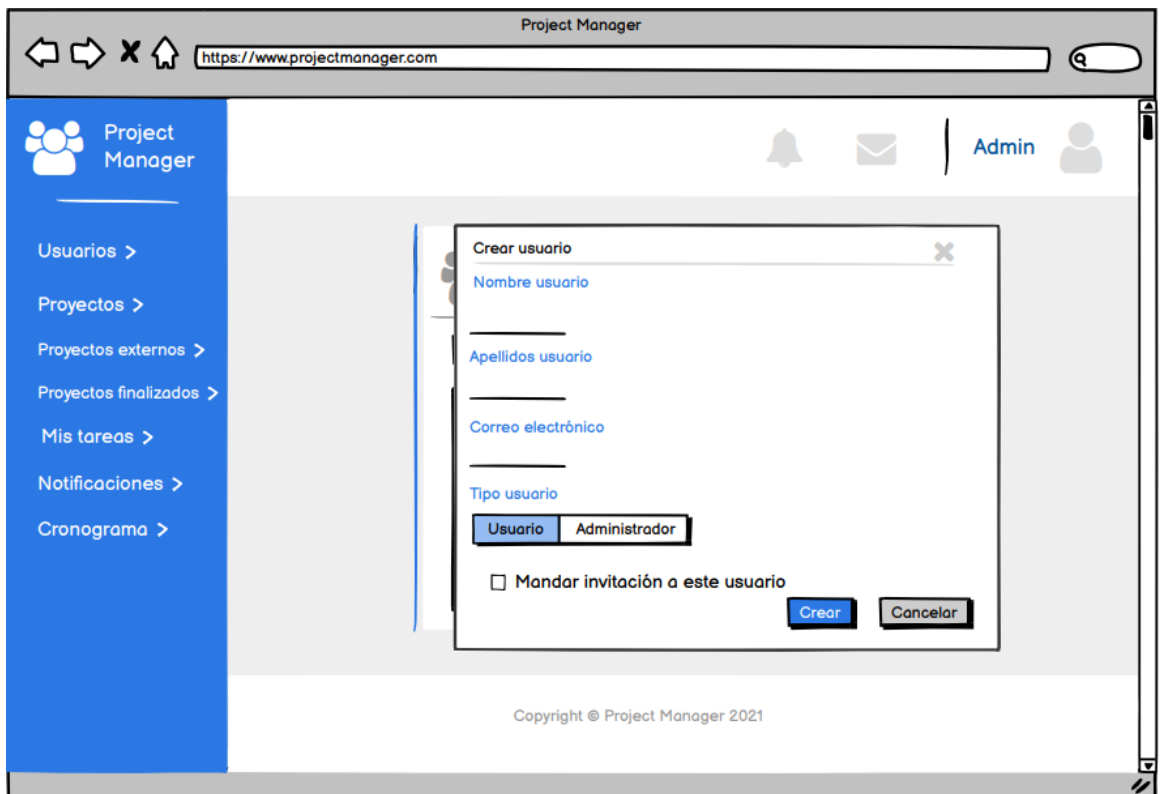


Figura 4.16: Maqueta de interfaz de usuario. Crear usuario.

5 Desarrollo de la aplicación

5.1 Entorno tecnológico

Para la creación de la aplicación web hemos usado diversas herramientas. En primer lugar, hablaremos de SQLyog, la cuál es un sistema gestor de bases de datos relacionales que permite a los usuarios definir, crear y controlar el acceso sobre la base de datos. Esta herramienta fue desarrollada por la empresa Webyog, Inc., cuya última versión estable fue lanzada al público el 29 de abril de 2021. [13]

Algunas de las principales características de SQLyog son:

- Sincronización automática de datos.
- Comparativa visual de datos.
- Comparativa visual y sincronización de esquemas.
- Importación de datos externos.
- Programación de backups.
- Programación e informes sobre consultas.
- Perfiles de consulta. [14]

Como herramienta para la escritura del código fuente de la aplicación web hemos usado Visual Studio Code, el cual es un editor de código para el desarrollo de operaciones como la depuración del código, ejecución de tareas o el control de versiones. [15]

Como lenguaje de programación para la realización de la página web nos hemos ayudado de PHP. Se podría definir PHP como un lenguaje de código abierto especializado para el desarrollo web que se puede incrustar en HTML. La principal ventaja de usar PHP respecto a cualquier otro lenguaje de programación es la gran facilidad de uso del propio lenguaje para la creación de aplicaciones web, aunque también ofrece muchas características avanzadas para gente con más experiencia que busque crear aplicaciones más profesionales. [16]

Además de PHP, también se ha usado tanto JQuery como Bootstrap para mejorar la implementación del front-end de la aplicación. JQuery es una característica incluida en las librerías de JavaScript que permite al usuario manejar tanto los elementos HTML (botones, links, imágenes, etc.), como la tecnología Ajax (técnica para el desarrollo web que consiste en el intercambio de información entre el cliente y el servidor de forma

asíncrona [17]) de forma mucho más fácil. [18]. Respecto a Bootstrap, es una herramienta usada principalmente para darle diseño y personalizar una aplicación web. [19]

Por último, en el desarrollo de este TFG también se ha usado Laragon, que es una herramienta usada para crear un servidor web y un servidor de base de datos, que, a su vez, integra, en este caso, PHP para la programación. [20]

5.2 Implementación de la aplicación

Lo primero que hicimos para implementar la aplicación fue la creación de la base de datos, siguiendo el modelo físico derivado del diagrama de clases de entidad presentado en la Figura 2.5, manteniendo las clases, atributos y relaciones entre las entidades.

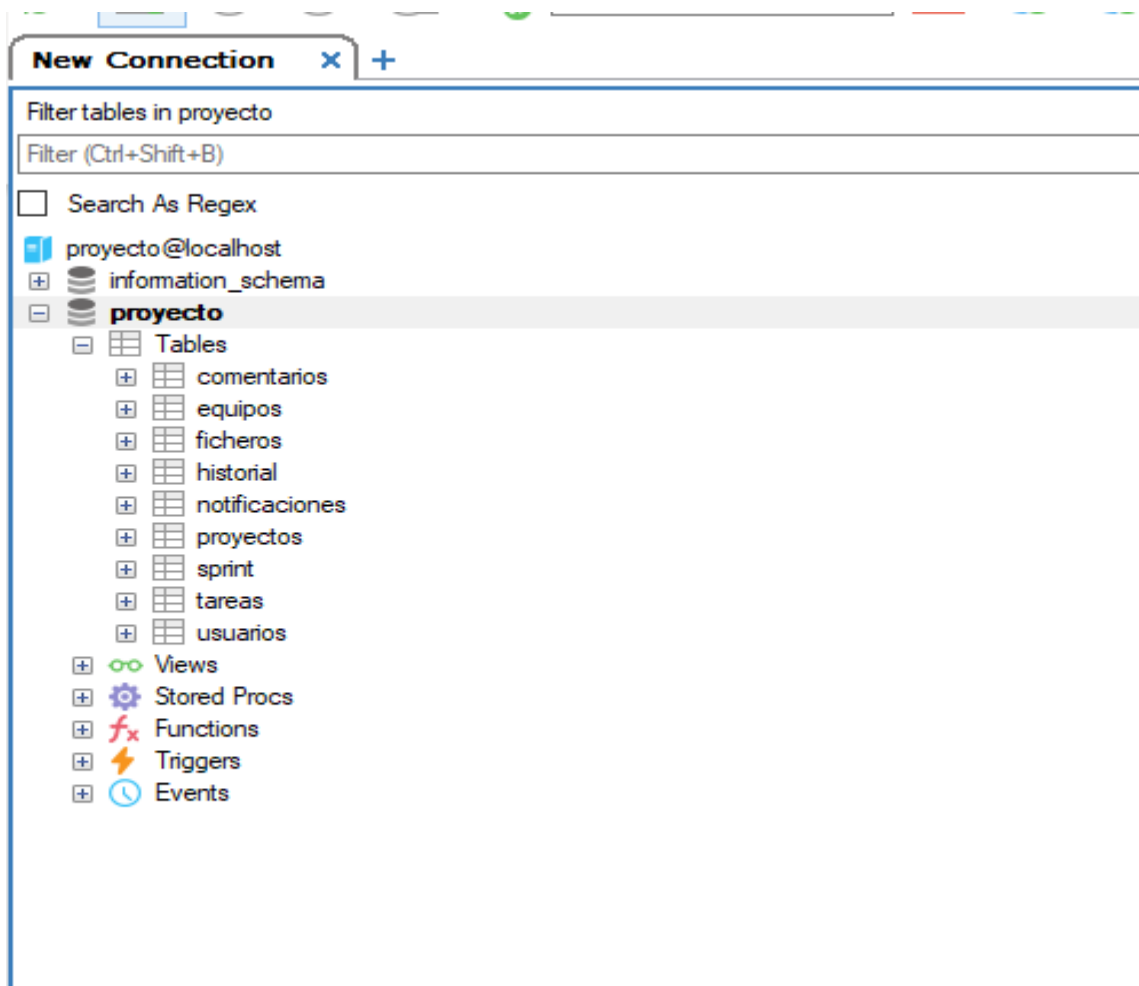


Figura 5.1: Interfaz de la base de datos en SQLyog.

id	nombre	apellidos	email	clave	tipo
101	Admin	Admin	admin@admin.com	\$2y\$10\$3Uw4K9Mu3nUwHEC5f3WLFu7uB5.Vv/WmIYaenfB10hz0hXUJJeST.G	0
104	Manolo	Valbuena Alba	manu3d1996@gmail.com	\$2y\$10\$Eq0s6ztDA0SPiGfZwWnJV2uKNZJlSgrHekQSe4ycQWGS6QdXHy/e9.	1
105	Cliente	Externo	manuvalbuena1996@gmail.com	\$2y\$10\$5viRprbaQgZbE7ha4o6Ou3CkuB6uHm7BKaz3Rk1WThlgZSi5FVSS	2
106	Carlos	Rossi	rossi@uma.es	\$2y\$10\$X0GS4lFkm2aMvzVkwWySTnea5jvYEN35aszn19Gha5aOUWi536y0eO	1
107	Manolo	Valbuena	mval24@hotmail.com	\$2y\$10\$8WYOH6NNZ2HafMeLv9vxu.avEhq5KEussgVqQQfdIQDIRYanyDLTC	1
108	Usuario	Uno	nada@nada.com	\$2y\$10\$8WYOH6NNZ2HafMeLv9vxu.avEhq5KEussgVqQQfdIQDIRYanyDLTC	1
*	(Auto)	(NULL)	(NULL)	(NULL)	1

Figura 5.2: Interfaz de la entidad “Usuarios” en SQLyog.

En la Figura 5.2, podemos ver el resultado de la entidad “Usuarios”, con los atributos que la componen, destacando el atributo “ID”, el cual actúa como PK o clave primaria de la entidad. De la misma forma están compuestas el resto de las clases de la base de datos.

Respecto a la realización del código fuente, el proyecto se ha estructurado de la siguiente forma: todos los archivos empiezan con dos *includes* que representan una cabecera y un menú de opciones en la parte lateral izquierda de la vista.

```
<?php
|   include_once("include/cabecera.php");
|   include_once("include/topbar.php");
?>
```

Figura 5.3: Estructura de los includes en VSC.



Figura 5.4: Cabecera en la aplicación web.

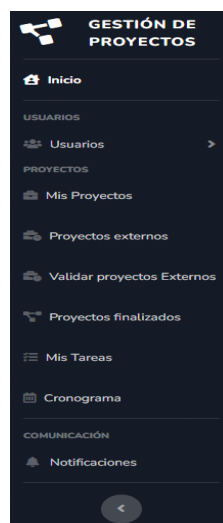


Figura 5.5: Menú lateral izquierda en la aplicación web

El resto del código de cada uno de los archivos está estructurado mediante HTML y JQWidgets. La librería JQWidgets para capturar los distintos elementos creados a través de HTML para la confección de la interfaz de usuario y, después, a través de jqwidgets se accede a ciertos elementos específicos de la interfaz para su manipulación. Cada elemento de la interfaz de HTML tiene su propio jqwidgets de configuración. [21]

Por ejemplo, para los elementos de un formulario HTML, tenemos sus correspondientes jqwidgets, como los siguientes:

- Para los elementos `<input type="text" />` está el elemento jqwidgets **jqxInput**.
- Para los elementos `<textarea />` está el elemento jqwidgets **jqxTextArea**.
- Para los elementos `<input type="button" />` está el elemento jqwidgets **jqxButtons**.

Por ejemplo, en el siguiente bloque de interfaz usuario de HTML hay un input de tipo text, un textarea y un button.

```
<div class="form-group">
  <input type="text" id="proyecto" name="proyecto" required/ >
</div>
<br>
<div class="form-group">
  <textarea id="descripcion" name="descripcion"></textarea>
</div>
<div class="form-group">
  <p>Definir color</p>
  <button type="button" id="primary" class="color btn btn-primary"
value="border-left-primary" style="height: 30px; width: 30px;"></button>
```

Figura 5.6: Ejemplo de código HTML

Utilizando los correspondientes jqwidgets, se accede a cada uno de dichos elementos para su configuración y manipulación. Todo ello se realiza a través de javascript.

En este caso en concreto, para acceder a dichos elementos de HTML desde javascript se hace uso del atributo `id` de cada uno de los elementos, indicando el tipo de elemento.

```
// Configurar jqxwidget
$("#proyecto").jqxInput({
  width: 400,
  height: 35,
  placeholder: 'Nombre del proyecto'
});
$('#descripcion').jqxTextArea({
  placeholder: 'Haz la descripción del proyecto',
  height: 90,
  width: 400,
  minLength: 1
```

```
});
```

Figura 5.7: Ejemplo de cómo manipular elementos HTML con JQWidgets.

Para finalizar el código de cualquier archivo encontramos otro include que representa el pie de página.

```
<?php  
| include_once("include/pie.php");  
?>
```

Figura 5.8: Estructura del include "pie" en VSC.



Figura 5.9: Pie de página en la aplicación web.

Es conveniente comentar la característica más compleja que se ha implementado. Esta es la captura de eventos comunes en elementos diferentes.

En la aplicación web, hay numerosos apartados donde se muestra una serie de elementos o bloques, donde cada uno de ellos tiene un menú idéntico. Además, el número de elementos o bloques puede ir variando a lo largo del uso de la aplicación.

Por ejemplo, en la opción MIS PROYECTOS, donde un usuario gestiona los distintos proyectos que tiene asignado, hay un bloque que representada a cada uno de los proyectos que tiene asignados.

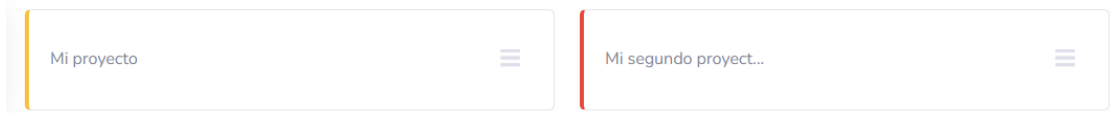


Figura 5.10: Vista "Mis Proyectos" en la aplicación web.

Pero todos los bloques tienen el mismo menú con las mismas opciones.

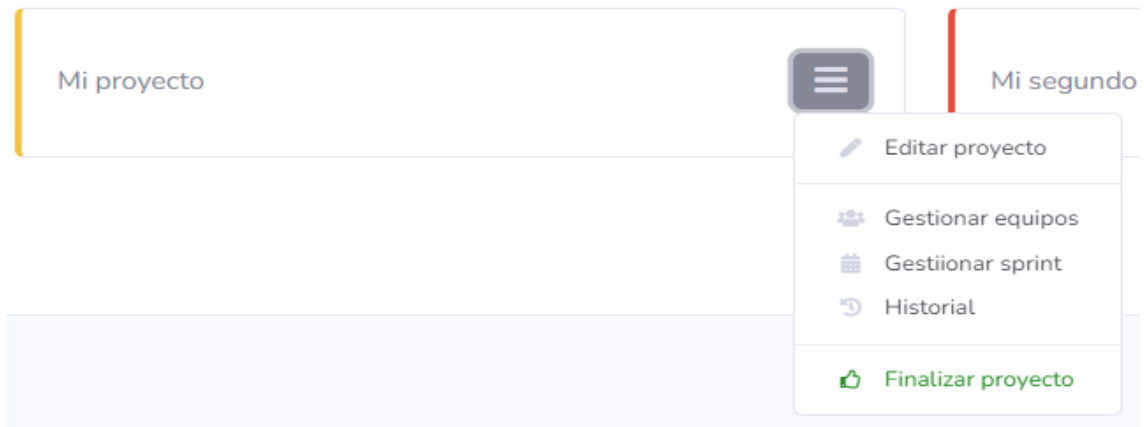


Figura 5.11: Vista "Opciones" en "Mis Proyectos".

Antes de nada, hay que comentar que cuando se selecciona una determinada opción del menú, se genera lo que se llama un evento. Los eventos se pueden generar por muy diversos motivos. Concretamente, en el caso de las opciones del menú el evento que se genera se llama click.

Para capturar los eventos generados utilizo la librería jquery. Esta librería permite identificar y gestionar los distintos elementos de una página web a través del atributo id de cualquier etiqueta HTML.

El atributo id tiene un problema en este caso, cada atributo id referente a un proyecto debe ser distinto, por lo que sería imposible saber todos los atributos id de los proyectos, ya que el número de proyectos no es fijo. Lo que se quiere decir con esto, es que, en este caso, no resulta útil utilizar el atributo id del proyecto y capturar su evento click, ya que sería imposible.

La librería jquery permite identificar elementos por el nombre de una clase. En este caso, el nombre de la clase si se puede repetir en diferentes elementos HTML.

Como se puede observar en el ciclo que se encarga de recuperar todos los proyectos del usuario de la tabla para mostrarlos en la pantalla, en la misma opción del menú de cada uno de los proyectos, en su atributo class tienen en mismo nombre de clase.

Por ejemplo, en el caso de la opción del menú Editar proyecto de cada uno de los bloques de proyecto, todos tienen una clase común que se llama modificar.

```
<!-- Menú Dropdown del proyecto -->
<div class="dropdown-menu" aria-labelledby="p<?php echo $fila->id; ?>">
<a class="modificar dropdown-item" href="#" value="<?php echo $fila->id; ?>" >
<i class="fas fa-pen fa-sm fa-fw mr-2 text-gray-400"></i> Editar proyecto
</a>
```

Figura 5.12: Ejemplo de código HTML usando la librería jquery.

Jquery permite capturar eventos a partir del nombre de la clase. Es decir, la clase va a servir para capturar un único evento click de una determinada opción del menú independiente del menú en donde se haya seleccionado la opción.

Cuando se selecciona la opción Editar proyecto, tal y como se ve en la Figura 4.11, independientemente sea de Mi proyecto o de Mi segundo proyecto, en evento click generado lo va a capturar a través del nombre de la clase común para dicha opción en cada uno de los distintos menús. En este caso la clase se llama modificar.

Ya dentro del evento click capturado por la clase es donde se va a diferenciar si dicho evento click ha sido generado por el primer proyecto o el segundo. De esta forma ya se está accediendo de una forma concreta a la opción de un determinado proyecto.

```
// Modificar los datos de un proyecto
$(".modificar").click(function(){
    operacion="M";
    idProyecto=$(this).attr("value");
```

Figura 5.13: Ejemplo de código en el que se recupera el proyecto concreto sobre el que se ha pulsado.

En definitiva, captura un evento que es común a distintos elementos y dentro de este evento común es donde se diferencia sobre qué elemento en concreto se está trabajando.

El uso de esta forma de programación ahorra mucho código y mejora, sobre el todo, el mantenimiento de la aplicación.

Como se ha comentado anteriormente, esta codificación es utilizada muy a menudo en las diferentes opciones de la aplicación.

Para finalizar, presentamos algunas capturas de pantalla de la aplicación web.

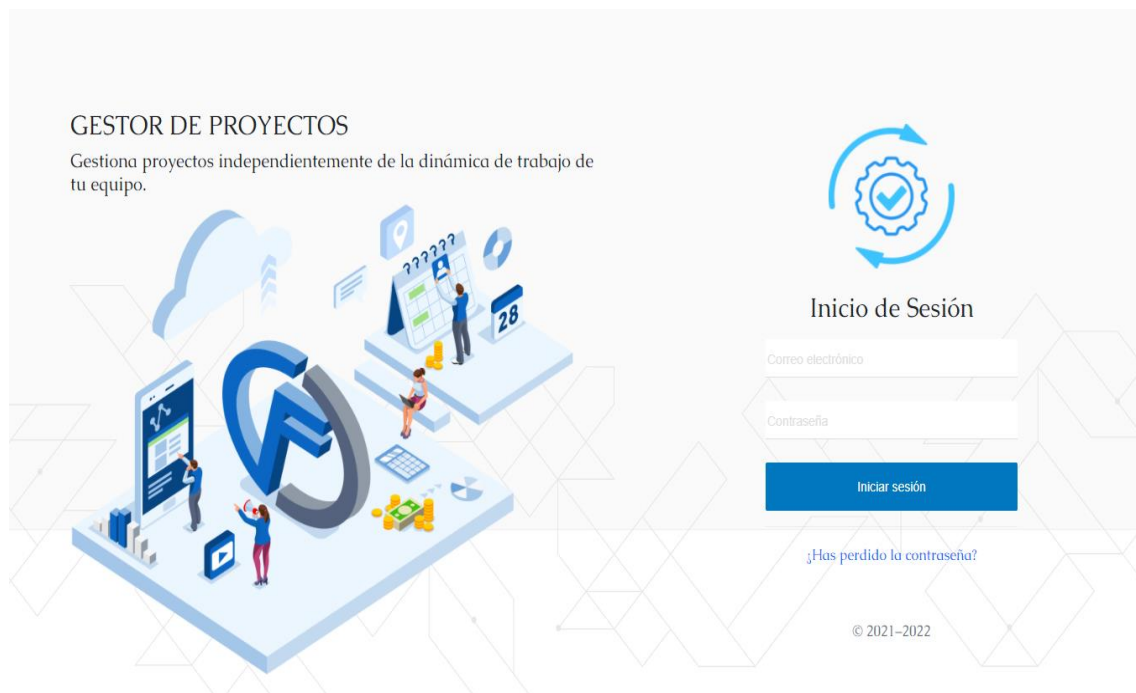


Figura 5.14: Vista "Inicio de sesión".

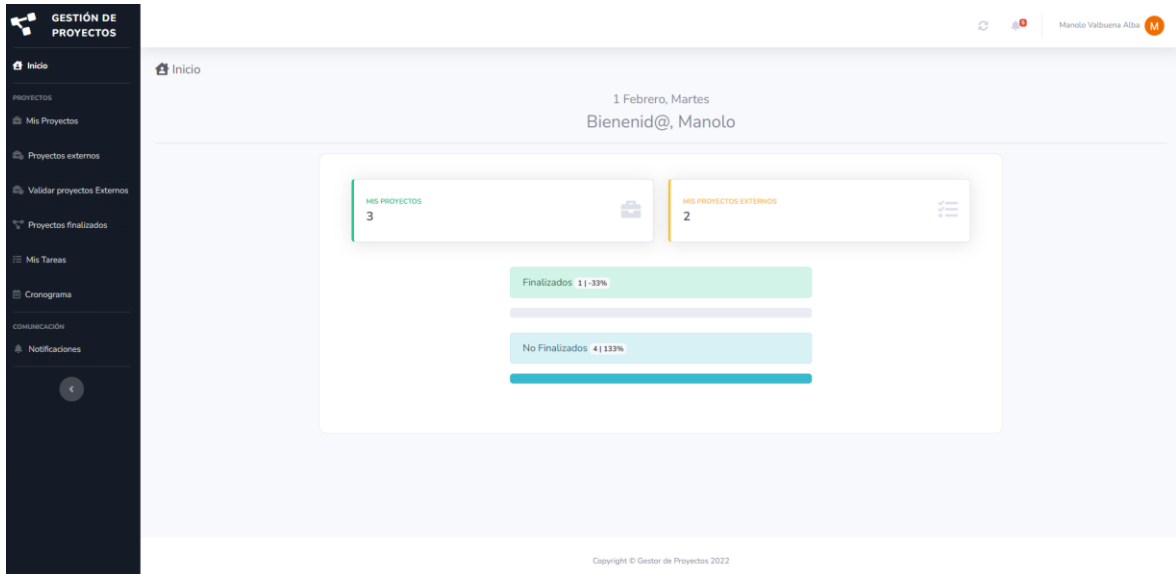


Figura 5.15: Vista "Principal".

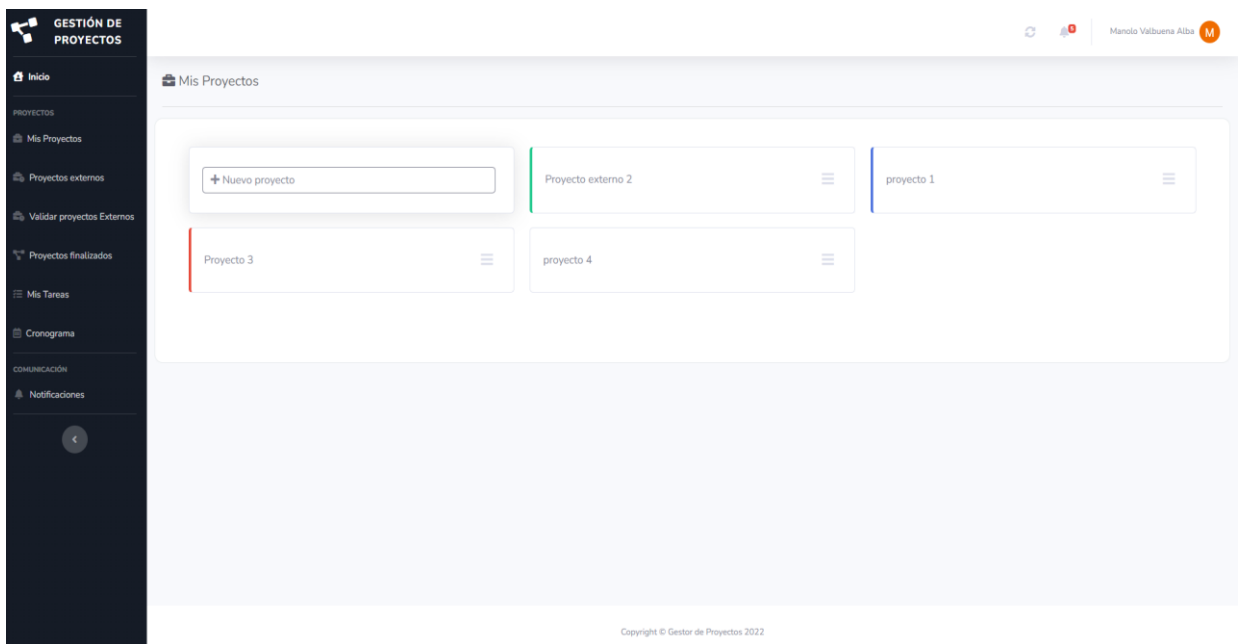


Figura 5.16: Vista "Mis proyectos"

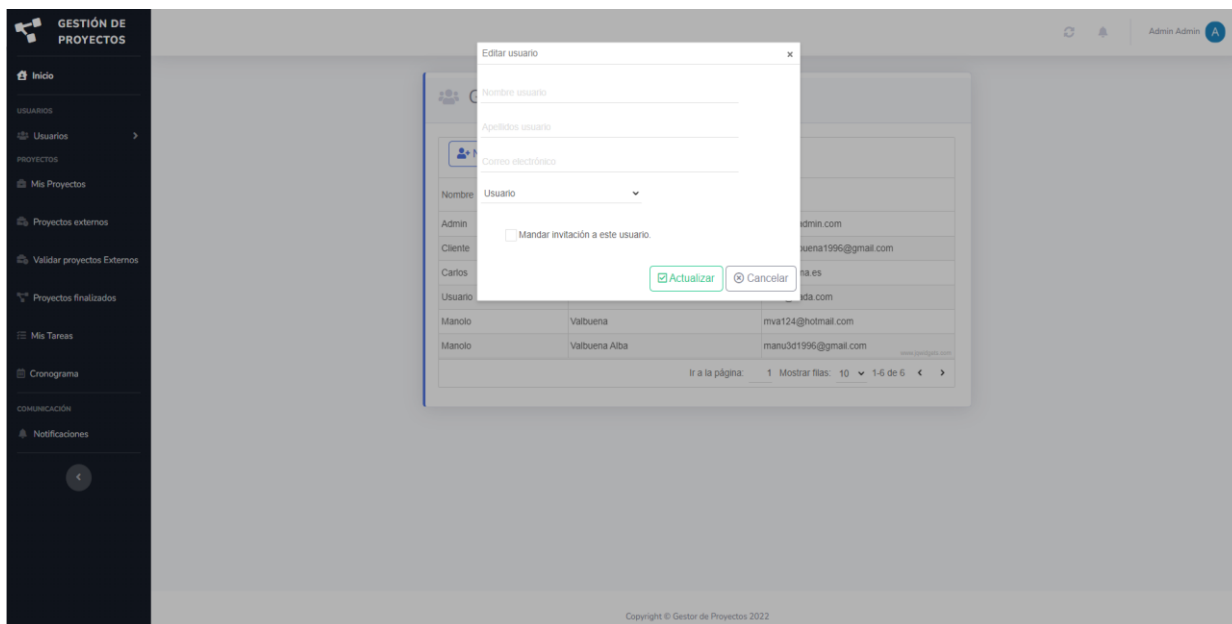


Figura 5.17: Vista “Crear usuario”.

Como se ha podido ver en estas últimas figuras, y se ha comentado anteriormente, todas las vistas de la aplicación cuentan con la cabecera, pie de página y menú lateral izquierdo, siendo el contenido central lo que cambia de una vista respecto a otra.

5.3 Resultados. Casos de prueba.

Podemos definir los casos de prueba como una serie de condiciones o variables que se estudiarán para determinar si un sistema o aplicación son válidos o no. [22]

A continuación, se presentan algunos ejemplos de los casos de prueba que se han tenido en cuenta para validar el correcto funcionamiento de la aplicación web desarrollada en este TFG:

Identificador	CP. Unitaria – Inicio de sesión - 01
Descripción	Comprobar que el controlador envía correctamente los datos introducidos.
Componentes afectados	Equipo, viewLogin, CTRL Login, Usuarios
Resultado esperado	Un usuario rellena los campos “correo electrónico” y “contraseña”, pulsa el botón “Iniciar sesión” y el controlador debe enviar correctamente los datos introducidos por el usuario para verificar el inicio de sesión y mostrar la página principal.
Resultado obtenido	Al pulsar el botón “Iniciar sesión”, el controlador envía los datos, comprueba que son correctos y muestra la vista principal.

Corrección realizada	No ha sido necesaria realizar ninguna corrección.
Fecha	18/01/2022
Responsable	Manuel Valbuena Alba

Identificador	CP. Unitaria – CRUD Usuario - 01
Descripción	Integración de interfaces.
Componentes afectados	Administrador, viewPrincipal, viewUsuarios, viewNuevoUsuario, CTRL Principal, CTRL Usuarios, CTRL Nuevo Usuario, Usuarios.
Resultado esperado	El administrador rellena los campos, pulsa el botón “Crear usuario” y el controlador deberá crear el usuario con los datos rellenos.
Resultado obtenido	Al pulsar el botón “Crear usuario”, el usuario cuyos datos han sido rellenos se crea.
Corrección realizada	No ha sido necesaria realizar ninguna corrección.
Fecha	18/01/2022
Responsable	Manuel Valbuena Alba

Identificador	CP. Unitaria – CRUD Usuario - 02
Descripción	Integración de interfaces.
Componentes afectados	Administrador, viewPrincipal, viewUsuarios, viewEliminarUsuario, CTRL Principal, CTRL Usuarios, CTRL Eliminar Usuario, Usuarios.
Resultado esperado	El administrador selecciona un usuario que deberá ser eliminado del sistema por el controlador.
Resultado obtenido	Al pulsar el botón “Eliminar usuario”, el usuario seleccionado es eliminado de la aplicación.
Corrección realizada	No ha sido necesaria realizar ninguna corrección.
Fecha	18/01/2022
Responsable	Manuel Valbuena Alba

Identificador	CP. Unitaria – CRUD Usuario - 03
Descripción	Integración de interfaces.
Componentes afectados	Equipo, viewPrincipal, viewPerfil, CTRL Principal, CTRL Perfil
Resultado esperado	Un usuario pulsa el botón “Perfil”, el controlador deberá mostrar el perfil del usuario.
Resultado obtenido	Al pulsar el botón “Perfil”, se muestra el perfil del usuario que lo ha solicitado.
Corrección realizada	No ha sido necesaria realizar ninguna corrección.
Fecha	18/01/2022
Responsable	Manuel Valbuena Alba

Identificador	CP. Unitaria – CRUD Usuario - 04
Descripción	Integración de interfaces.
Componentes afectados	Equipo, viewPrincipal, viewPerfil, CTRL Principal, CTRL Perfil
Resultado esperado	El administrador pulsa el botón “Invitar usuario” y el controlador deberá enviar una invitación a dicho usuario mediante un email.
Resultado obtenido	Al pulsar el botón “Invitar usuario”, el sistema envía una invitación al usuario a través de un correo electrónico.
Corrección realizada	No ha sido necesaria realizar ninguna corrección.
Fecha	18/01/2022
Responsable	Manuel Valbuena Alba

Identificador	CP. Unitaria – CRUD Usuario - 05
Descripción	Integración de interfaces.
Componentes afectados	Equipo, viewPrincipal, viewPerfil, CTRL Principal, CTRL Perfil
Resultado esperado	Un usuario pulsa el botón “Modificar datos”, el controlador deberá mostrar el perfil del usuario y la posibilidad de modificar sus datos personales.
Resultado obtenido	Al pulsar el botón “Perfil”, se muestra el perfil del usuario que lo ha solicitado y se muestran los campos para que el usuario los modifique.
Corrección realizada	No ha sido necesaria realizar ninguna corrección.
Fecha	18/01/2022
Responsable	Manuel Valbuena Alba

Identificador	CP. Unitaria – CRUD Proyectos - 01
Descripción	Integración de interfaces.
Componentes afectados	Jefe de proyecto, viewPrincipal, viewProyectos, viewNuevoProyecto, CTRL Principal, CTRL Proyectos, CTRL Nuevo Proyecto, Proyectos.
Resultado esperado	El jefe de proyecto pulsará el botón “Nuevo proyecto”, rellenará los campos necesarios y el controlador deberá crear un nuevo proyecto con los datos introducidos.
Resultado obtenido	Al pulsar el botón “Nuevo proyecto”, se crea correctamente el proyecto con los datos introducidos.
Corrección realizada	No ha sido necesaria realizar ninguna corrección.
Fecha	19/01/2022
Responsable	Manuel Valbuena Alba

Identificador	CP. Unitaria – CRUD Proyectos - 02
Descripción	Integración de interfaces.
Componentes afectados	Jefe de proyecto, viewProyectos, viewEliminarProyecto, CTRL Proyectos, CTRL Eliminar Proyecto, Proyectos.
Resultado esperado	El jefe de proyecto pulsará el botón “Eliminar proyectos” y el controlador deberá eliminar el proyecto seleccionado.
Resultado obtenido	Al pulsar el botón “Eliminar proyecto”, se elimina correctamente el proyecto seleccionado.
Corrección realizada	No ha sido necesaria realizar ninguna corrección.
Fecha	19/01/2022
Responsable	Manuel Valbuena Alba

Identificador	CP. Unitaria – CRUD Proyectos - 03
Descripción	Integración de interfaces.
Componentes afectados	Jefe de proyecto, viewProyectos, viewEditarProyecto, CTRL Proyectos, CTRL Editar Proyecto, Proyectos.
Resultado esperado	El jefe de proyecto pulsará el botón “Editar proyecto”, rellenará los campos necesarios y el controlador deberá actualizar el proyecto con los datos introducidos.
Resultado obtenido	Al pulsar el botón “Editar proyecto”, se actualizar correctamente el proyecto con los datos introducidos.
Corrección realizada	No ha sido necesaria realizar ninguna corrección.
Fecha	19/01/2022
Responsable	Manuel Valbuena Alba

Identificador	CP. Unitaria – Asignación de historias de usuario
Descripción	Integración de interfaces.
Componentes afectados	Jefe de proyecto, viewSprints, viewHistoriadeUsuario, viewAsignarResponsable, CTRL Sprints, CTRL Historia de Usuario, CTRL Asignar Responsable, Usuarios.
Resultado esperado	El jefe de proyecto seleccionará un miembro del equipo de desarrollo y el controlador asignará como responsable de una historia de usuario.
Resultado obtenido	Al pulsar el botón “Asignar responsable”, el usuario seleccionado queda como responsable de la historia de usuario.
Corrección realizada	No ha sido necesaria realizar ninguna corrección.
Fecha	20/01/2022
Responsable	Manuel Valbuena Alba

Identificador	CP. Unitaria – Adjuntar archivo a historia de usuario
Descripción	Integración de interfaces.
Componentes afectados	Jefe de proyecto, viewSprints, viewHistoriasdeUsuario, CTRL Sprints, CTRL Historias de Usuario.
Resultado esperado	Al pulsar el botón “Subir archivo”, el controlador deberá abrir una ventana para que el usuario pueda subir a la aplicación cualquier archivo a la web.
Resultado obtenido	Al pulsar el botón, se abre la ventana y permite al usuario subir un archivo.
Corrección realizada	No ha sido necesaria realizar ninguna corrección.
Fecha	20/01/2022
Responsable	Manuel Valbuena Alba

Identificador	CP. Unitaria – Log de historia de usuario
Descripción	Integración de interfaces.
Componentes afectados	Desarrolladores, viewSprints, viewHistoriasdeUsuario, viewLogHistoriadeUsuario, CTRL Sprints, CTRL Historia de Usuario.
Resultado esperado	Al pulsar el botón “Historial”, el controlador deberá poder acceder al historial se mostrará un log con todos los cambios que se hayan producido sobre una historia de usuario.
Resultado obtenido	Al pulsar el botón, se muestra correctamente el log de la historia de usuario.
Corrección realizada	No ha sido necesaria realizar ninguna corrección.
Fecha	21/01/2022
Responsable	Manuel Valbuena Alba

Identificador	CP. Unitaria – Log de historia de usuario
Descripción	Comprobar que el controlador muestra los cambios sobre una historia de usuario.
Componentes afectados	Desarrolladores, viewSprints, viewHistoriasdeUsuario, viewLogHistoriadeUsuario, CTRL Sprints, CTRL Historia de Usuario.
Resultado esperado	Si se produce un cambio sobre una historia de usuario, el controlador deberá reflejar dicho cambio en el log de la historia de usuario.
Resultado obtenido	El controlador actualiza correctamente cualquier modificación realizada sobre una historia de usuario.
Corrección realizada	No ha sido necesaria realizar ninguna corrección.
Fecha	21/01/2022
Responsable	Manuel Valbuena Alba

Identificador	CP. Unitaria – Cronograma
Descripción	Integración de interfaces.
Componentes afectados	Desarrolladores, viewPrincipal, viewCronograma, CTRL Principal.
Resultado esperado	Al pulsar el botón “Cronograma”, el controlador deberá mostrar el cronograma del usuario con todas las historias de usuario que tiene asignadas.
Resultado obtenido	Al pulsar el botón, se muestra correctamente el cronograma del usuario.
Corrección realizada	No ha sido necesaria realizar ninguna corrección.
Fecha	21/01/2022
Responsable	Manuel Valbuena Alba

6 Conclusión

6.1 Conclusiones

En la realización de este proyecto se han aplicado diferentes conocimientos adquiridos durante el grado, como lo pueden ser la generación del modelo físico para crear la base de datos o el lenguaje MySQL para realizar las consultas sobre la base de datos. Igualmente, también hemos usado algunas herramientas previamente usadas en la carrera como MagicDraw para la elaboración de los distintos diagramas UML, por ejemplo, el modelado de clases o los diagramas de secuencia, entre otros.

MagicDraw sin duda, ha sido la herramienta más fácil de usar durante el desarrollo del proyecto, ya que la única dificultad presentada fue el recordar su uso, así como el manejo de cada uno de los diagramas disponibles dentro de la herramienta. Una vez recordado esto, realizar cada uno de los diagramas fue un proceso muy fácil y fluido.

Para la creación de la base de datos, se ha usado la herramienta SQLyog la cual implementa el lenguaje MySQL, el cual se ha usado para realizar las diferentes consultas. Respecto a esta herramienta, también ha sido novedoso su uso, pero es una herramienta tan sencilla de usar que prácticamente no hace falta seguir ningún tipo de tutorial o consultar documentación alguna para hacer uso de ella, debido a la interfaz tan amigable que dispone.

Por otro lado, este proyecto ha contado con la dificultad de usar el lenguaje de programación PHP, debido que es una tecnología que se ha aprendido desde cero para la realización de este trabajo. Por suerte PHP es un lenguaje que lleva mucho tiempo instaurado como uno de los lenguajes de programación más populares, por lo que ha sido fácil el aprendizaje de este debido a la gran variedad de documentación y ejemplos disponibles. Una de las principales razones para el uso de este lenguaje y no otro, es la facilidad que ofrece para realizar la conexión con la parte del servidor.

Como última conclusión, hay que destacar que se ha podido crear una aplicación web totalmente funcional, que incluye las principales funcionalidades de otras aplicaciones para el desarrollo de proyecto ágiles que actualmente se encuentran en el mercado, con la diferencia que nuestra aplicación, cuenta con una interfaz muy amigable y con una usabilidad muy sencilla para cualquier usuario, independientemente del nivel de informática que posea.

Como se ha mencionado anteriormente, solo conocíamos parte de las herramientas y lenguajes usados en el proyecto, como MagicDraw o MySQL, sin embargo, nunca habíamos trabajado con el lenguaje PHP o con la herramienta de base de datos SQLyog,

por lo que los conocimientos previos a la realización del proyecto se podrían decir que eran 50/50.

La idea de usar PHP como lenguaje de programación web para elaborar la aplicación web de este TFG era para poder aprender a usar uno de los lenguajes más populares hoy en día para el desarrollo web.

Podríamos decir que hemos aprendido bastante sobre todas las tecnologías nuevas usadas en este proyecto, así como hemos reforzado los conocimientos, e incluso adquirido nuevos, sobre las herramientas que si conocíamos previamente.

Otro de los puntos interesantes al realizar este TFG ha sido el hecho de realizar un proyecto entero partiendo desde cero, es decir, a partir de una idea externa, ir realizando poco a poco el proyecto. Inicialmente obteniendo los requisitos y los casos de uso del sistema, pasando por la obtención de los diagramas de clases, de secuencia, modelo físico de datos o el modelado de arquitectura, hasta finalmente realizar una serie de maquetas que servirán posteriormente para la realización de la aplicación web.

6.2 Trabajos futuros

Como posibles mejoras a implementar para conseguir una aplicación más completa, se proponen las siguientes funcionalidades:

- **Buscador:** implementar un buscador que sirva de ayuda para los usuarios para buscar tanto proyectos, como otros usuarios que estén registrados en la aplicación para poder establecer comunicación con ellos.
- **Backlog:** implementar un backlog cuyo uso principal será el de llevar un mejor seguimiento y control de las historias de usuario pertenecientes a un mismo proyecto.
- **Wiki:** crear una wiki para que los miembros de un equipo de desarrollo tengan disponible un lugar donde encontrar la definición de tecnicismos que tengan que usar en el proyecto, y que no estén familiarizados con el término.
- **Log de historia de usuario:** a diferencia del log que ya existe, este se enfocaría en cada una de las tareas de forma individual, es decir, se reflejarían todas las modificaciones que se hayan producido única y exclusivamente en una historia de usuario.

Bibliografía

- [1] ProyectosAgiles, ¿Qué es SCRUM? <https://proyectosagiles.org/que-es-scrum/>
- [2], [3], [4] UML 2. Iniciación, ejemplos y ejercicios corregidos. Debrauwer, L. y Van der Heyde, F.; ENI ediciones, 2013;
- [5] Junta de Andalucía, Procedimiento para desarrollar los requisitos de un sistema software que satisfaga las necesidades de negocio
<http://www.juntadeandalucia.es/servicios/madeja/contenido/procedimiento/20#:~:text=Requisitos%20de%20informaci%C3%B3n%2C%20describen%20la,cumplimiento%20debe%20velar%20el%20sistema>
- [6] IBM, Definición de casos de uso
<https://www.ibm.com/docs/es/elm/6.0.3?topic=requirements-defining-use-cases>
- [7] Manuel Cillero, Diagrama de clases
<https://manuel.cillero.es/doc/metodologia/metrica-3/tecnicas/diagrama-de-clases/>
- [8] Manuel Cillero, Diagrama de secuencia
<https://manuel.cillero.es/doc/metodologia/metrica-3/tecnicas/diagrama-de-interaccion/diagrama-de-secuencia/>
- [9] IBM, Modelo físico de datos <https://www.ibm.com/docs/es/data-studio/4.1.1?topic=modeling-physical-data-models>
- [10] Carlos Manuel Rossi Jiménez, Diagramas de arquitectura, Curso 19-20
- [11] Ulises Villanueva Resendiz, Maquetado de sistemas
<https://es.slideshare.net/UlisesVillanueva1/maquetado-110798534>
- [12] Balsamiq, Balsamiq Wireframes <https://balsamiq.com/wireframes/>
- [13], [14] Webyog, SQLyog <https://webyog.com/product/sqlyog/>
- [15] <https://code.visualstudio.com/docs/supporting/FAQ>
- [16] Visual Studio Code, Visual Studio Code FAQ
<https://www.php.net/manual/es/intro-what-is.php>
- [17] Platzi, Ventajas y desventajas de Ajax y como usarlo con jQuery
<https://platzi.com/blog/ajax-con-jquery/>
- [18] JQuery <https://jquery.com/>
- [19] Bootstrap <https://getbootstrap.com/>
- [20] Desarrolloweb, Manual de PHP: Laragon
<https://desarrolloweb.com/articulos/laragon.html>
- [21] JQWidgets <https://www.jqwidgets.com/jquery-widgets-demo/>
- [22] Medium, ¿Qué son los casos de prueba?
<https://medium.com/grupo-carricay/qu%C3%A9-son-los-casos-de-pruebas-4893799b5b84>

A. Álvarez, R. de las Heras, C. Lasa. Métodos Ágiles y Scrum. Anaya, 2012

Booch, G., Rumbaugh, J. y Jacobson, I. El lenguaje Unificado de Modelado. Addison Wesley, 2009.

Anexo

Manual de implantación

Requisitos

Para poder ejecutar la aplicación web, el servidor debe tener instalado y configurado el siguiente software:

- Servidor de páginas web.
- Servidor de bases de datos Mysql.
- PHP como lenguaje de programación de script en el servidor.

Concretamente, para la realización de esta aplicación, el servidor que he utilizado tiene la siguiente configuración.

- Servidor de páginas web: Apache 2.4.47.
- Servidor de bases de datos: Mysql 5.7.33
- Versión de PHP 7.4.19

Instalación

Para realizar el proceso de instalación hay que realizar dos pasos:

1. Copiar toda la estructura de ficheros y directorios en el directorio de publicación del sitio web destinado a la aplicación web.
2. Importar la estructura de la base de datos a Mysql. Para realizar este proceso de importación, en la carpeta sql hay un fichero llamado importar.sql, en cual tiene toda la estructura de la base de datos.

La base de datos que se importa se llama proyectos. Esta base de datos tiene un usuario llamado proyecto, con la siguiente contraseña ProyectoMVA.

Para que funcione correctamente la aplicación, en el servidor de bases de datos Mysql se debe crear dicho usuario con la contraseña y darle todos los privilegios sobre la base de datos proyectos.

Si se quiere utilizar un usuario distinto se debe modificar la configuración de la conexión a la base de datos. Para ello se debe acceder al fichero funciones.php. En dicho fichero se ha de localizar una función llamada conectar(). Esta función recibe los siguientes parámetros:

- Nombre del servidor, por defecto localhost.
- Usuario con privilegios sobre la base de datos, por defecto proyecto.
- Contraseña de dicho usuario, por defecto ProyectoMVA.
- Nombre de la base de datos, por defecto proyecto.

Una vez realizados los cambios se ha de guardar el fichero.

Anexo

Manual de usuario

Para acceder a la aplicación web se debe abrir un navegador de internet y en la barra de dirección teclear la URL de la aplicación. Si todo ha funcionado correctamente, debe aparecer la pantalla de inicio de sesión.



Figura 1 Anexo: Vista "Login" de la aplicación web.

Al importar la base de datos, solamente hay creado un usuario con permisos de administrador. Este debe ser el usuario con el cual se ha de iniciar sesión la primera vez. Dicho usuario irá invitando al resto de usuarios que tendrán acceso a la aplicación.

Los datos de acceso de este usuario son:

- Correo electrónico: admin@admin.com
- Contraseña: 123456789

Una vez iniciada la sesión, se mostrará el panel de gestión de la aplicación web. Este panel de gestión tiene tres configuraciones, dependiendo del perfil del usuario que accede a la aplicación.

Si se accede con el perfil de administrador, el panel de gestión tiene las siguientes opciones:

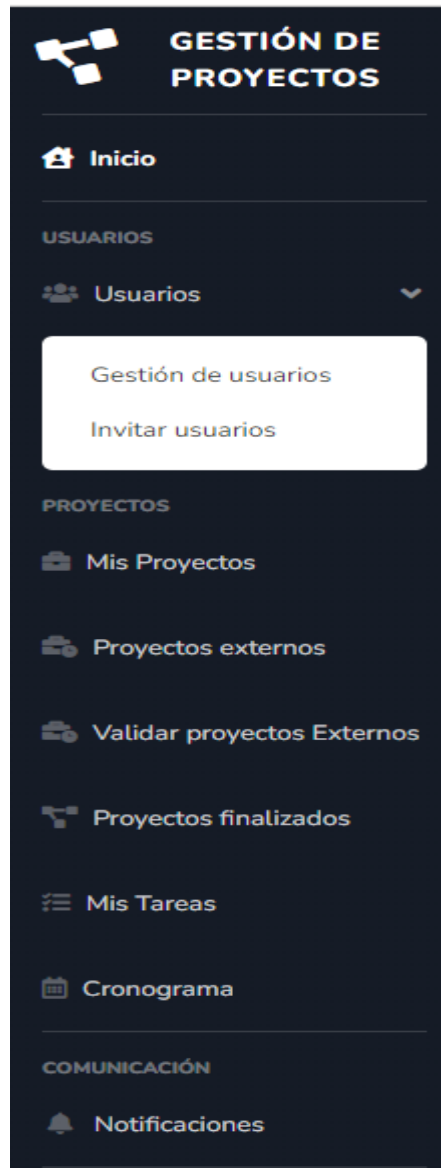


Figura 2 Anexo: Panel de gestión del perfil Administrador

Dentro de Gestión de usuarios, al administrador le aparecerá una vista con todos los usuarios que forman parte de la aplicación, así como tres botones: crear, modificar y eliminar usuario. Los dos últimos, basta con seleccionar un usuario y pulsar el botón deseado.

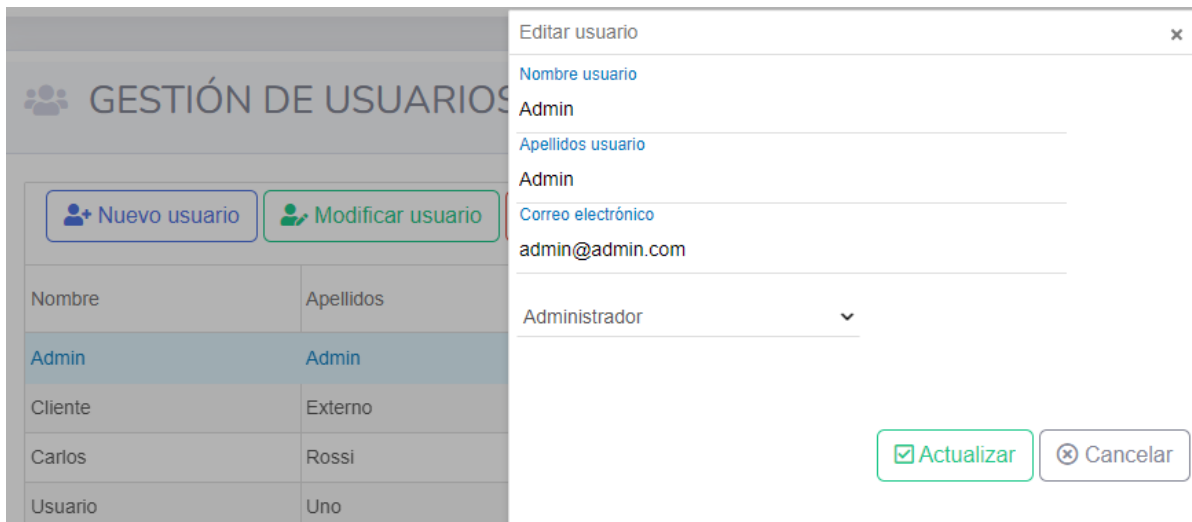


Figura 3 Anexo: Modificar usuario

Respecto a la creación, aparecerá una ventana emergente con una serie de datos a rellenar por el administrador y una opción para, además de crear el usuario, enviar una invitación vía email.



Figura 4 Anexo: Crear usuario

Si se accede con el perfil de usuario, el panel de gestión tiene las siguientes opciones:

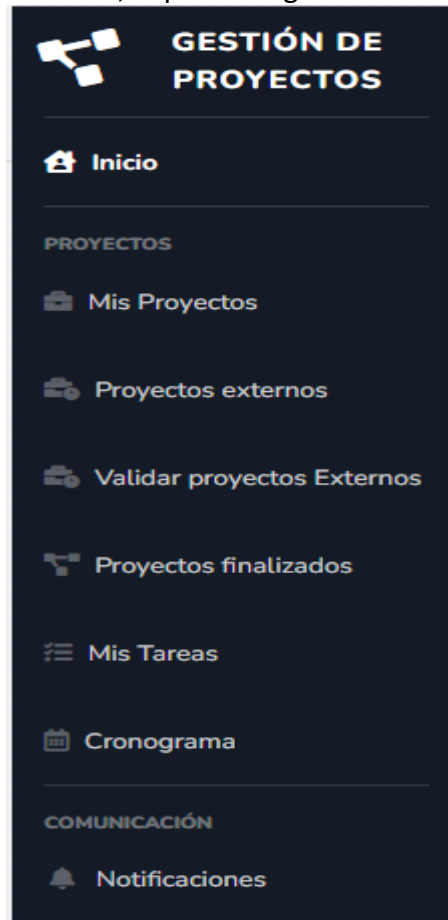


Figura 5 Anexo: Panel de gestión del perfil Usuario

Los usuarios que formen parte del equipo de desarrollo tendrán una serie de opciones vistas en la figura anterior, entre las que destaca la opción "Mis proyectos", dónde los usuarios podrán crear proyectos.

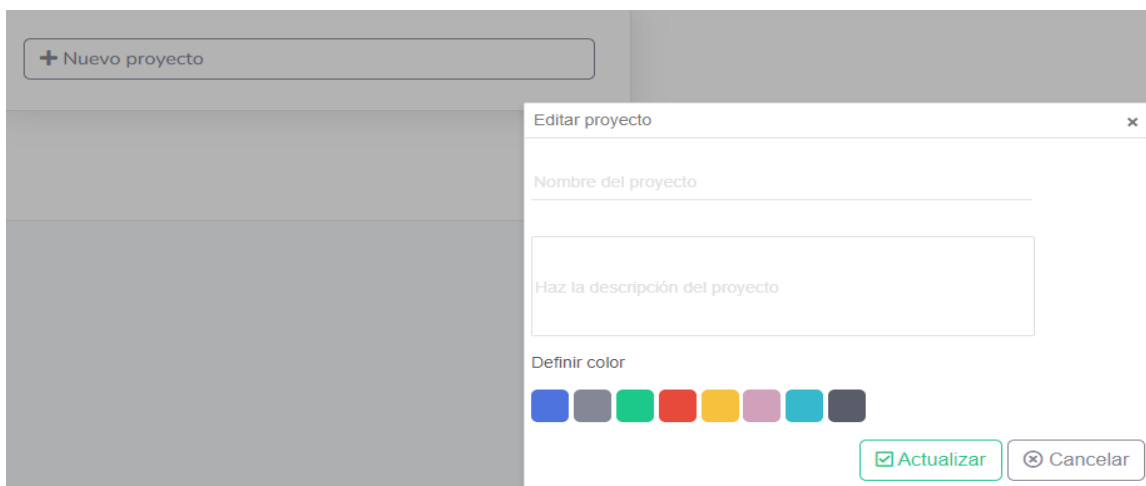


Figura 6 Anexo: Crear proyecto

Una vez creado el proyecto, aparecerá el proyecto dentro de la vista “Mis proyectos”. Cada proyecto tendrá un desplegable donde se podrá modificar el proyecto, así como crear Sprints, gestionar los miembros participantes del equipo, ver el historial de modificaciones producidas sobre el proyecto y, por último, dar por finalizado el proyecto.

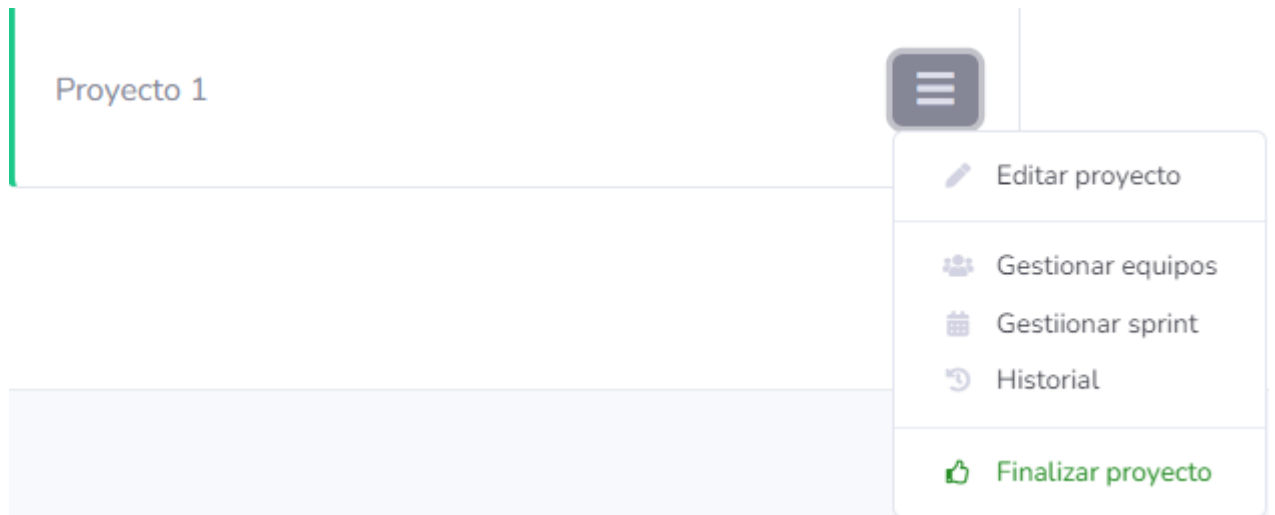


Figura 7 Anexo: Gestionar proyectos

#		
1	05/02/2022	Se ha creado la tarea Descripción de la tarea 1
2	05/02/2022	Se ha asignado la tarea Descripción de la tarea 1 a Manolo Valbuena Alba
3	05/02/2022	Se ha subido el fichero indice.txt a la tarea Tarea 1:

Figura 8 Anexo: Historial

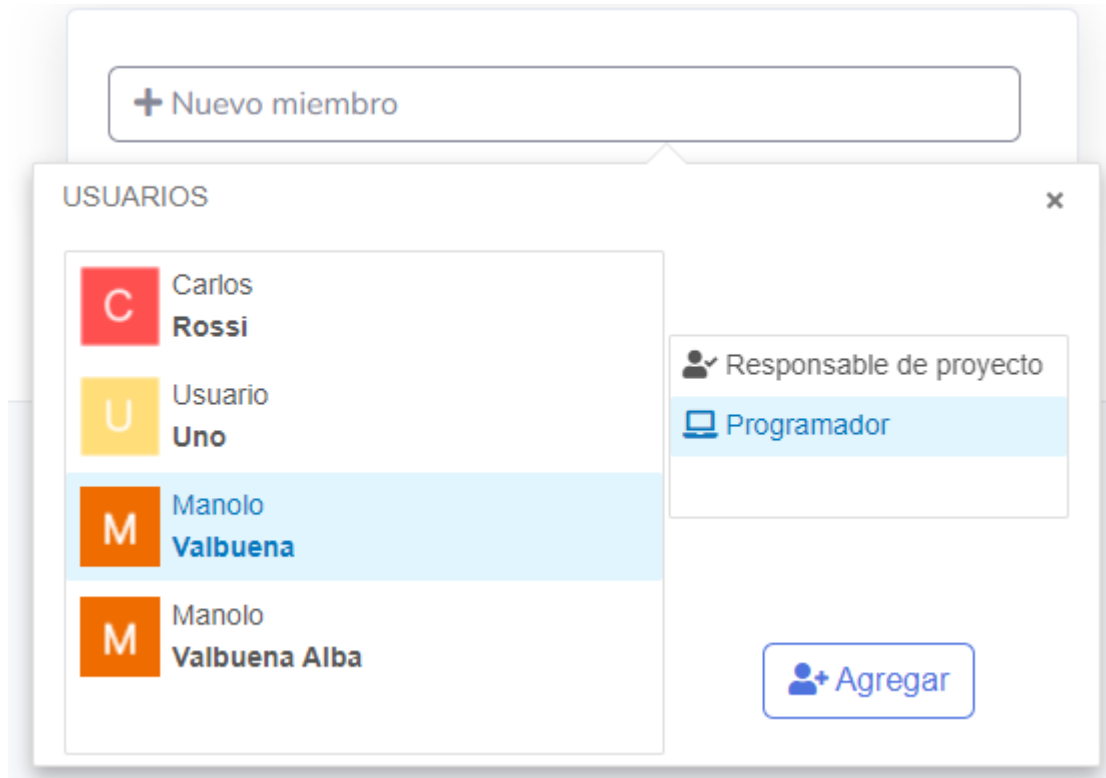


Figura 9 Anexo: Gestionar equipo

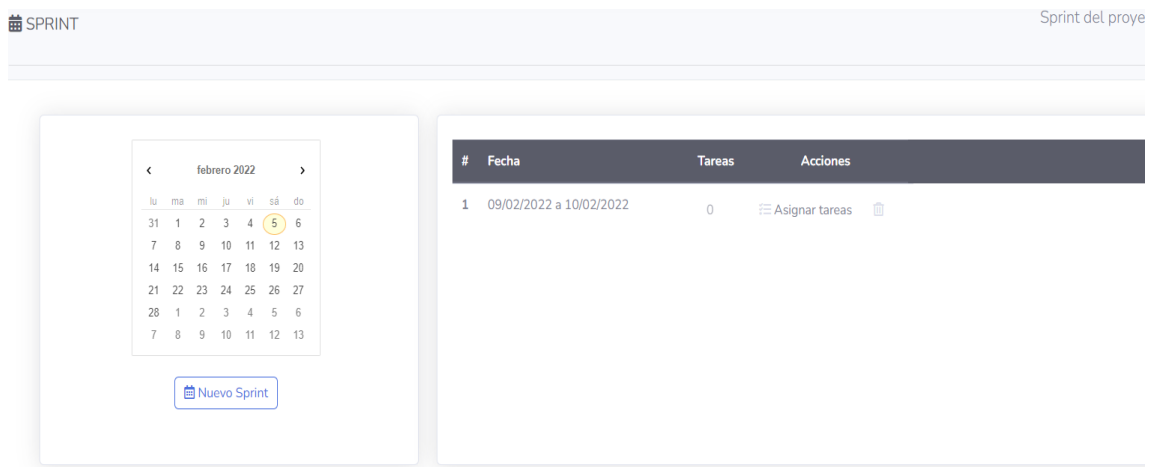


Figura 10 Anexo: Gestionar Sprints.

Pulsando sobre Asignar tareas, accederemos a la gestión de historias de usuario, dónde podremos crear, modificar, asignar responsable, subir archivos y eliminar historias de usuario.

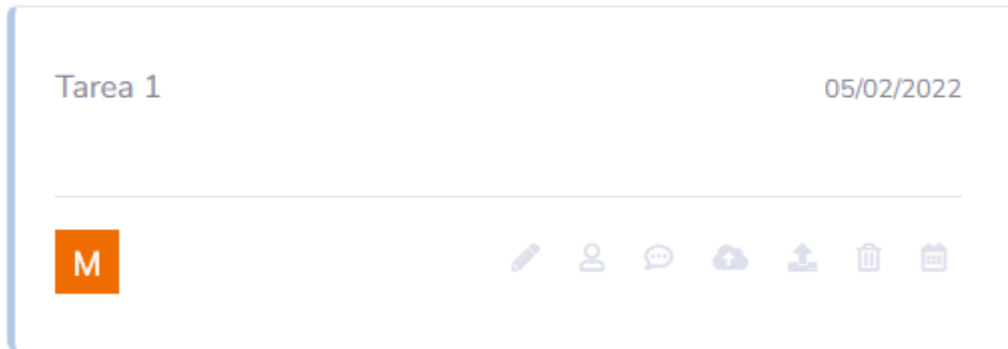


Figura 11 Anexo: Gestionar historias de usuario.

Los usuarios también contarán con las opciones de ver un Kanban para poder gestionar mejor el estado de las historias de usuario, de las cuales es responsable y un cronograma para establecer una organización a la hora de realizar y completar las historias de usuario.

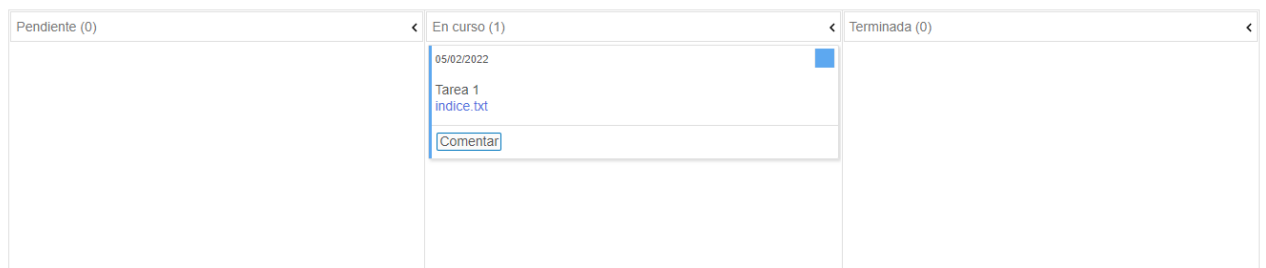


Figura 12 Anexo: Kanban Mis tareas

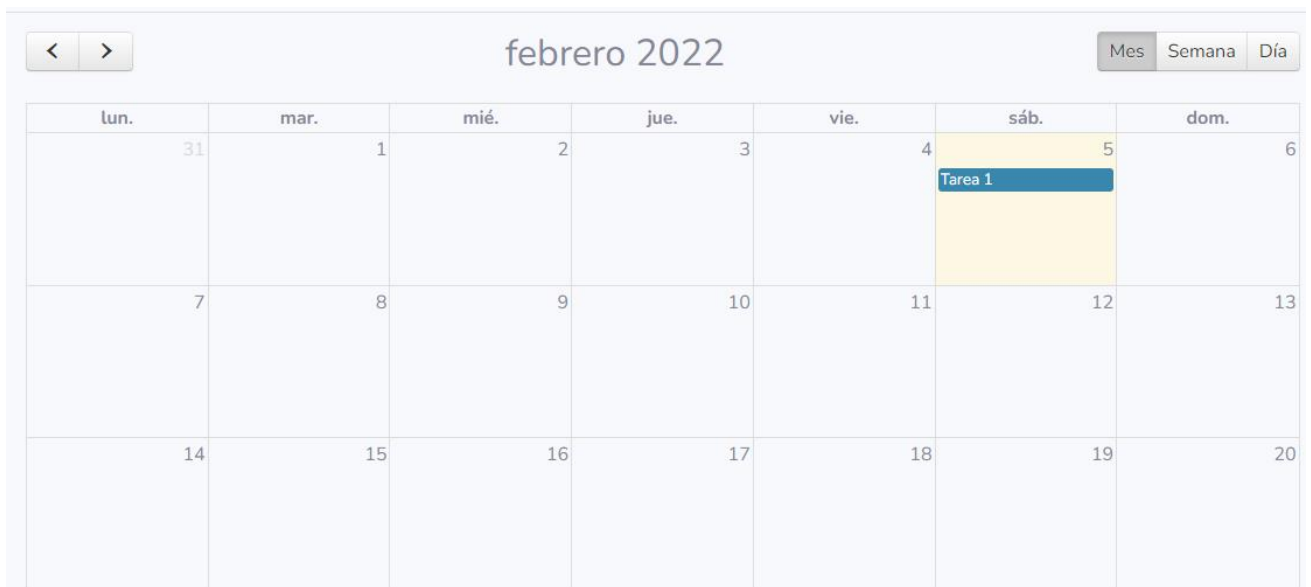


Figura 13 Anexo: Cronograma

Por último, si el usuario tiene el rol de Jefe de proyecto, podrá tanto asignar proyectos externos solicitados por clientes en la opción Proyecto externos, y validar proyectos externos en la opción Validar proyectos externos.

#	Proyecto	Cliente externo	
1	Proyecto externo 1	Cliente Externo	<button>Asignarme proyecto</button>

Figura 14 Anexo: Asignar proyectos externos.

Si se accede con el perfil de cliente, el panel de gestión tiene las siguientes opciones:



Figura 15 Anexo: Panel de gestión del perfil Cliente

En la primera de ellas, los clientes podrán solicitar un proyecto para que sea desarrollado por un equipo. La vista generada es muy similar a la de crear proyecto.

Nombre del proyecto

Haz la descripción del proyecto

Definir color

Solicitar

Figura 16 Anexo: Solicitar proyecto

En la opción Validar proyectos, los clientes deberán validar los proyectos que se han asignado a un equipo de desarrollo. Para la validación deberán pulsar el botón validar, y una vez pulsado, deberán validar una a una todas las historias de usuario que hayan creado el equipo de desarrollo.

#	Proyecto	Responsable	Tareas
1	Proyecto externo 1	Cliente Externo	<input type="button" value="Validar"/>

Figura 17 Anexo: Validar proyecto

Para finalizar todos los usuarios tendrán como opciones comunes: en la esquina superior derecha de la aplicación aparecen las opciones clásicas de perfil de cualquier aplicación: ver y modificar tu perfil, cambiar tu contraseña de acceso y cerrar la sesión.

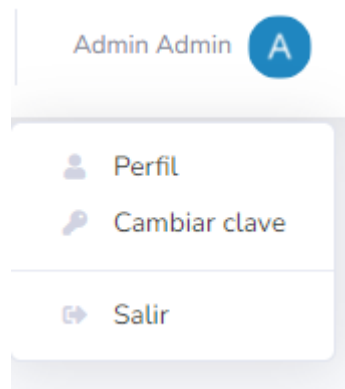


Figura 18 Anexo: Opciones de perfil

Y, tanto en la esquina superior derecha, justo antes del botón del perfil, y en el menú lateral una opción con todas las notificaciones recibidas por un usuario. Las notificaciones cuentan con dos posibles acciones: marcar la notificación como vista o eliminar la notificación.













#	Fecha	Notificación	Acciones
1	05/02/2022	Has sido invitado a participar en el proyecto: Proyecto 1 en el puesto: Programador	 
2	05/02/2022	Eres el responsable de la tarea: Descripción de la tarea 1	 
3	05/02/2022	Eres el responsable de la tarea: Tarea 1	 
4	05/02/2022	Has sido invitado a participar en el proyecto: Proyecto externo 1 en el puesto: Responssble de proyecto	 
5	05/02/2022	Eres el responsable de la tarea: Tarea 1	 
6	05/02/2022	Eres el responsable de la tarea: Tarea 2	 

Figura 19 Anexo: Notificaciones

Anexo

Maquetas de interfaz de usuario



Figura 1 Anexo MIU: Maqueta de interfaz de usuario "Login"

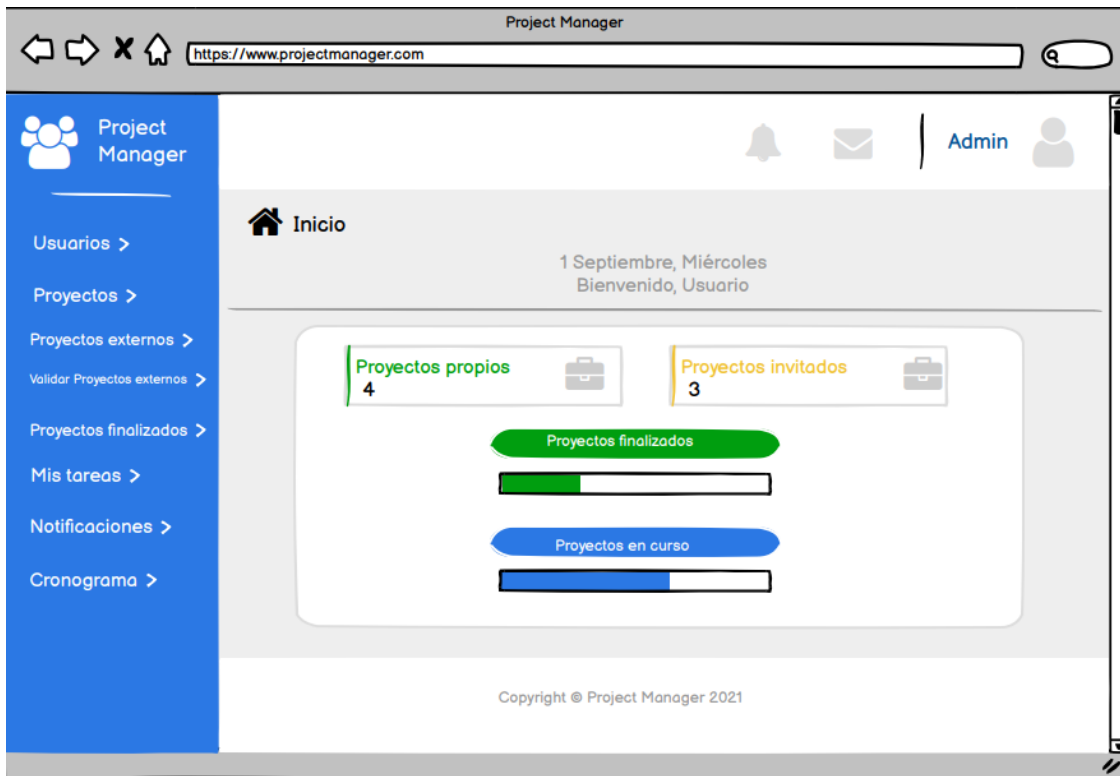


Figura 2 Anexo MIU: Maqueta de interfaz de usuario "Principal"

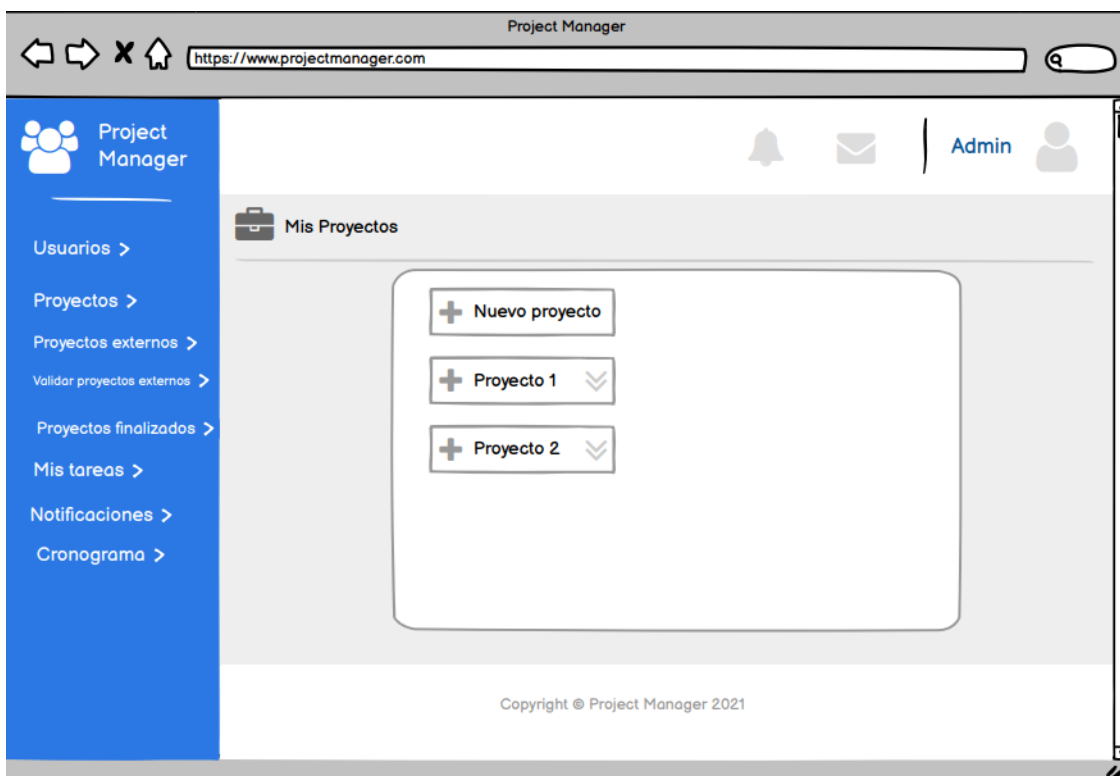


Figura 3 Anexo MIU: Maqueta de interfaz de usuario "Mis proyectos"

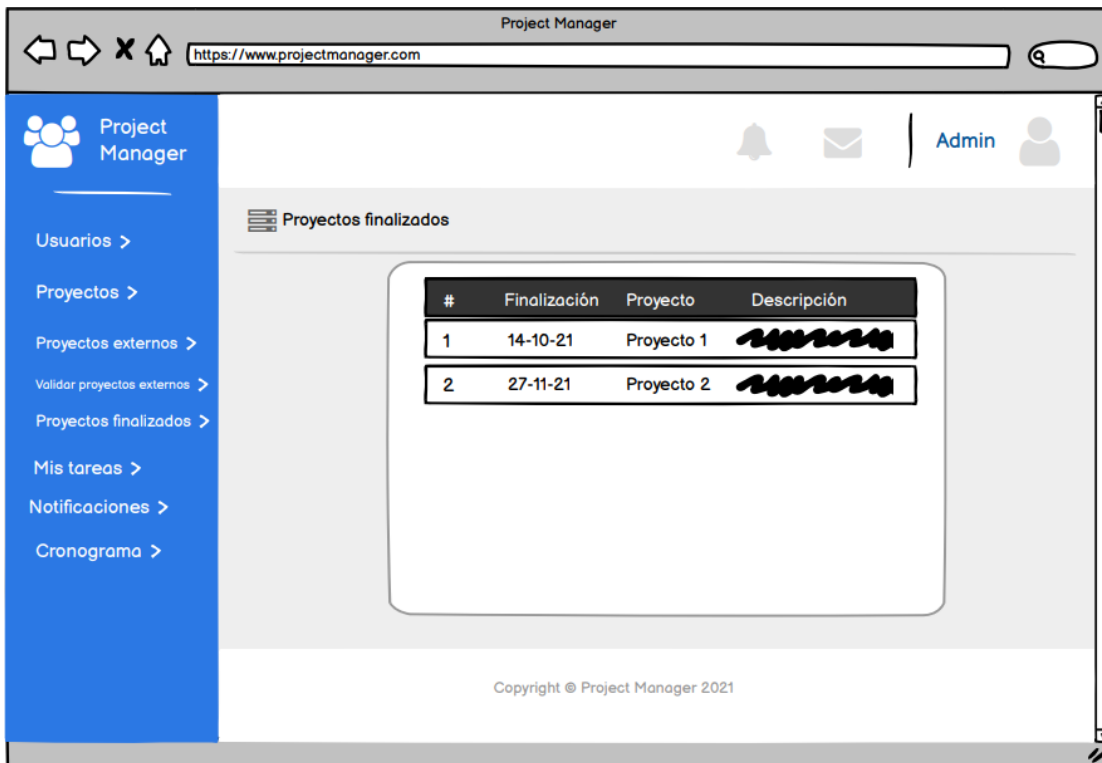


Figura 4 Anexo MIU: Maqueta de interfaz de usuario "Proyectos finalizados"

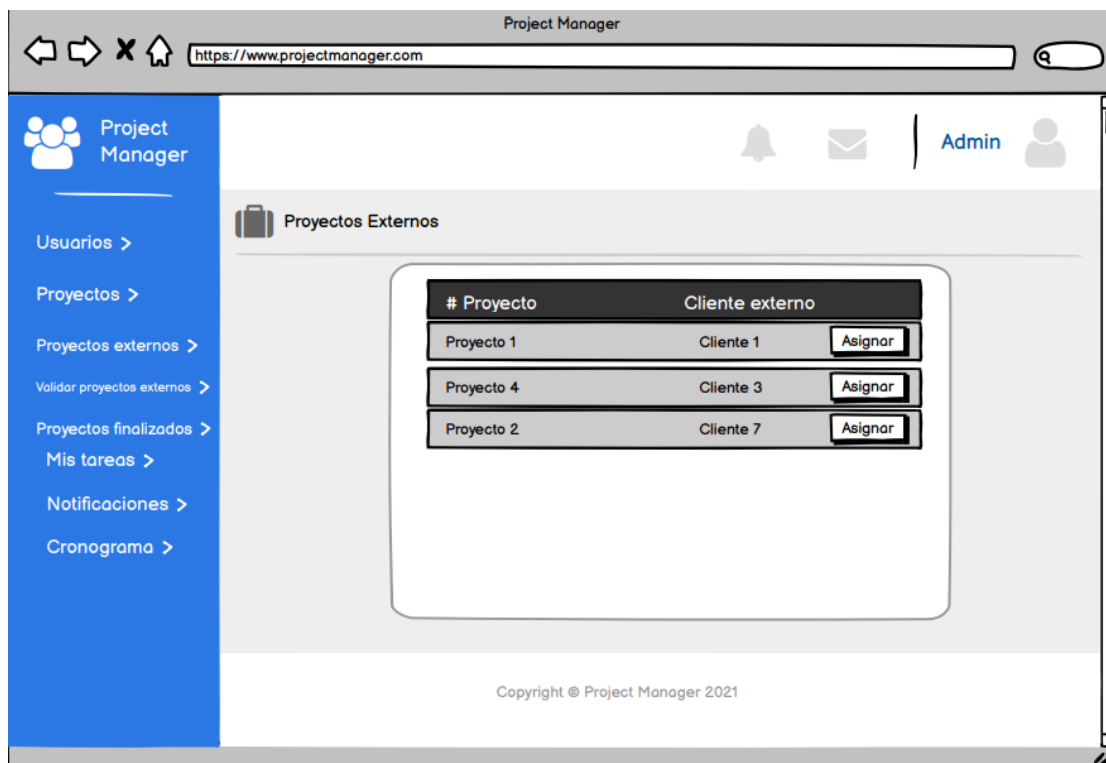


Figura 5 Anexo MIU: Maqueta de interfaz de usuario "Proyectos externos"

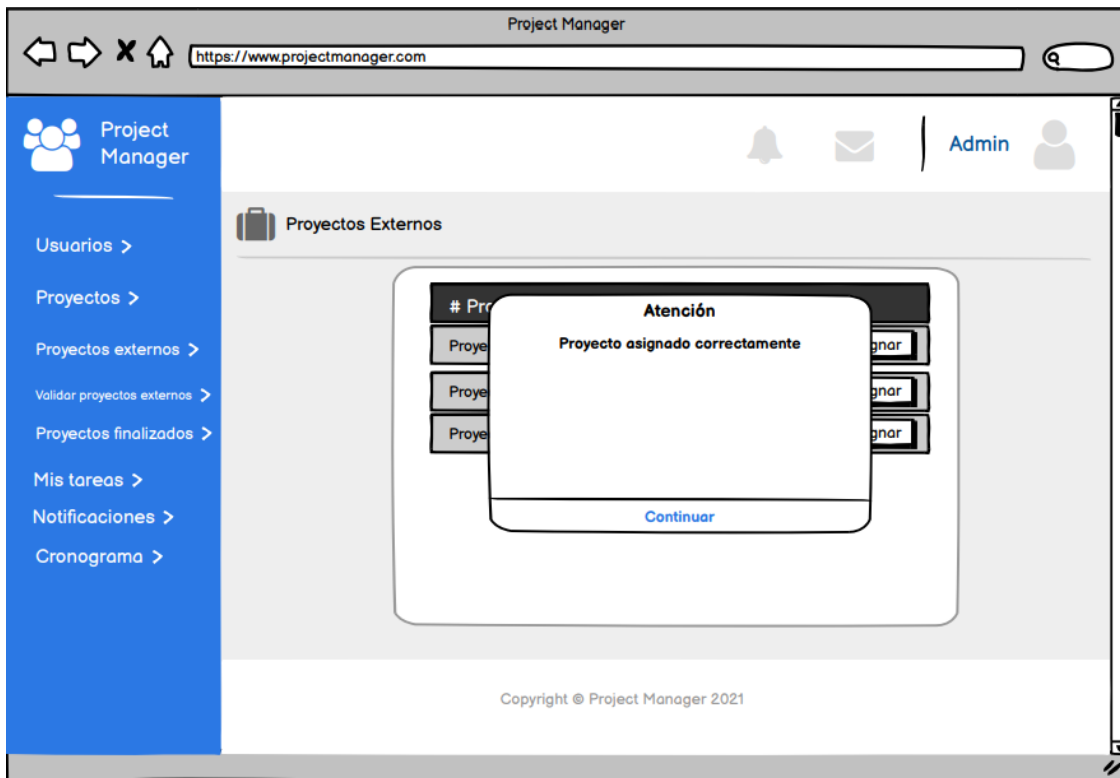


Figura 6 Anexo MIU: Maqueta de interfaz de usuario "Proyectos externo asignado"

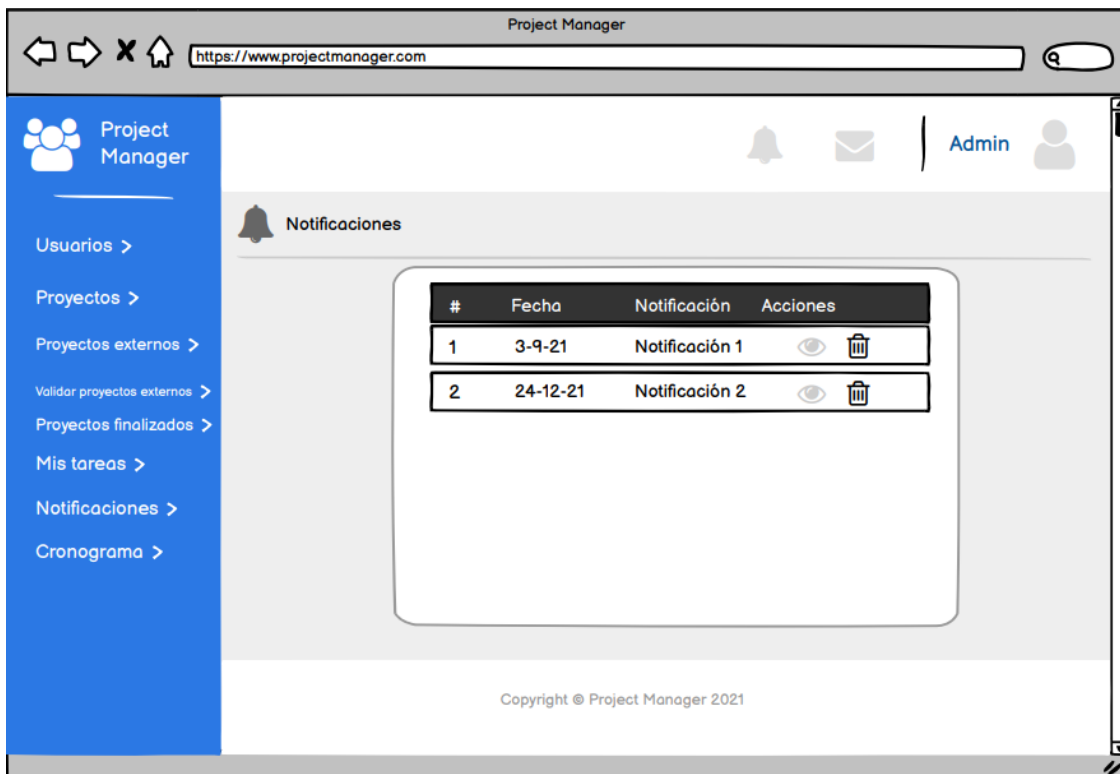


Figura 7 Anexo MIU: Maqueta de interfaz de usuario "Panel de notificaciones"

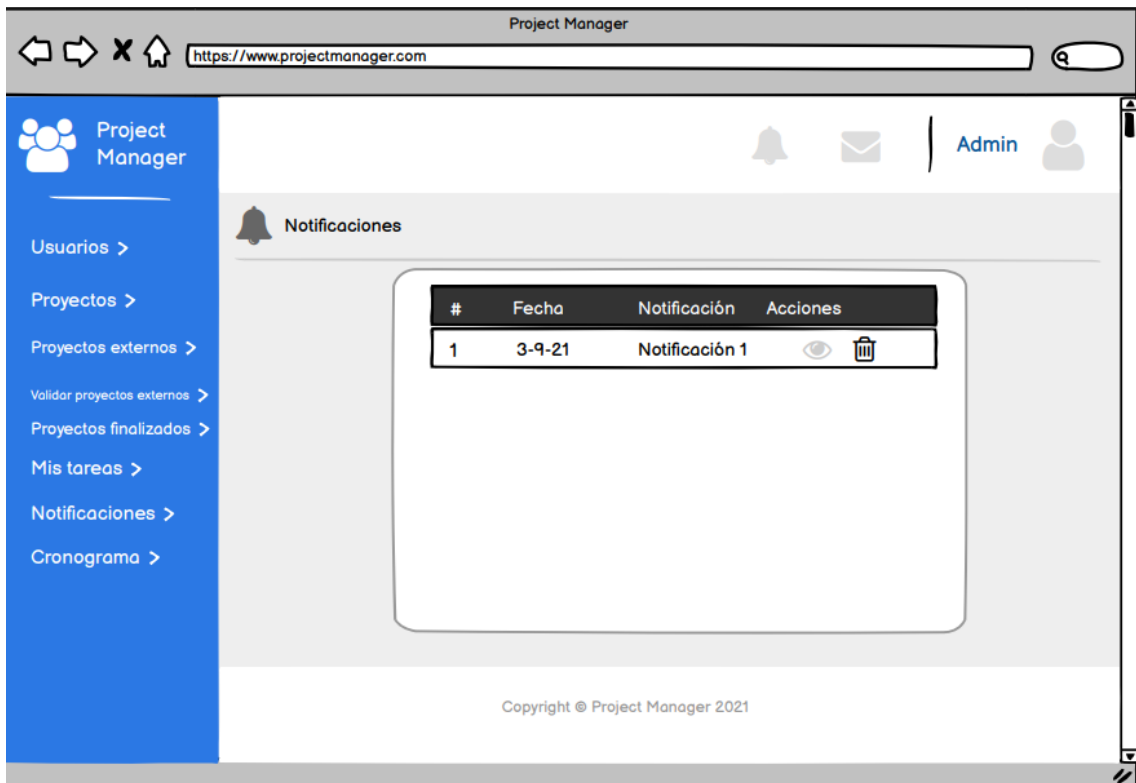


Figura 8 Anexo MIU: Maqueta de interfaz de usuario "Notificación eliminada"

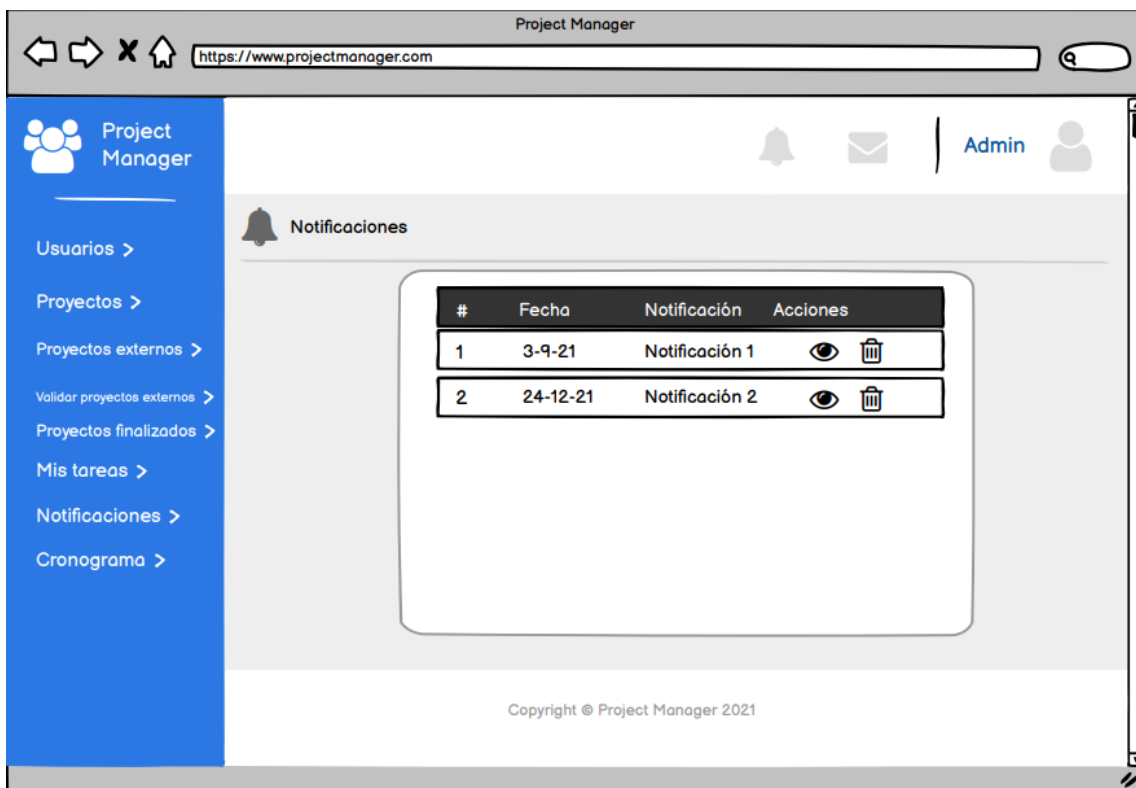


Figura 9 Anexo MIU: Maqueta de interfaz de usuario "Notificación vista"

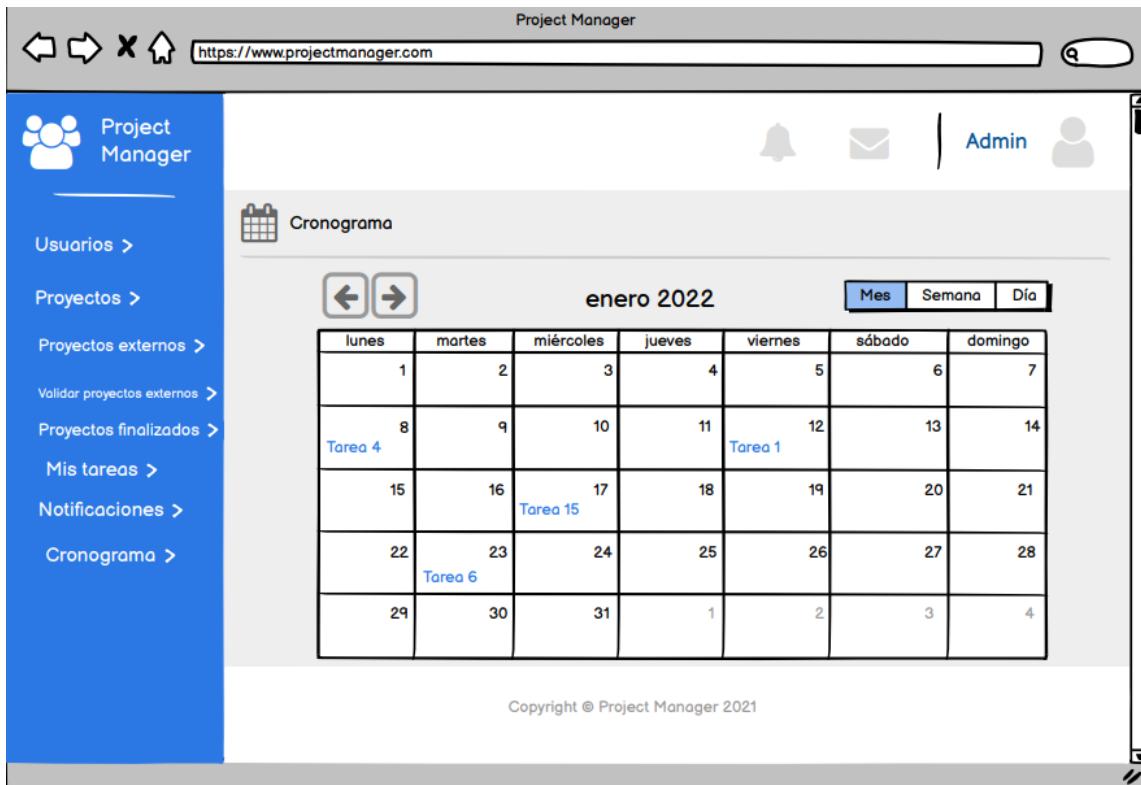


Figura 10 Anexo MIU: Maqueta de interfaz de usuario "Cronograma"

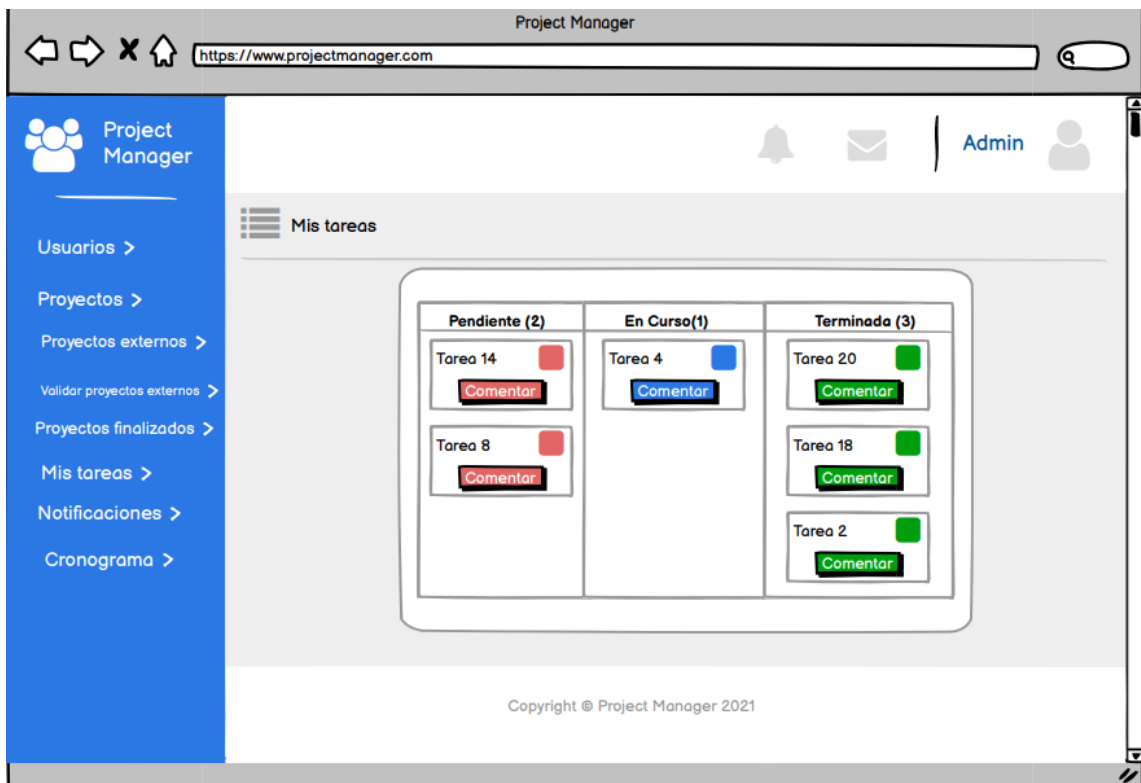


Figura 11 Anexo MIU: Maqueta de interfaz de usuario "Mis tareas"

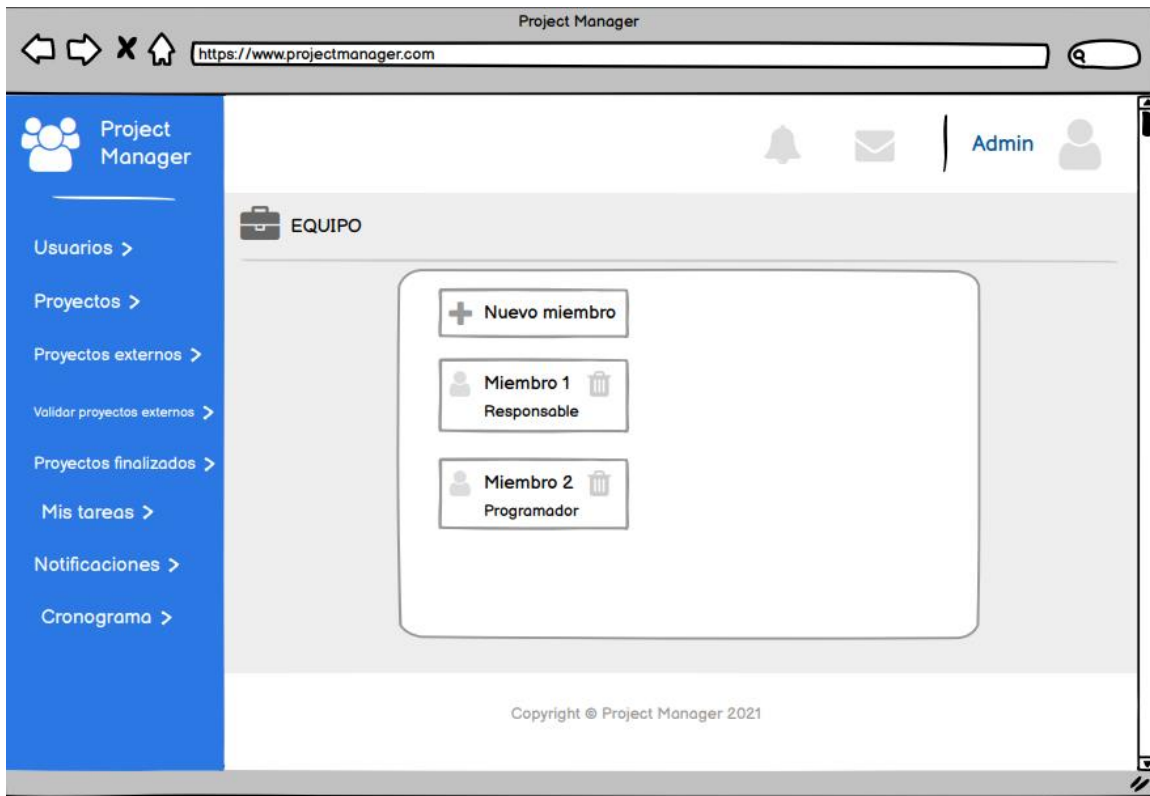


Figura 12 Anexo MIU: Maqueta de interfaz de usuario "Equipo"

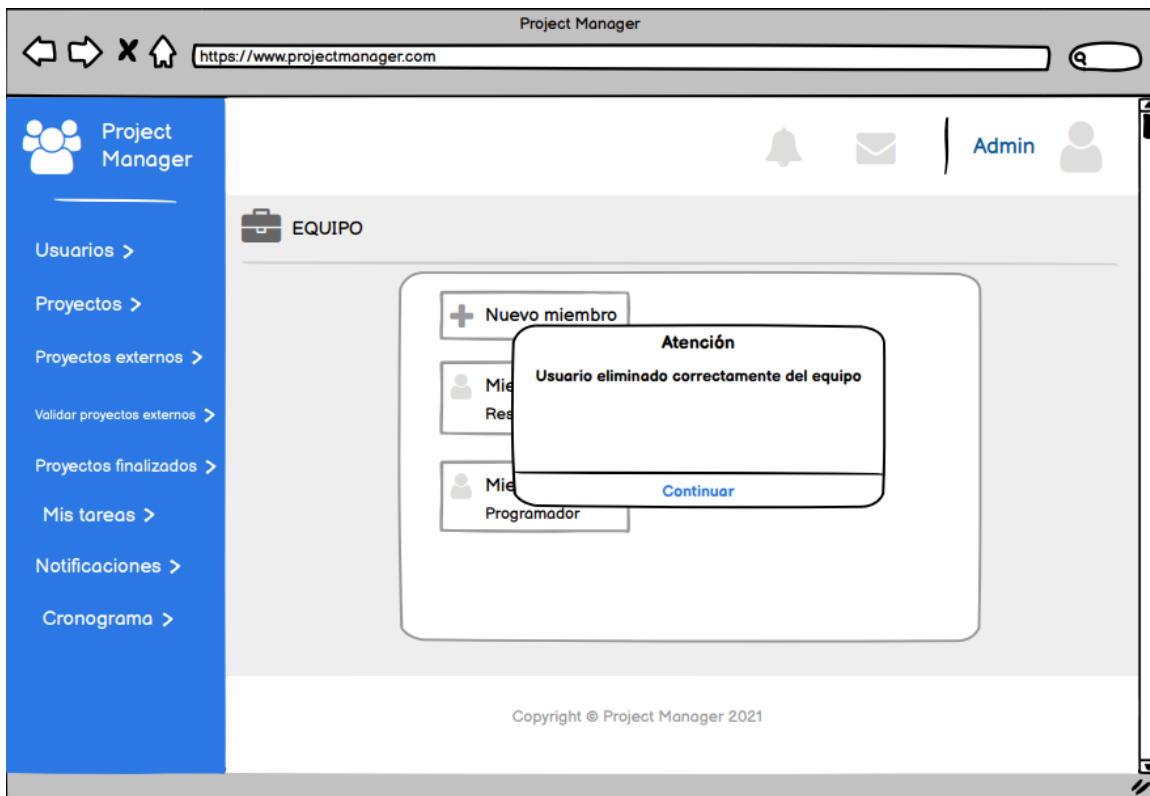


Figura 13 Anexo MIU: Maqueta de interfaz de usuario "Eliminar usuario de equipo"

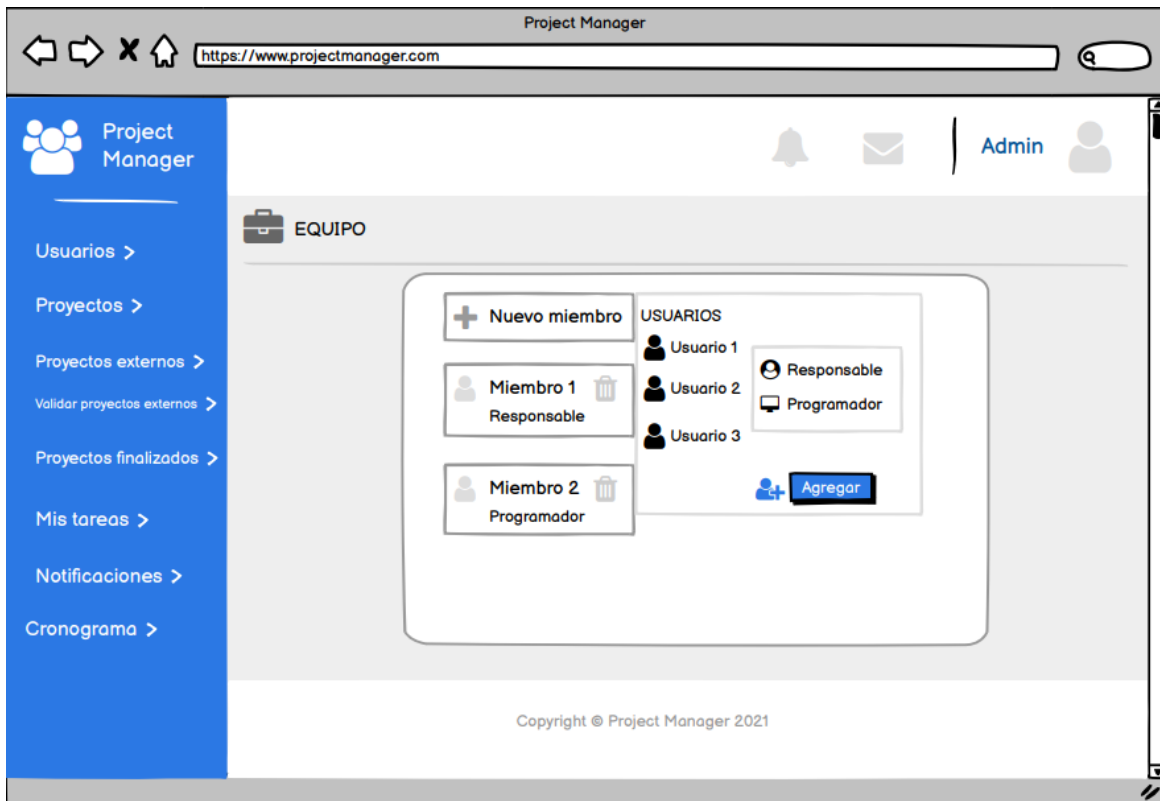


Figura 14 Anexo MIU: Maqueta de interfaz de usuario “Añadir usuario al equipo”

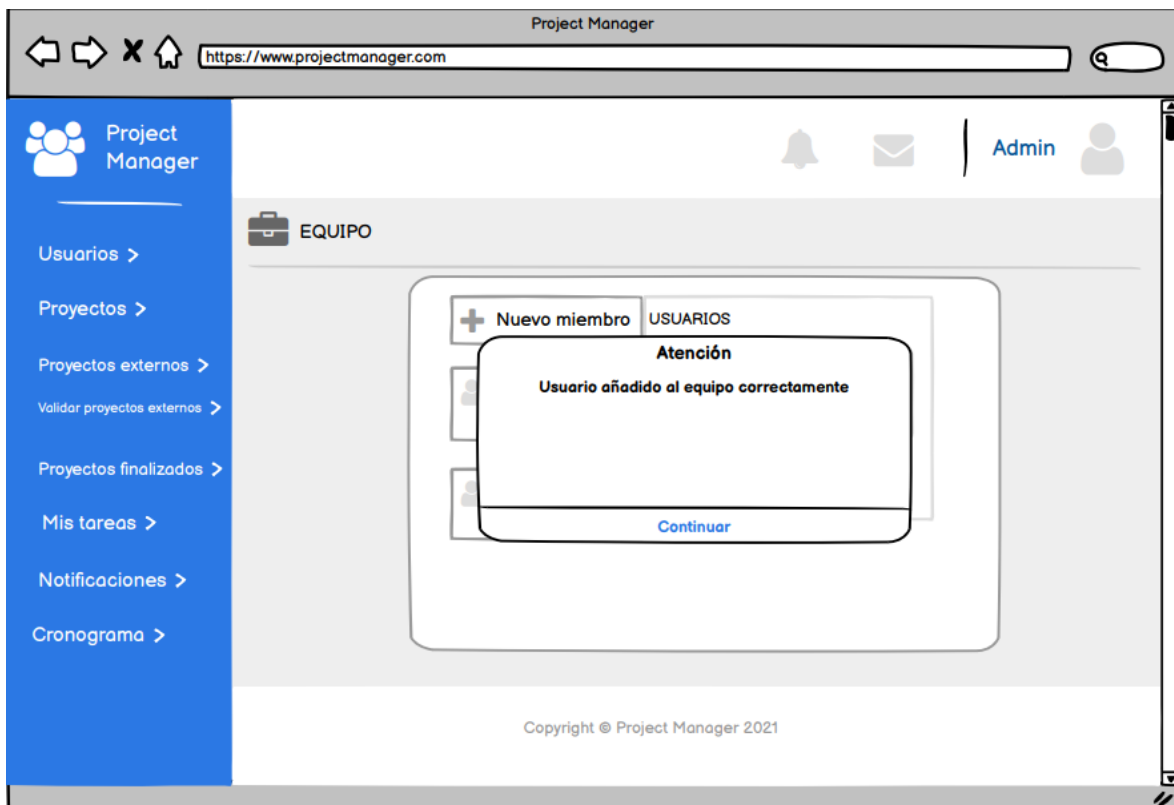


Figura 15 Anexo MIU: Maqueta de interfaz de usuario “Usuario añadido correctamente”

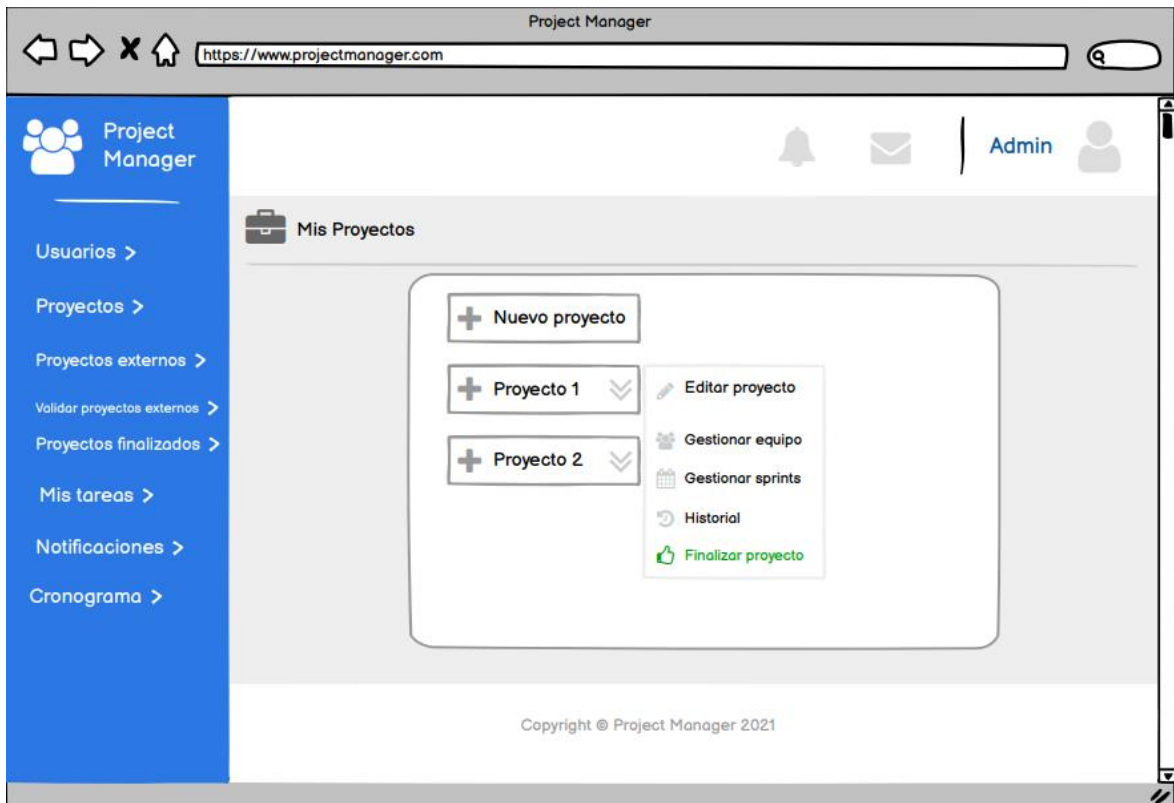


Figura 16 Anexo MIU: Maqueta de interfaz de usuario "Opciones de proyecto"

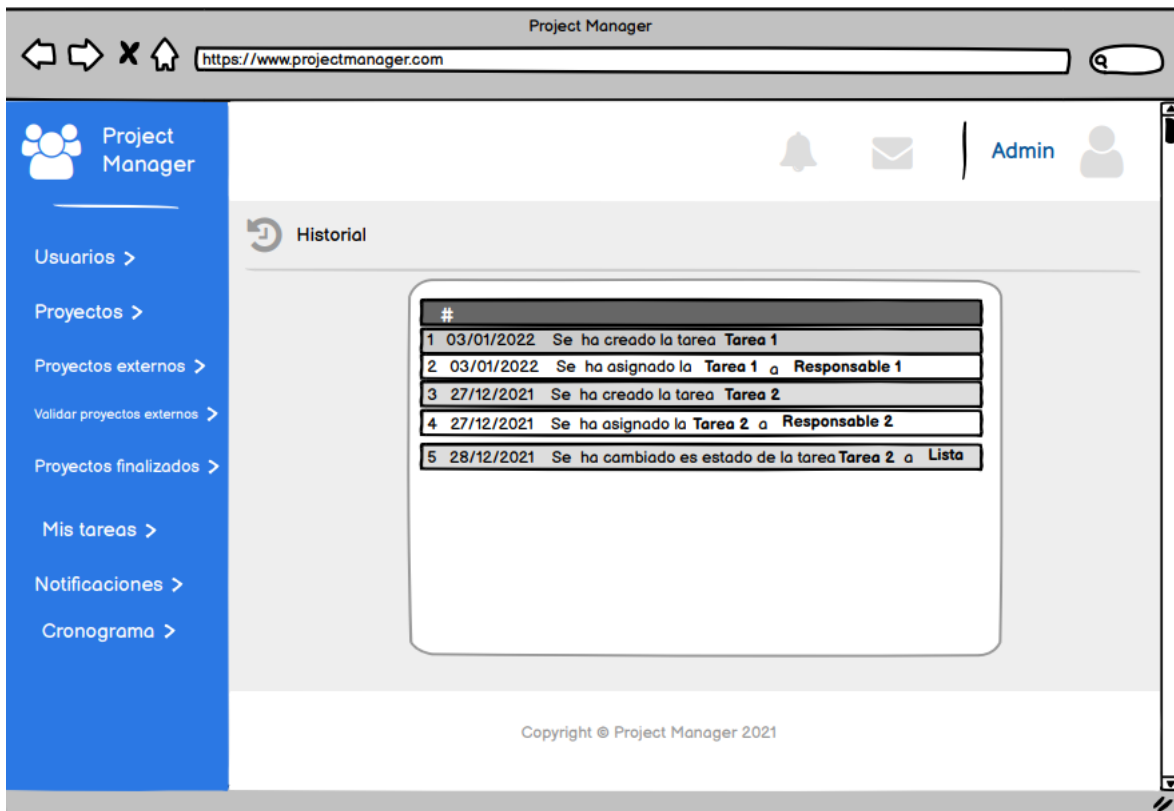


Figura 17 Anexo MIU: Maqueta de interfaz de usuario "Historial"

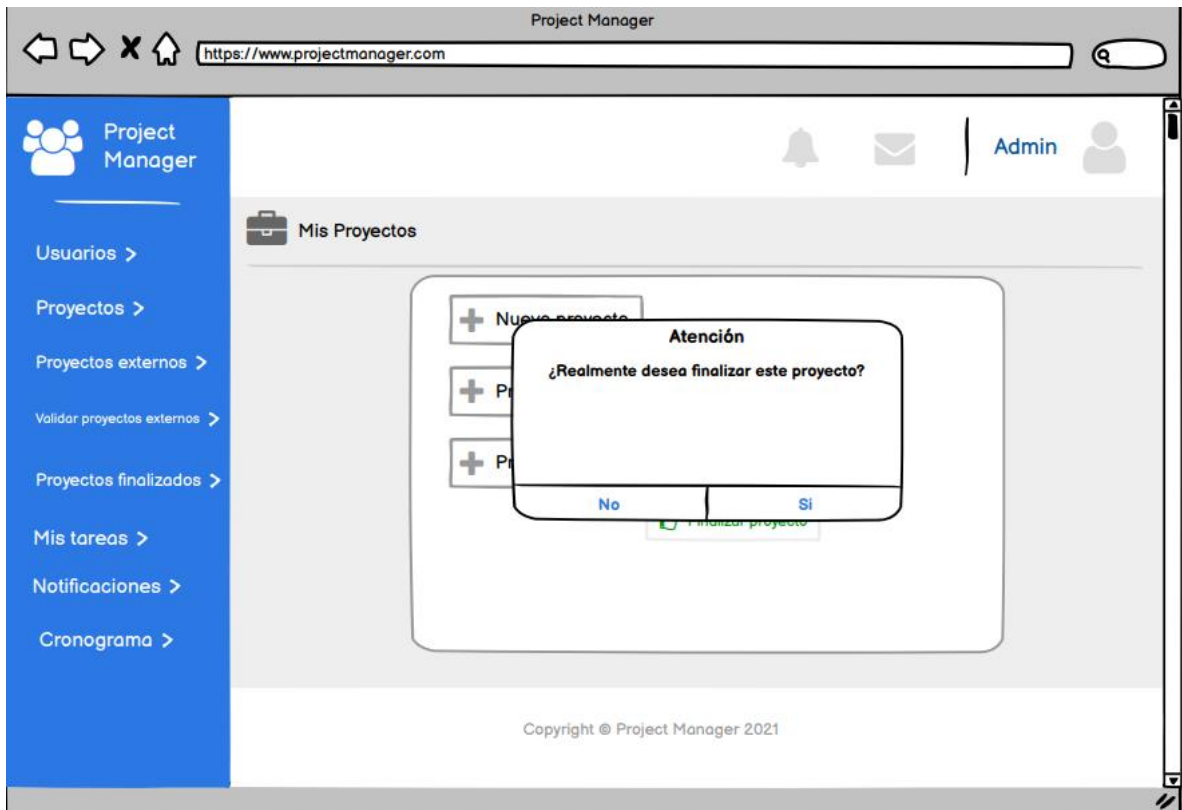


Figura 18 Anexo MIU: Maqueta de interfaz de usuario “Finalizar proyecto”

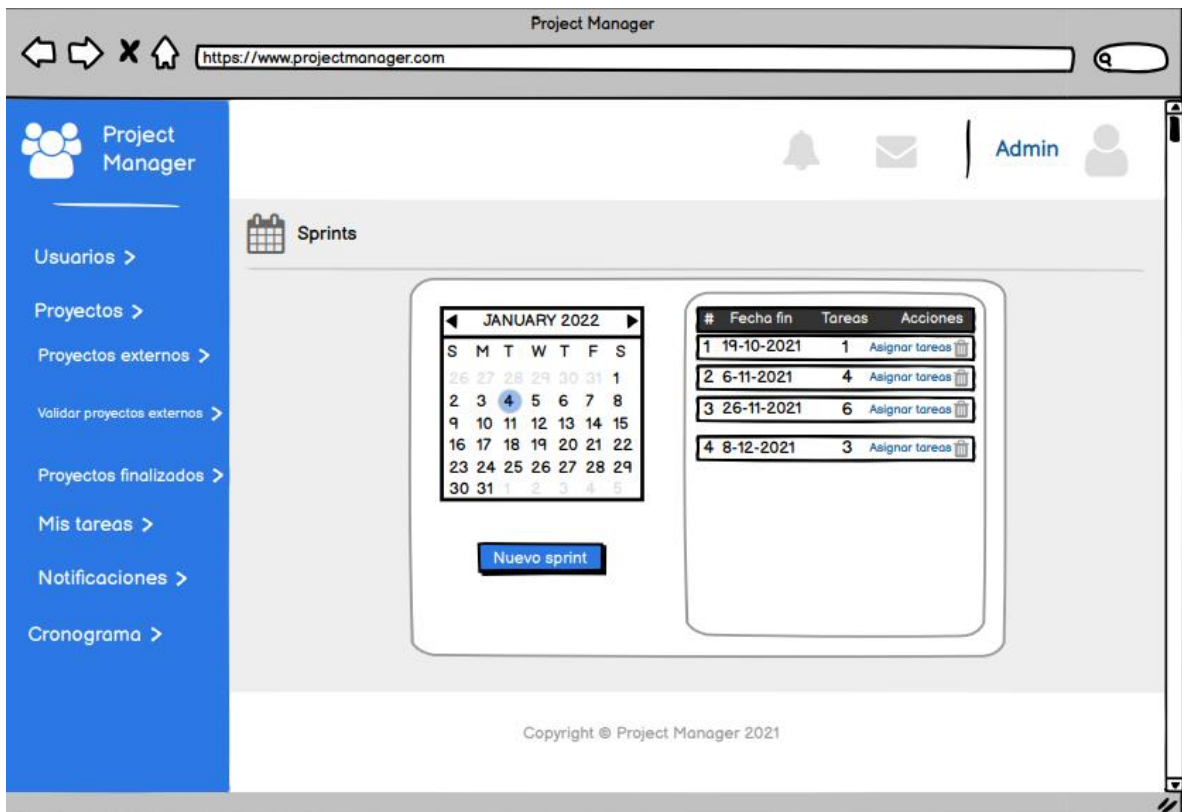


Figura 19 Anexo MIU: Maqueta de interfaz de usuario “Sprints”

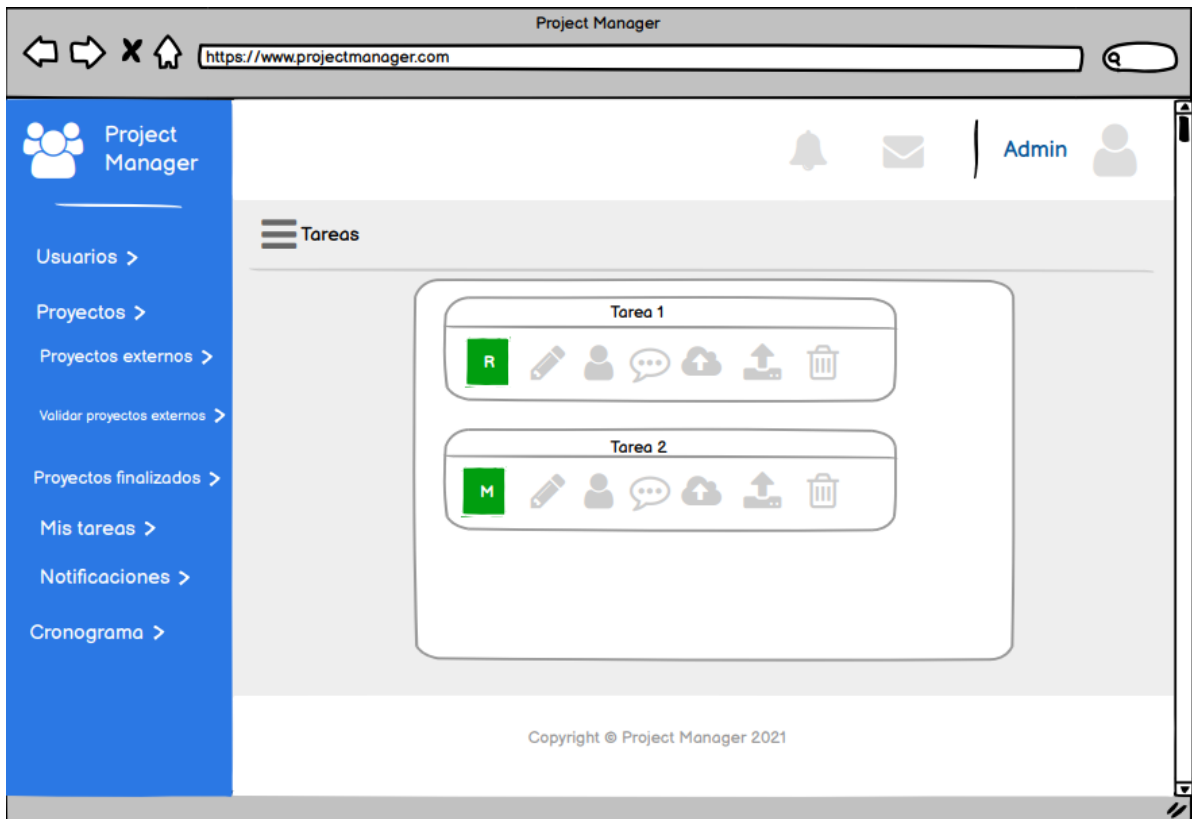


Figura 20 Anexo MIU: Maqueta de interfaz de usuario "Historias de usuario"

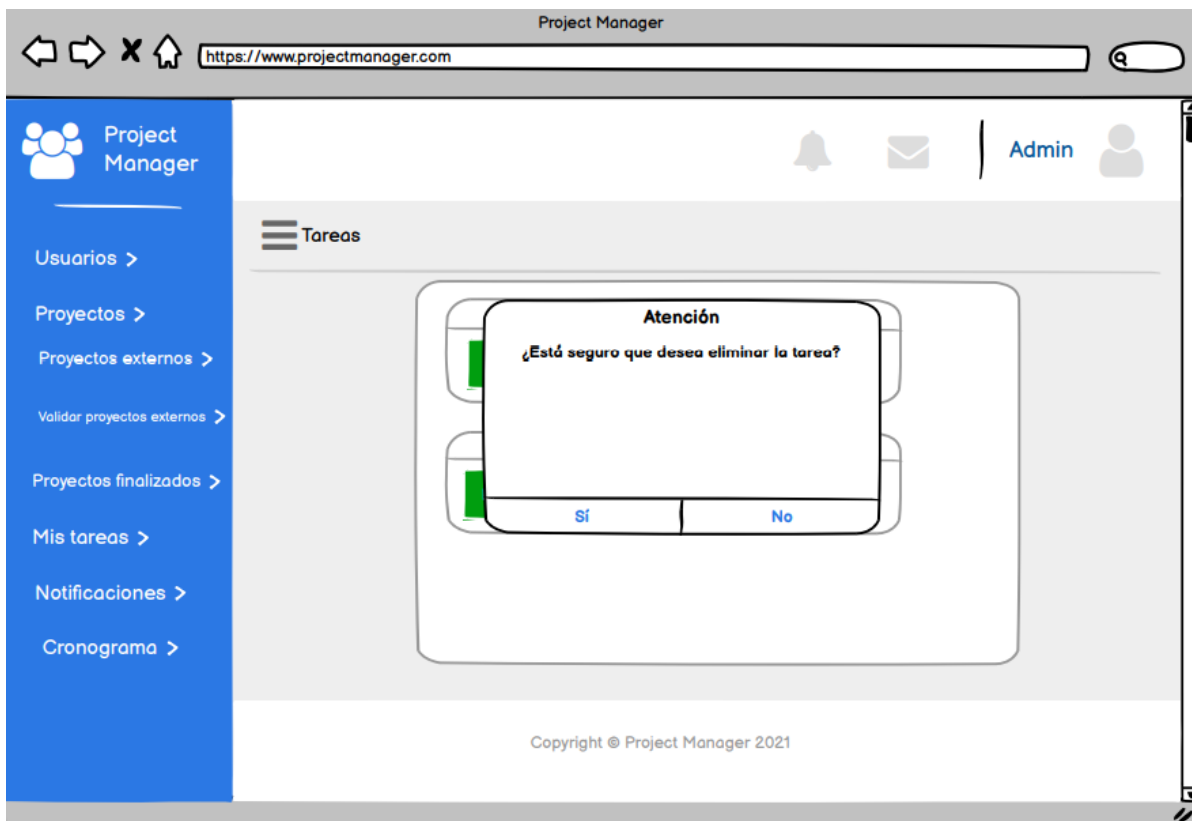


Figura 21 Anexo MIU: Maqueta de interfaz de usuario "Eliminar historia de usuario"

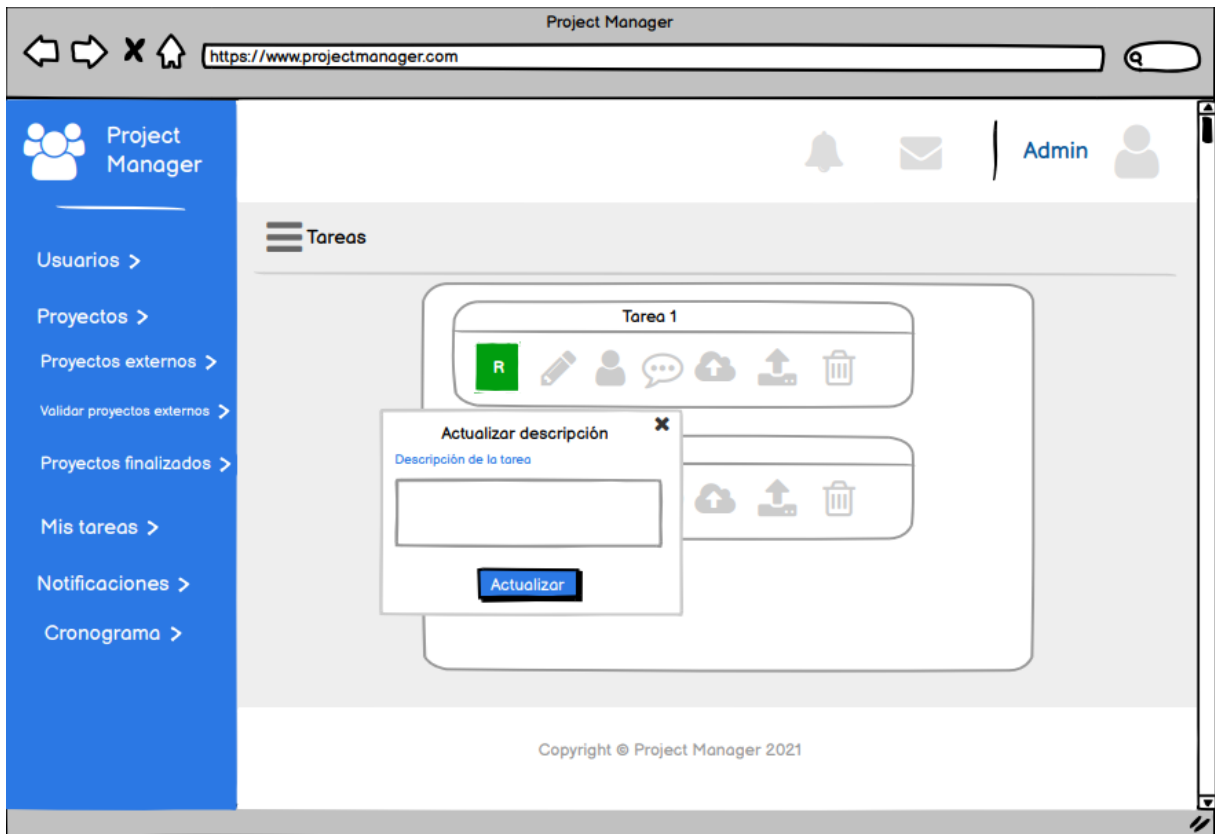


Figura 22 Anexo MIU: Maqueta de interfaz de usuario “Actualizar descripción de historia de usuario”

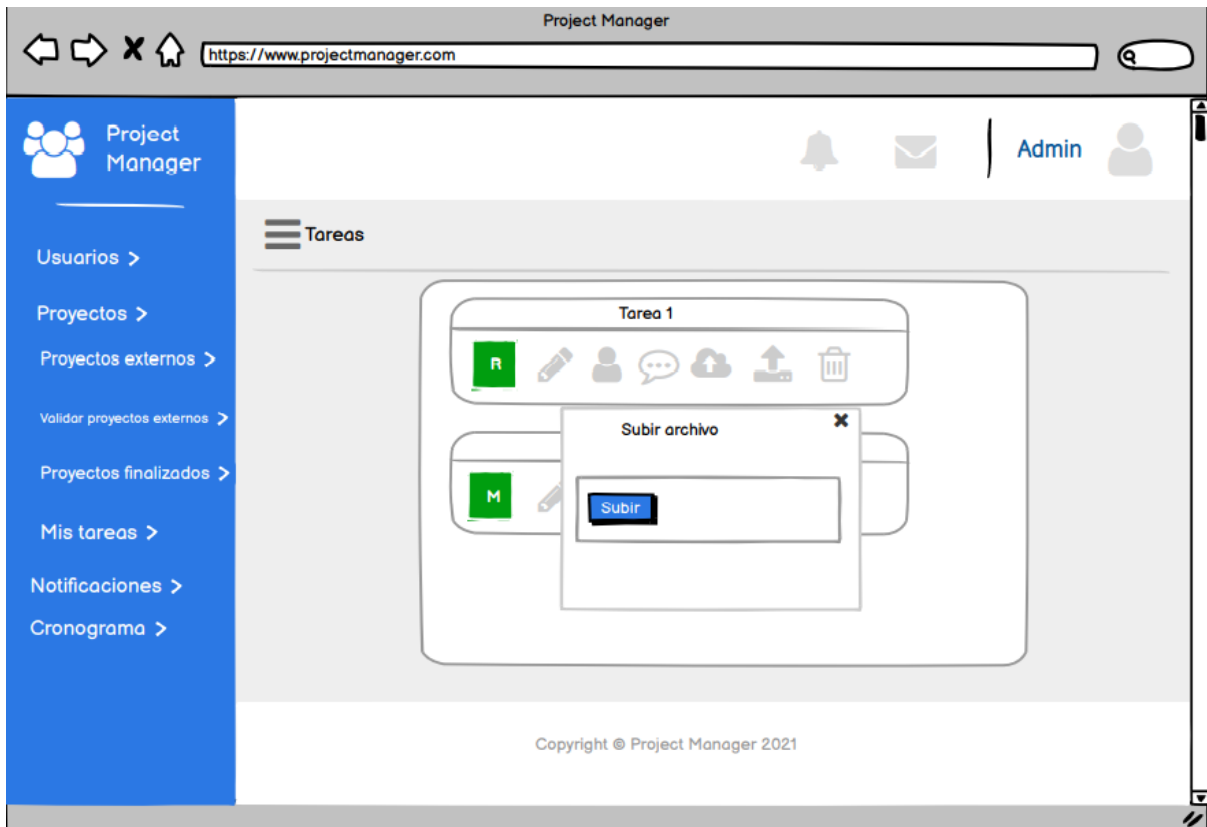


Figura 23 Anexo MIU: Maqueta de interfaz de usuario “Subir archivo a historia de usuario”

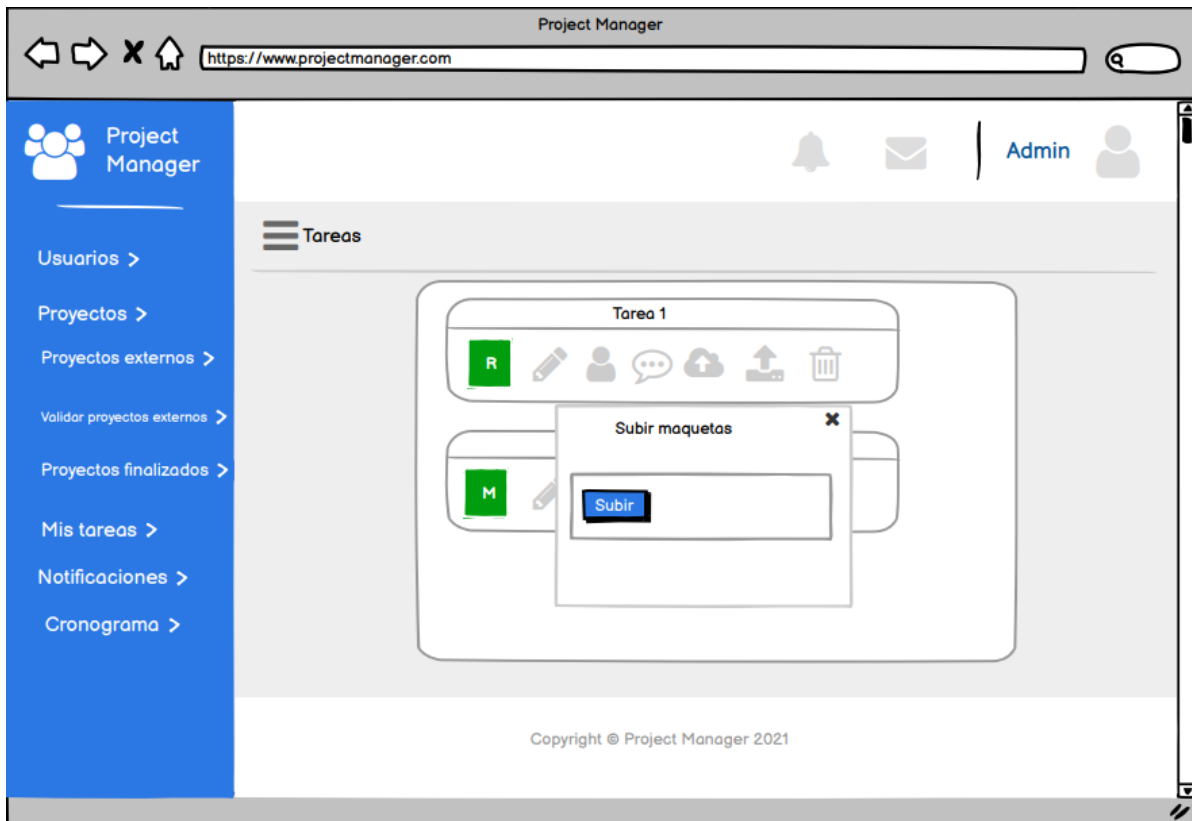


Figura 24 Anexo MIU: Maqueta de interfaz de usuario “Subir maqueta a historia de usuario”

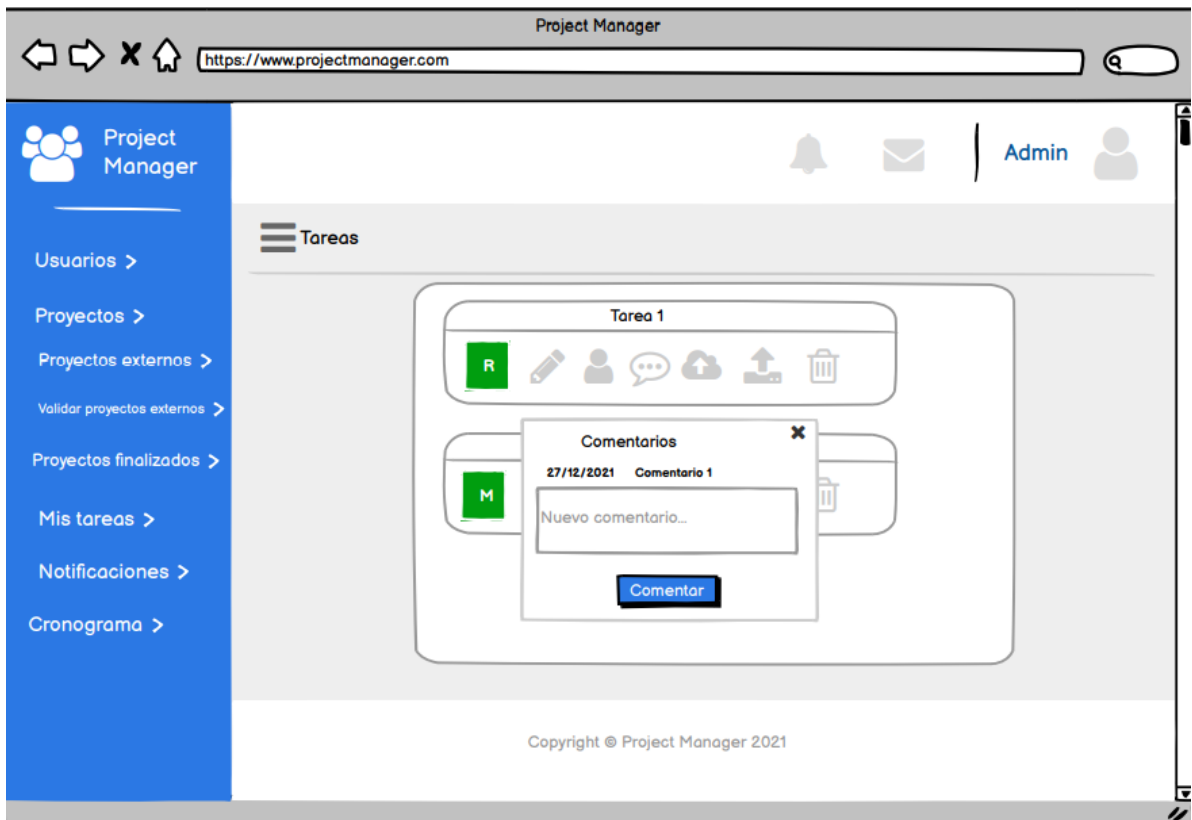


Figura 25 Anexo MIU: Maqueta de interfaz de usuario “Comentar historia de usuario”

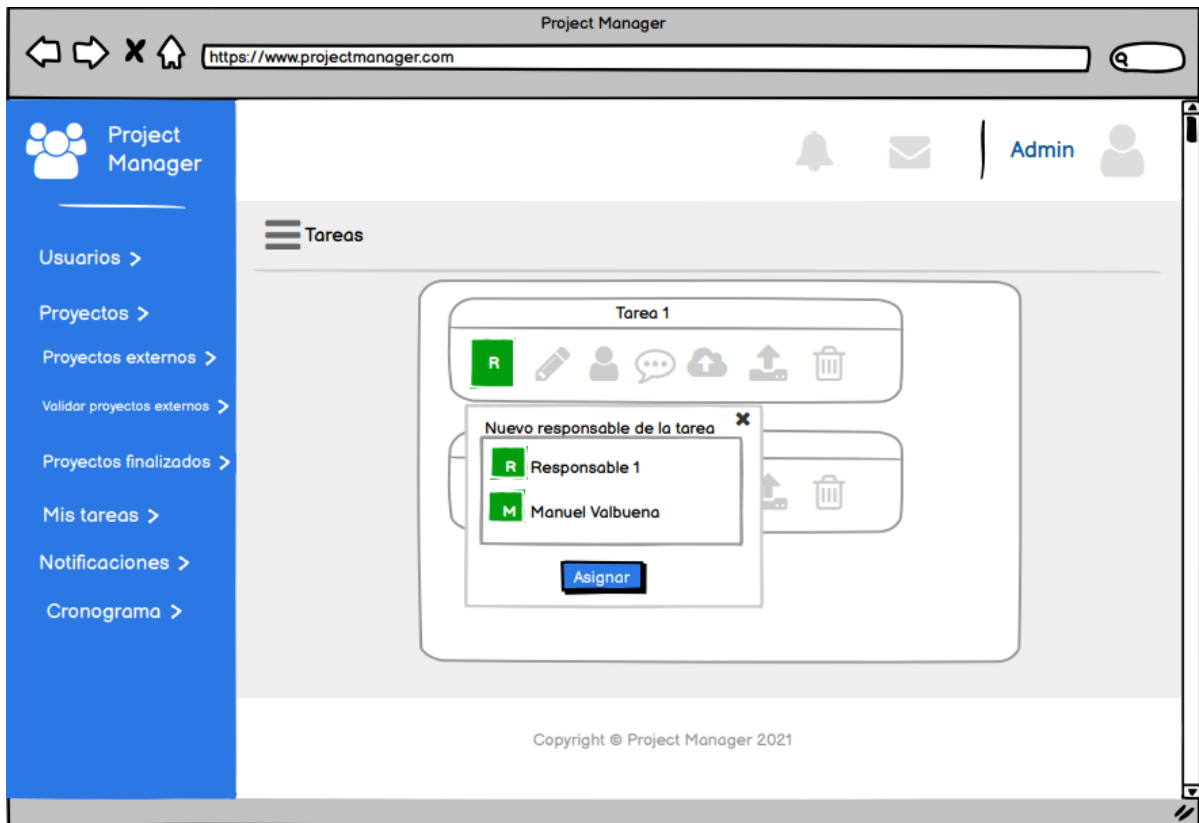


Figura 26 Anexo MIU: Maqueta de interfaz de usuario "Añadir responsable a historia de usuario"

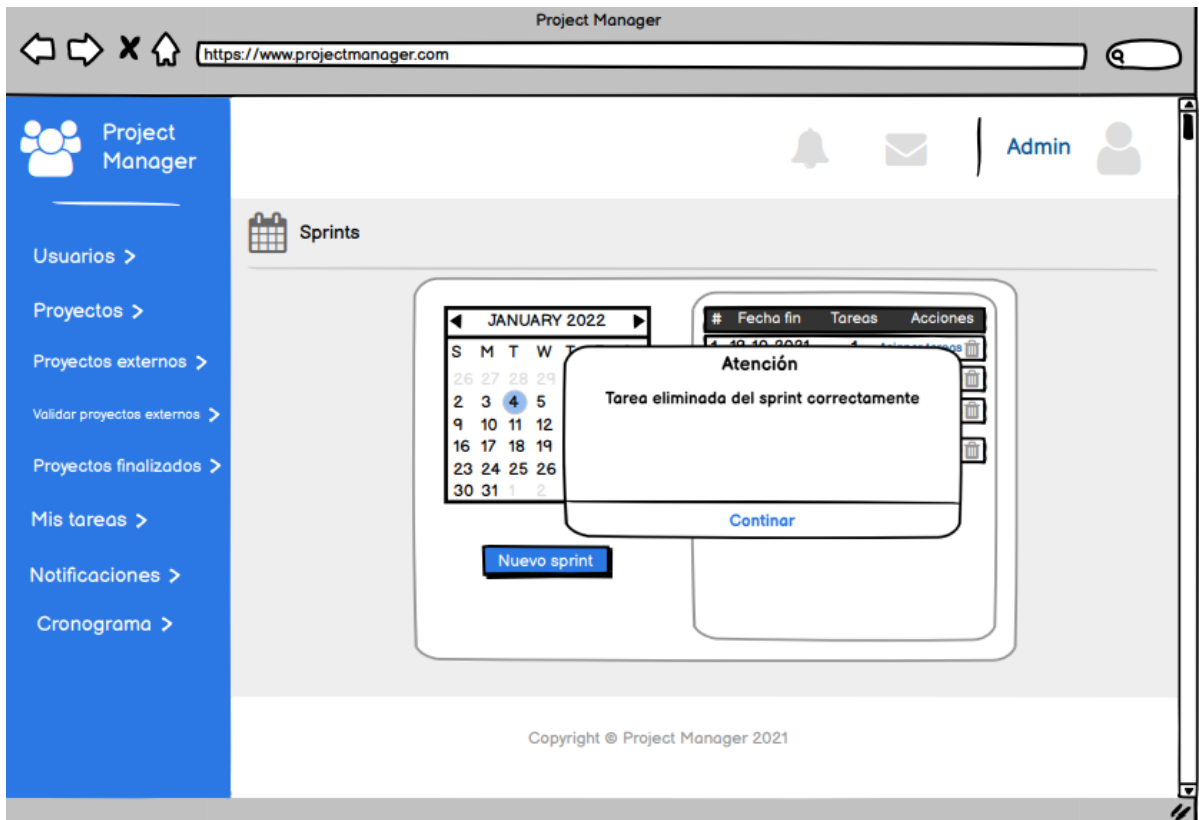


Figura 27 Anexo MIU: Maqueta de interfaz de usuario "Historia de usuario eliminada"

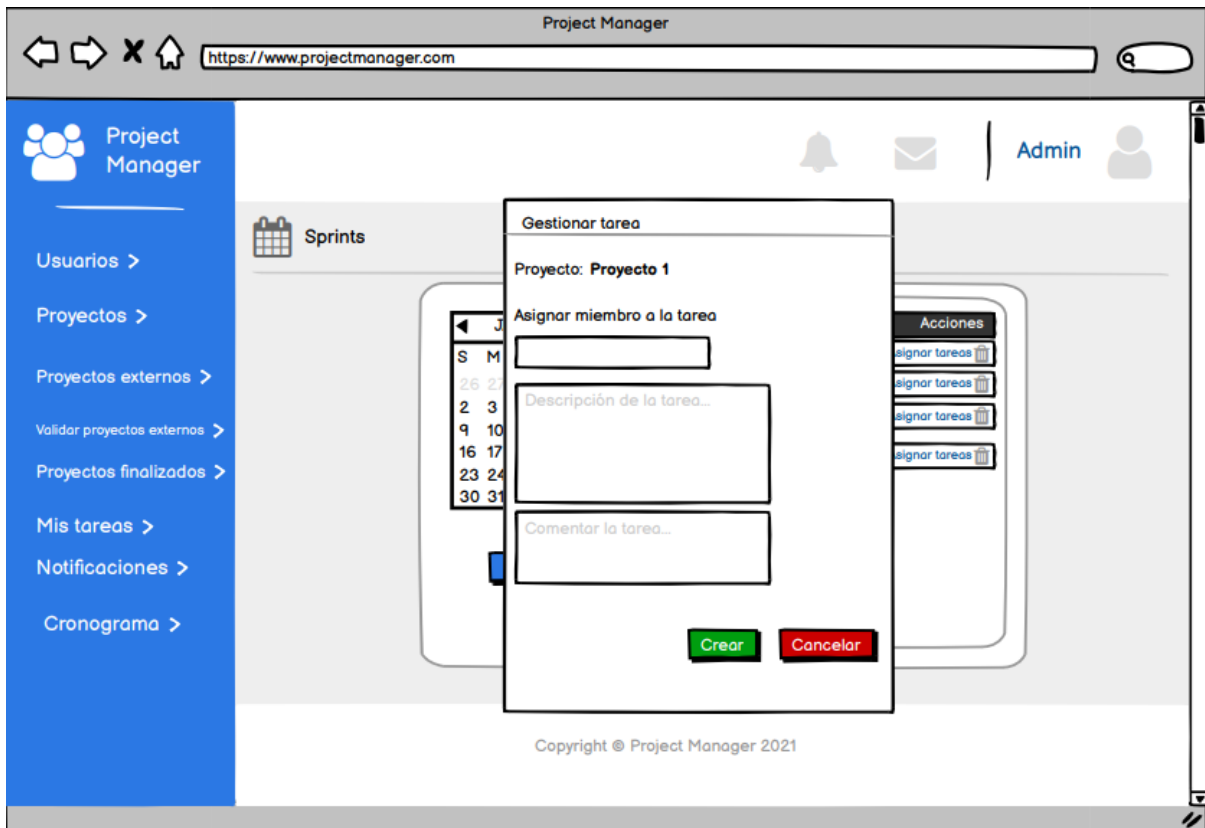


Figura 28 Anexo MIU: Maqueta de interfaz de usuario "Crear historia de usuario"

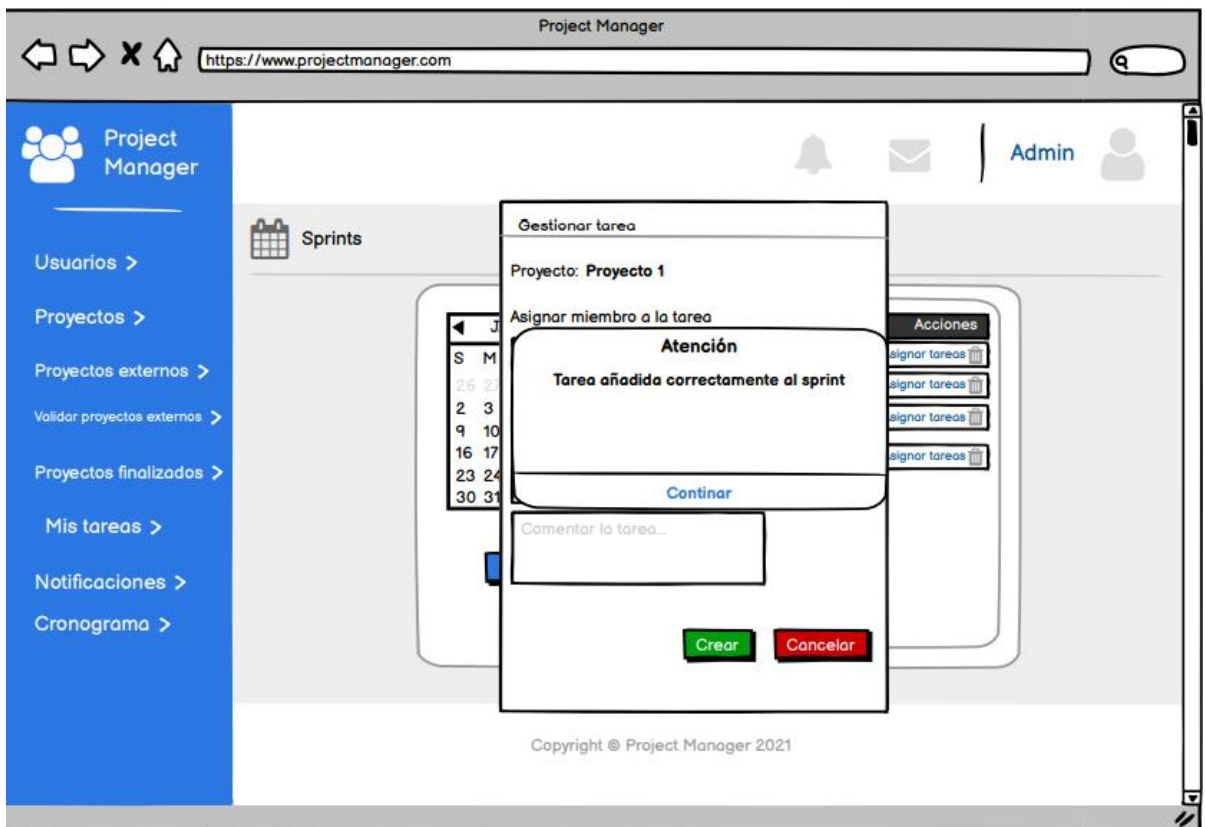


Figura 29 Anexo MIU: Maqueta de interfaz de usuario "Historia de usuario creada"

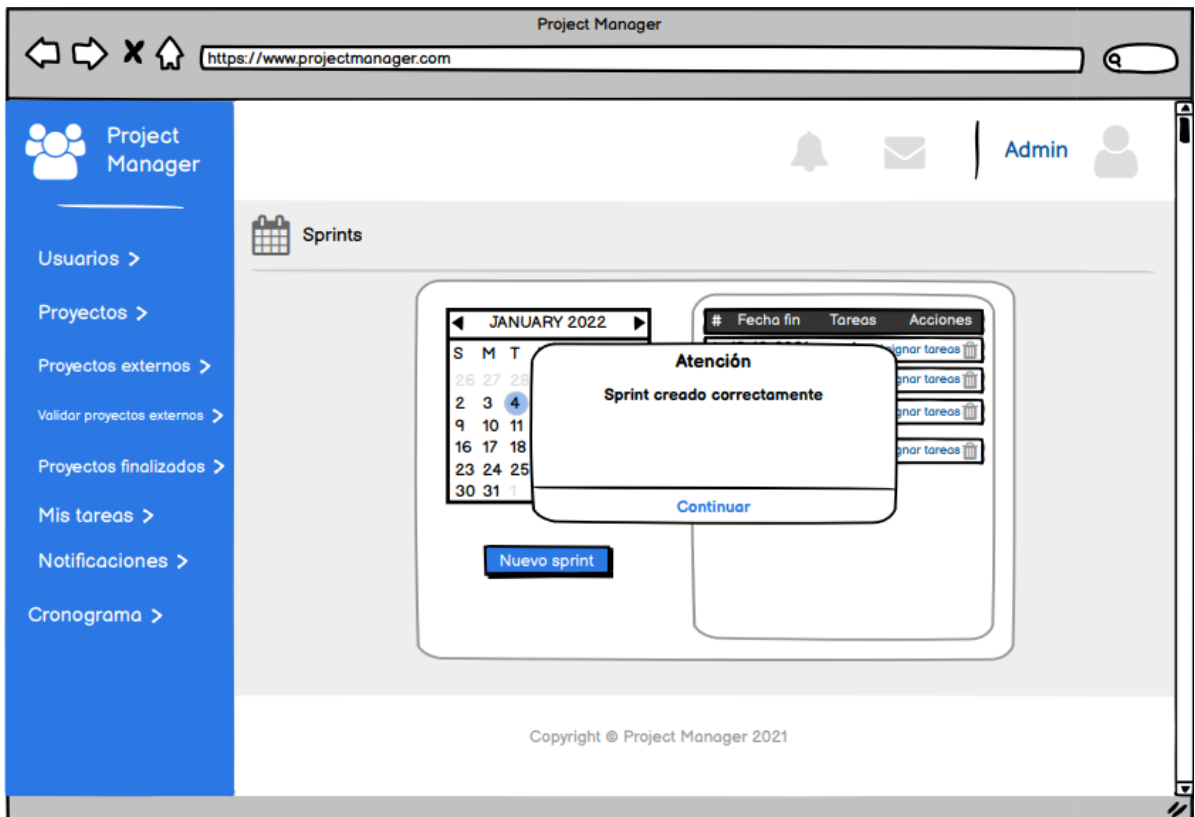


Figura 30 Anexo MIU: Maqueta de interfaz de usuario "Sprint creado"

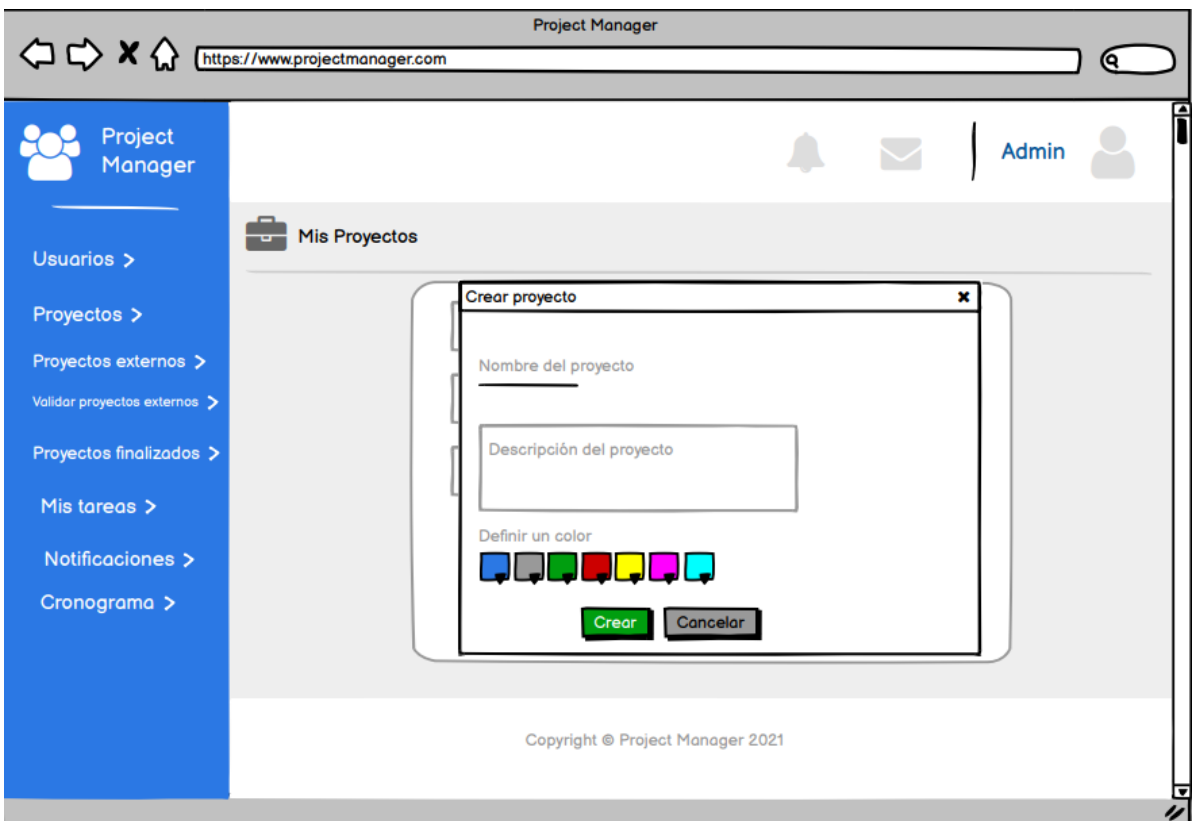


Figura 31 Anexo MIU: Maqueta de interfaz de usuario "Crear proyecto"

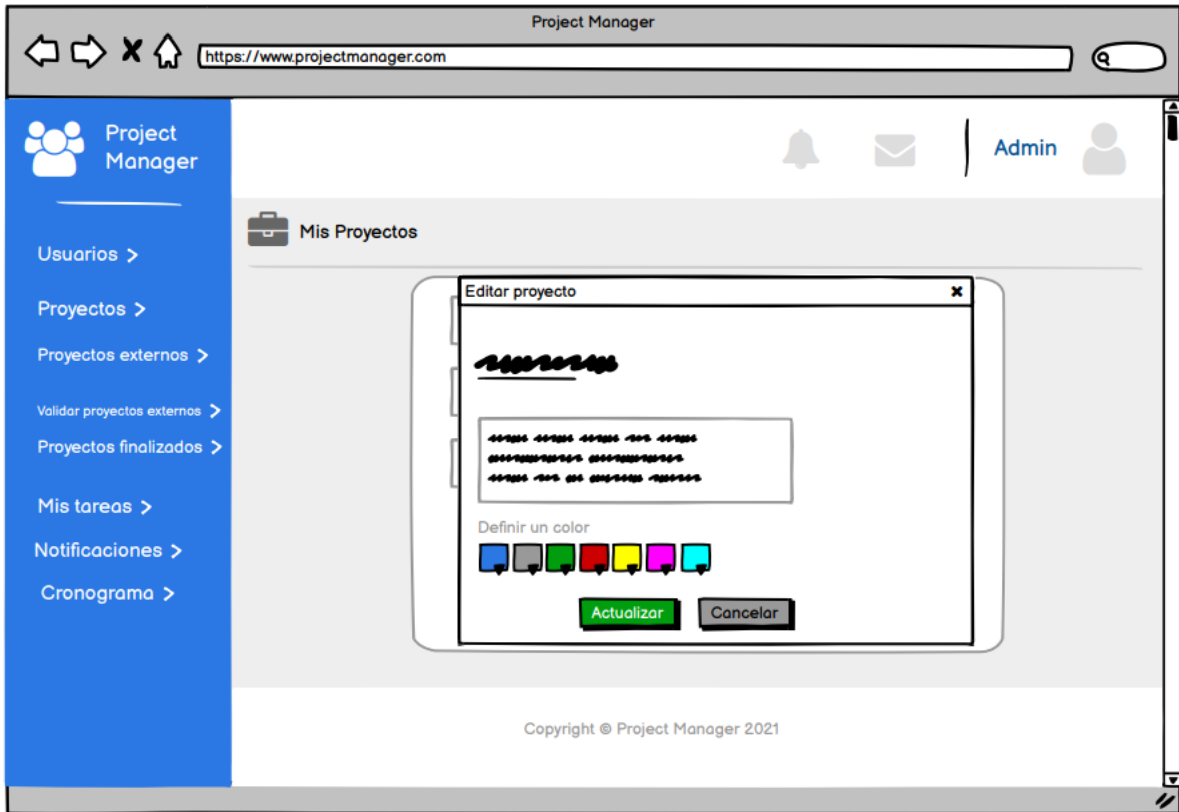


Figura 32 Anexo MIU: Maqueta de interfaz de usuario “Editar proyecto”

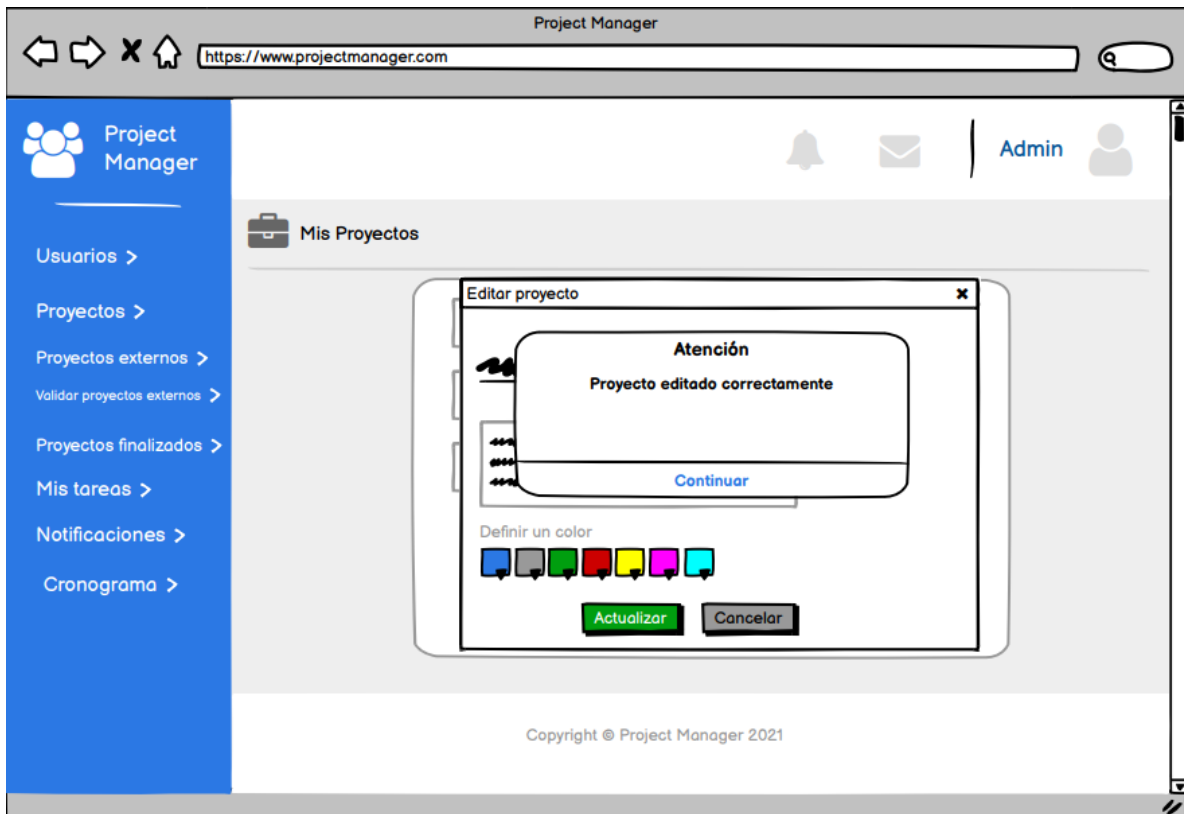


Figura 33 Anexo MIU: Maqueta de interfaz de usuario “Proyecto editado”

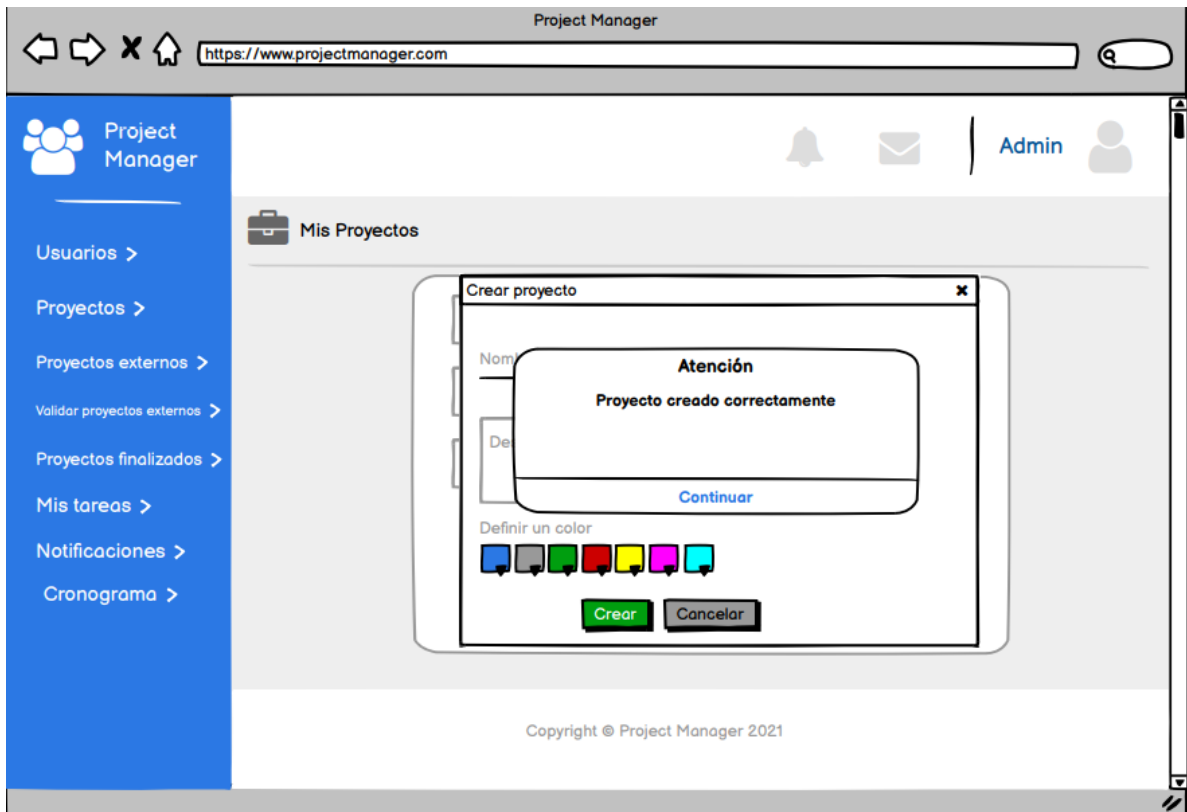


Figura 34 Anexo MIU: Maqueta de interfaz de usuario "Proyecto creado"

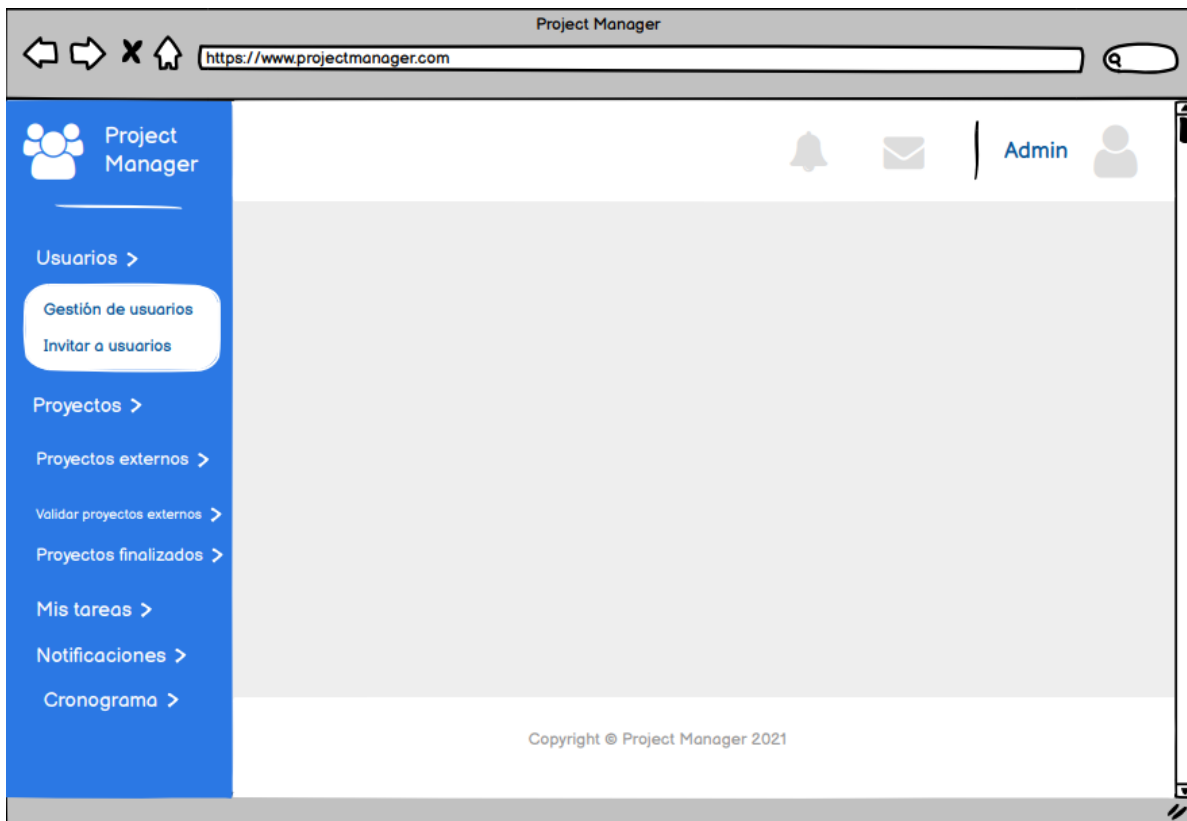


Figura 35 Anexo MIU: Maqueta de interfaz de usuario "Opciones de Usuarios"

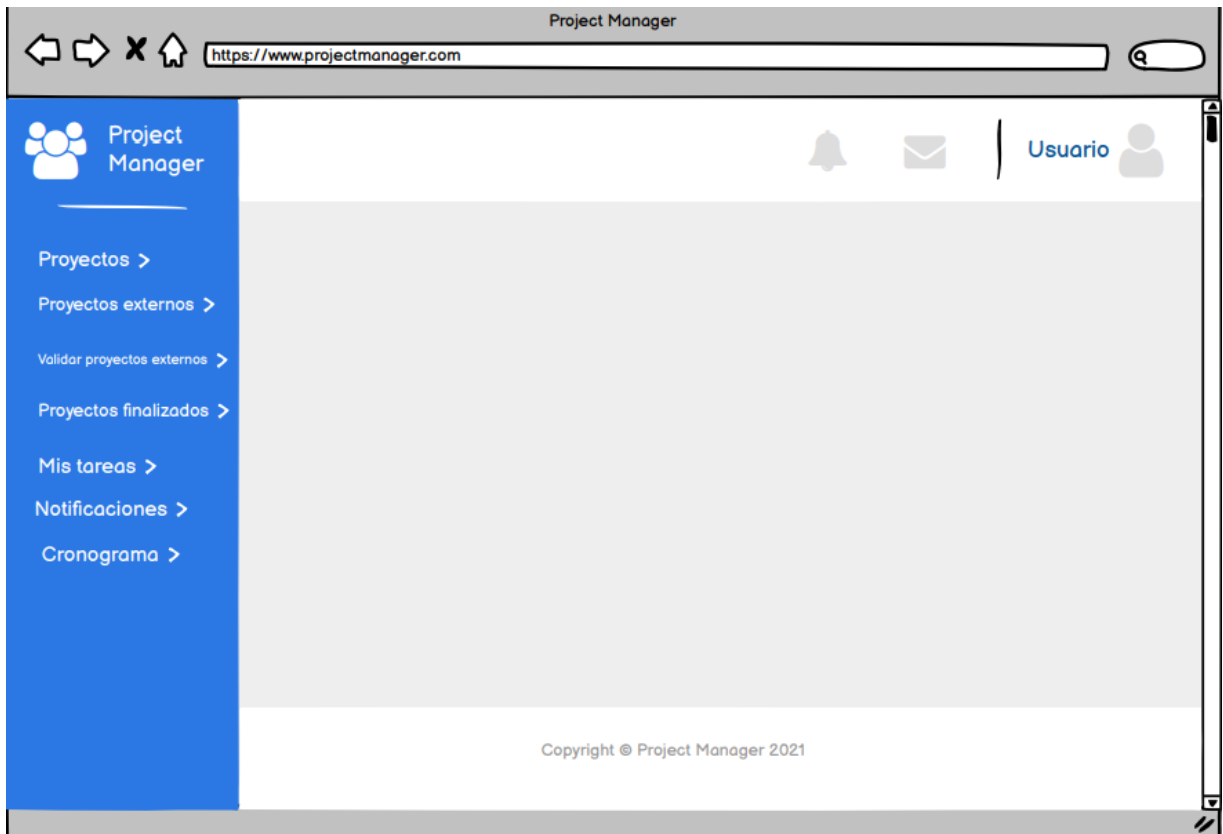


Figura 36 Anexo MIU: Maqueta de interfaz de usuario “Vista principal de los perfiles Equipo de desarrollo”

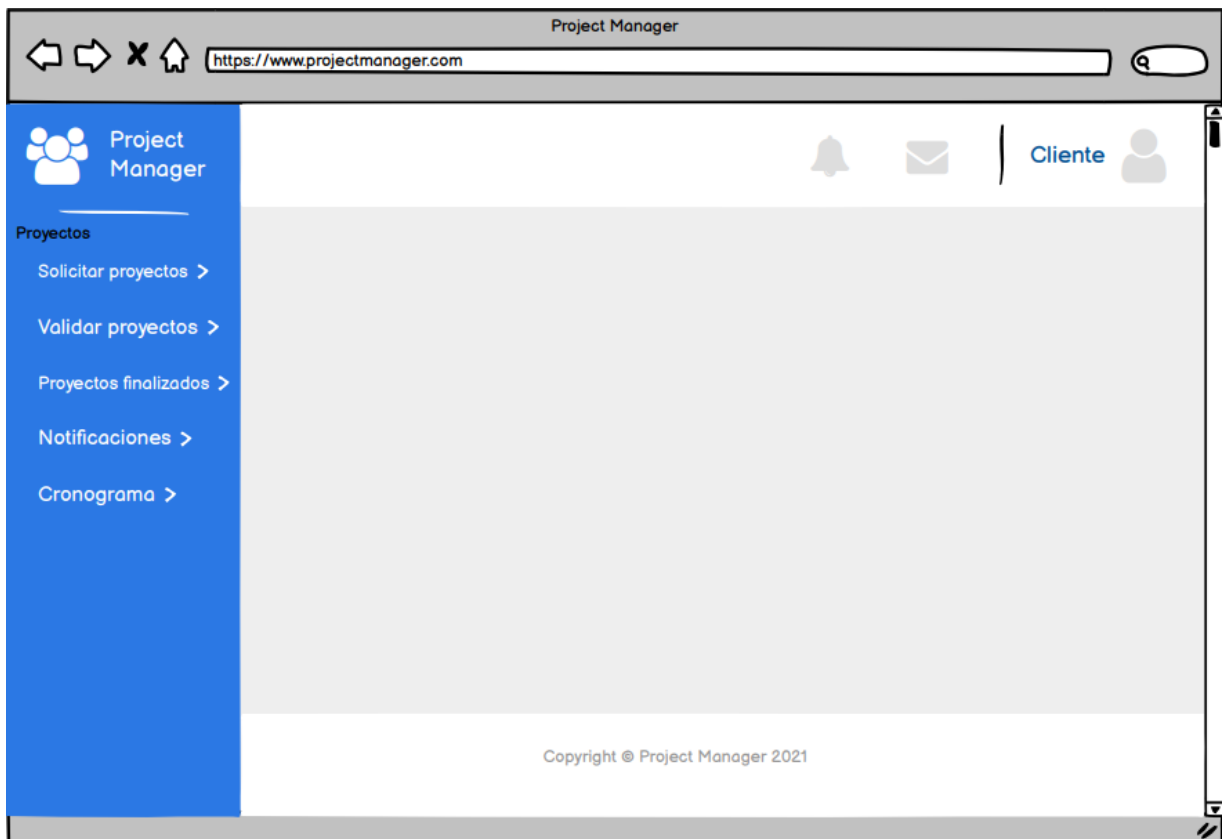


Figura 37 Anexo MIU: Maqueta de interfaz de usuario “Vista principal de los perfiles Equipo del cliente”

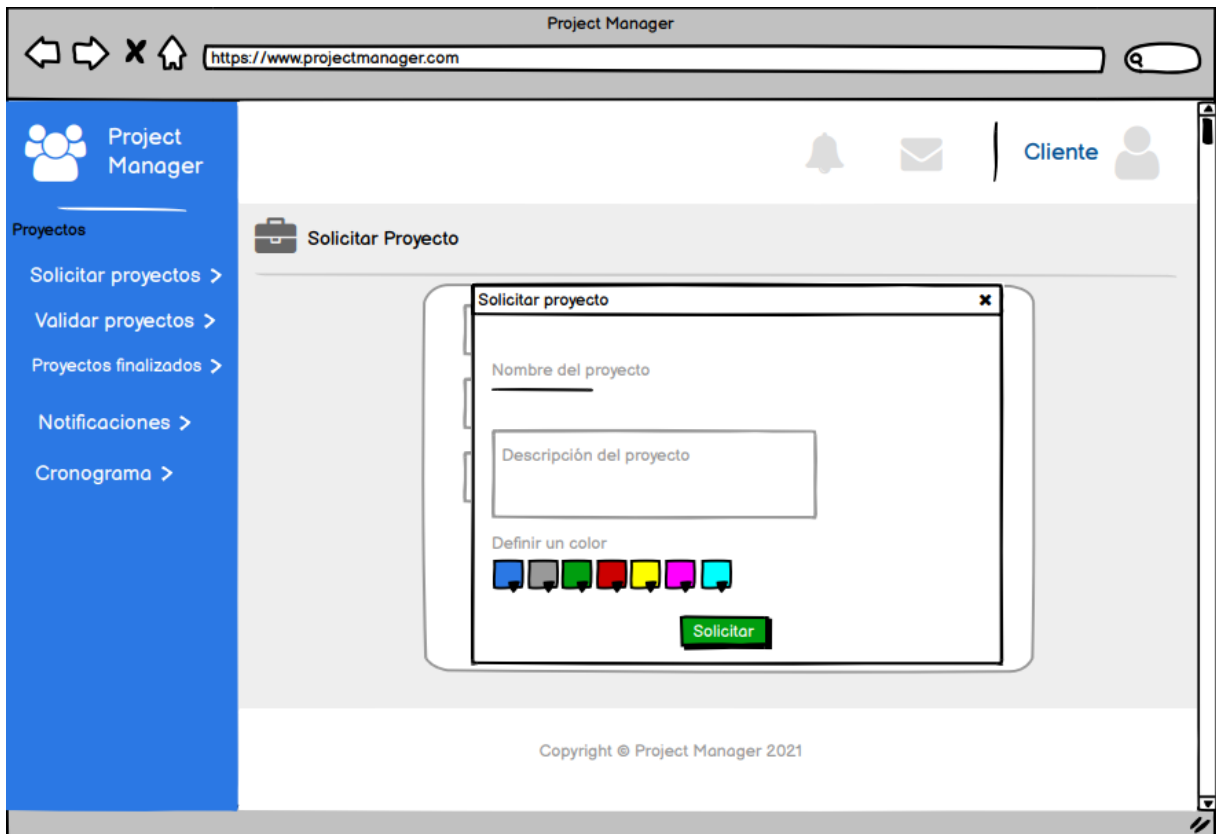


Figura 38 Anexo MIU: Maqueta de interfaz de usuario "Solicitar proyecto"

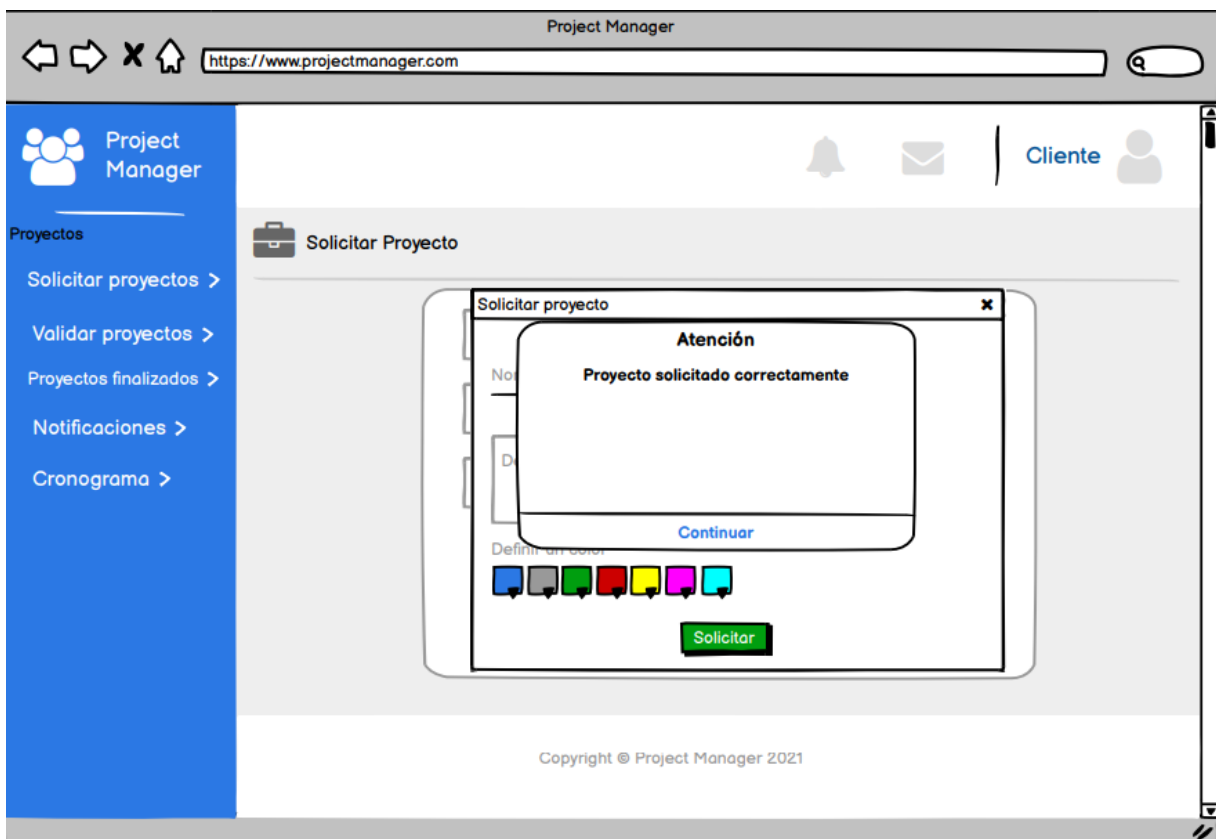


Figura 39 Anexo MIU: Maqueta de interfaz de usuario "Proyecto solicitado"

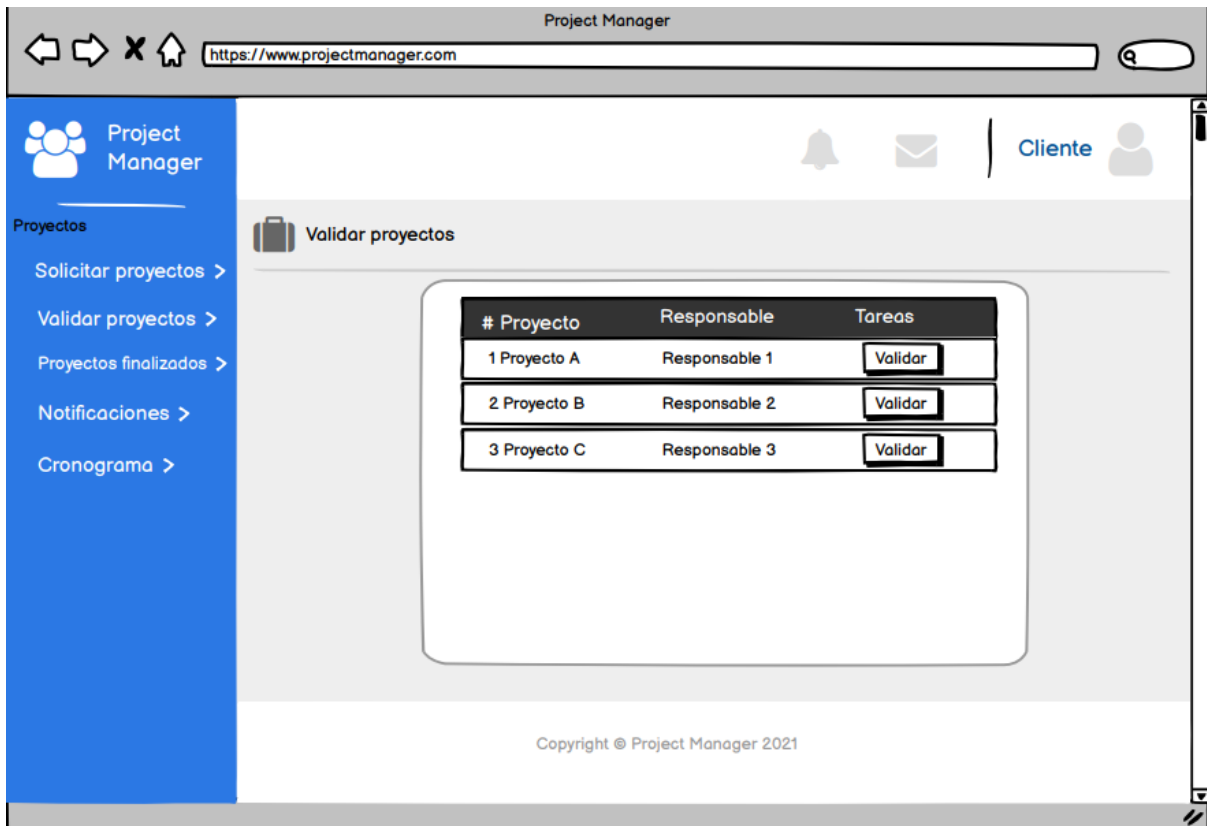


Figura 40 Anexo MIU: Maqueta de interfaz de usuario “Validar proyectos responsable del cliente”

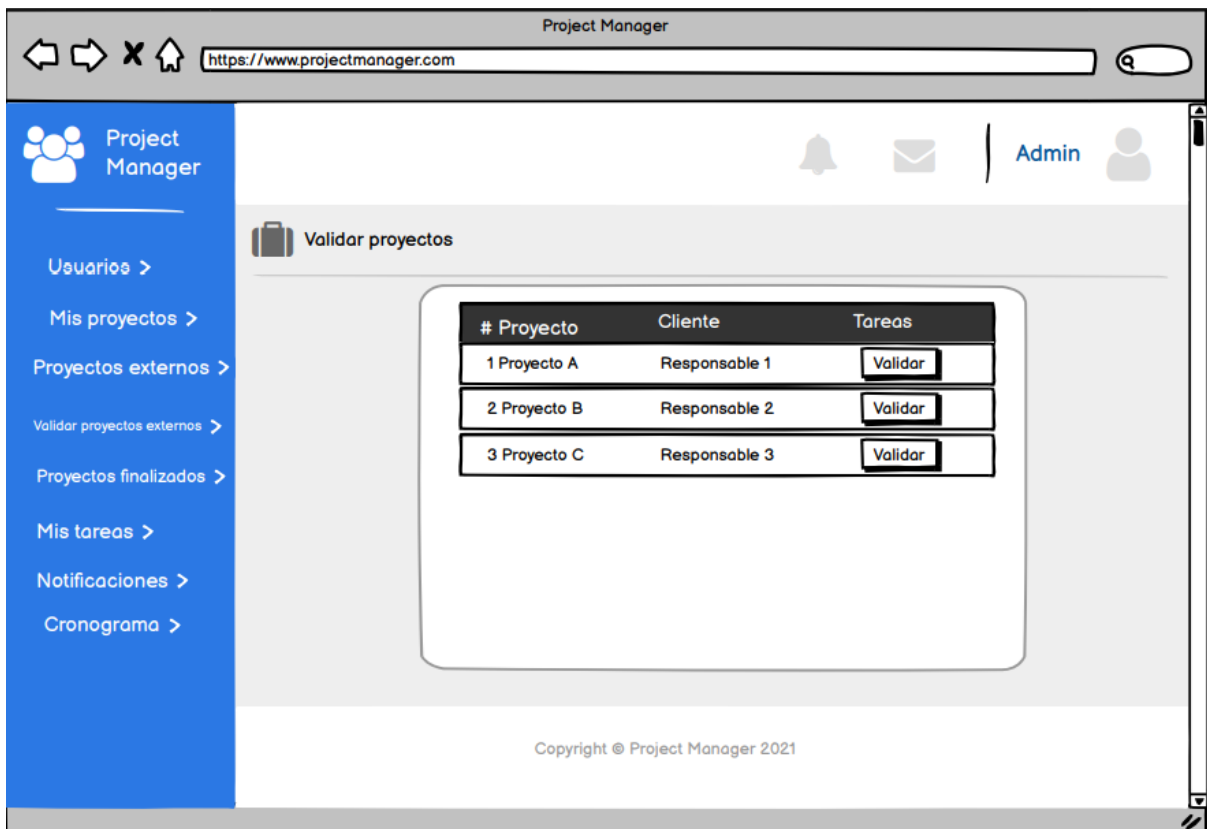


Figura 41 Anexo MIU: Maqueta de interfaz de usuario “Validar proyectos Jefe de proyecto”

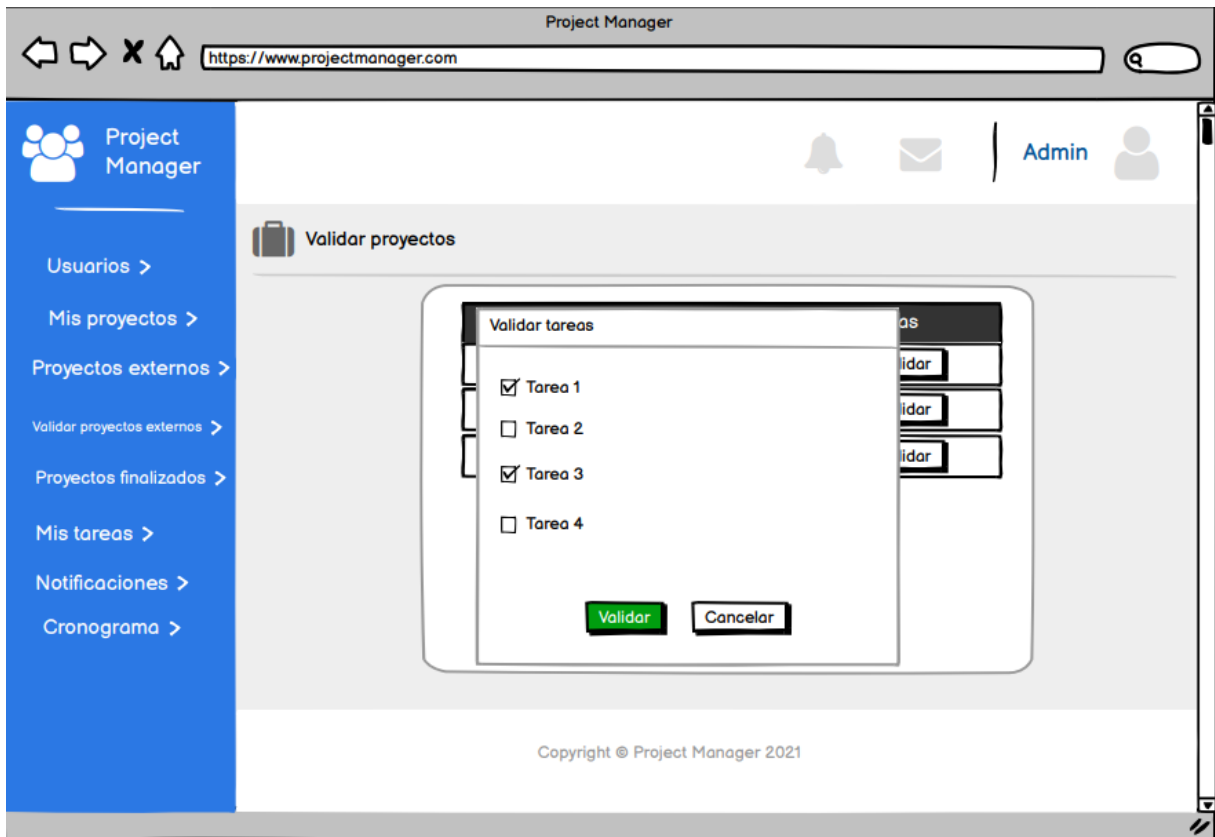


Figura 42 Anexo MIU: Maqueta de interfaz de usuario “Validar historias de usuario Jefe de proyecto”

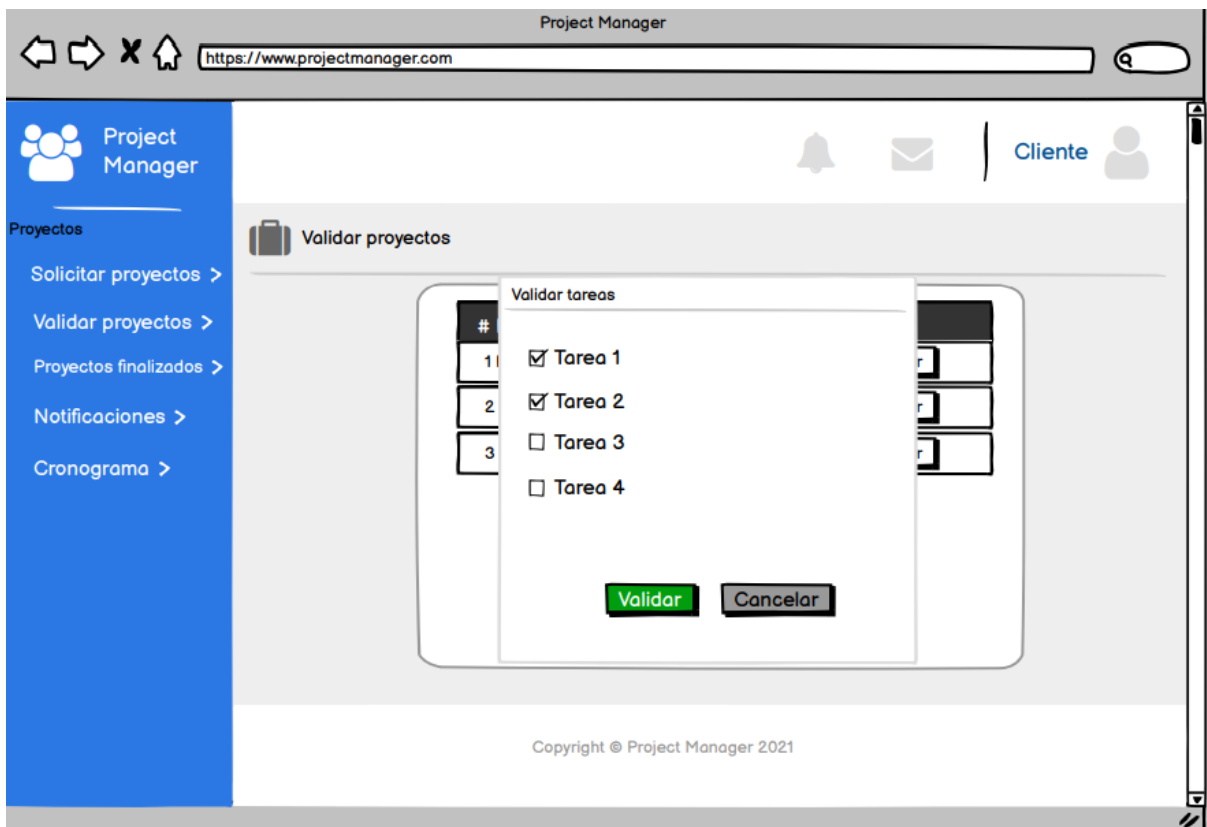


Figura 43 Anexo MIU: Maqueta de interfaz de usuario “Validar historias de usuario Responsable del cliente”

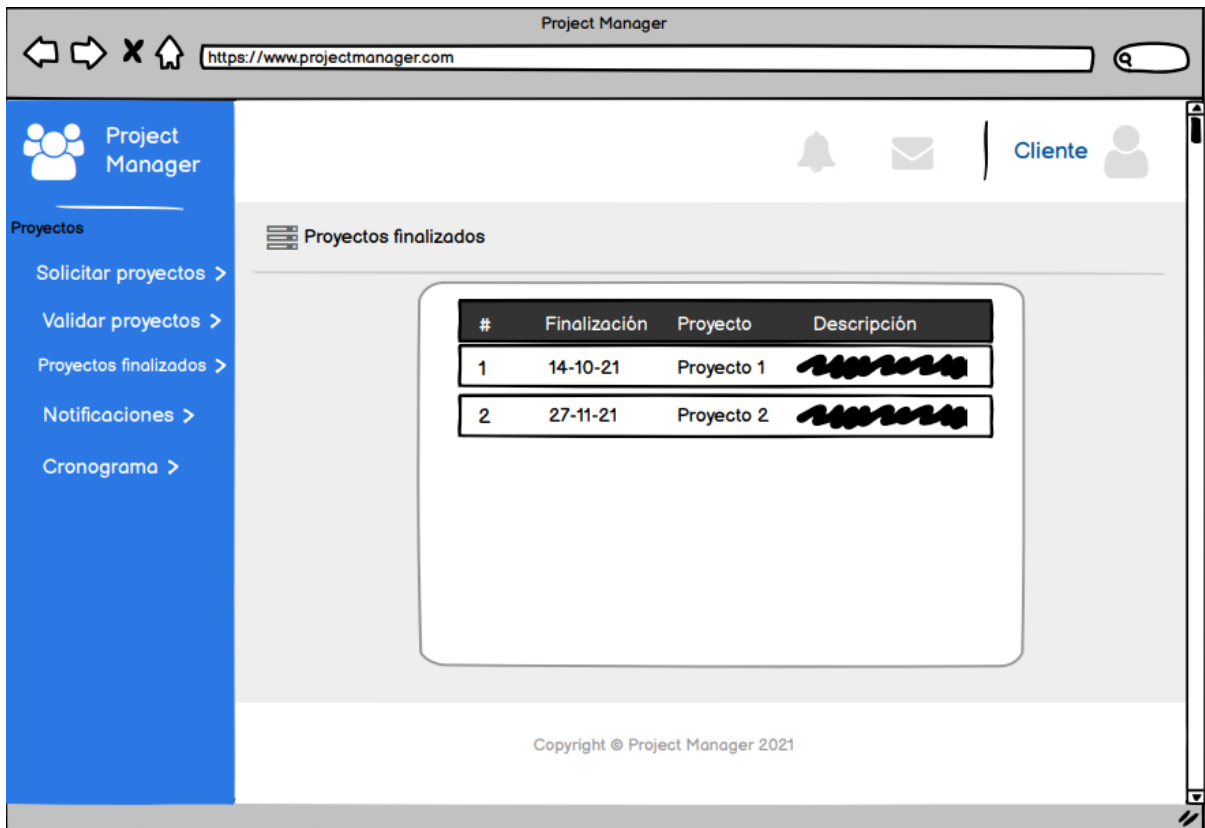


Figura 44 Anexo MIU: Maqueta de interfaz de usuario "Proyectos finalizados Responsable del cliente"

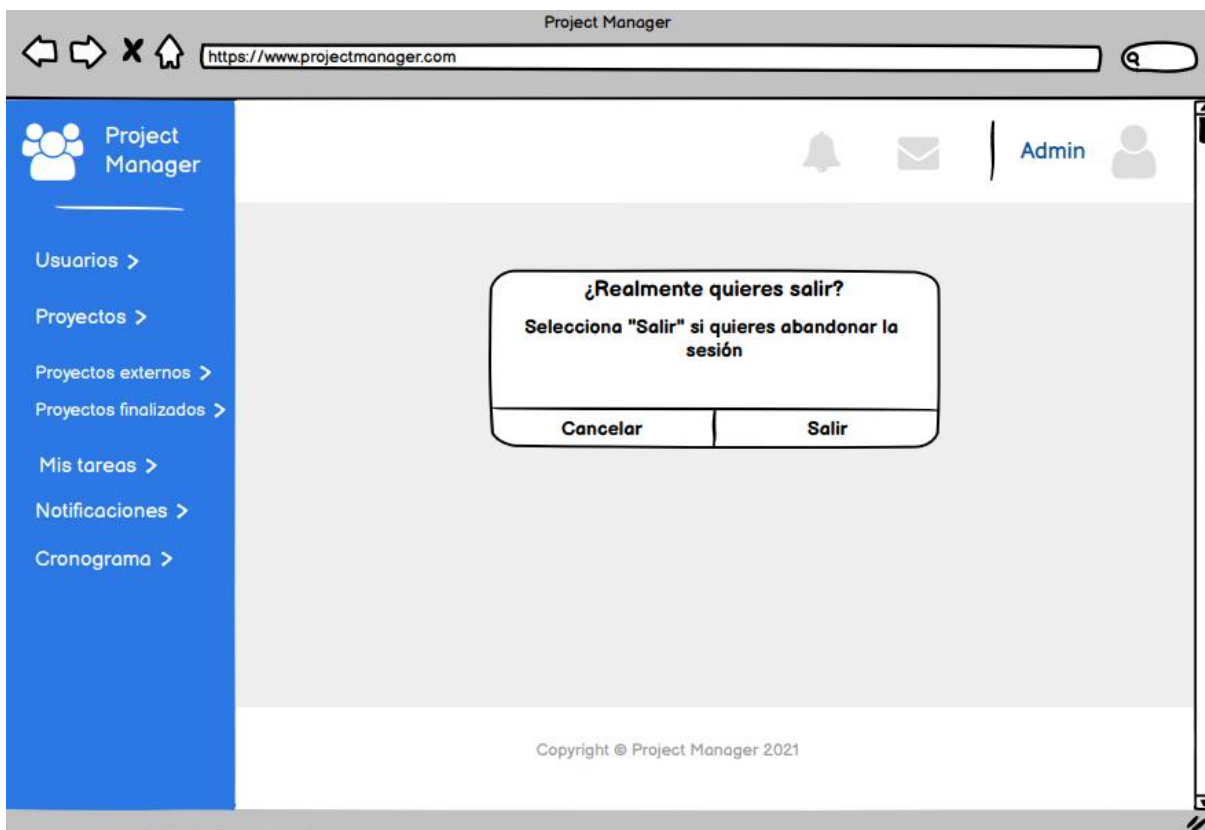


Figura 45 Anexo MIU: Maqueta de interfaz de usuario "Cerrar sesión"

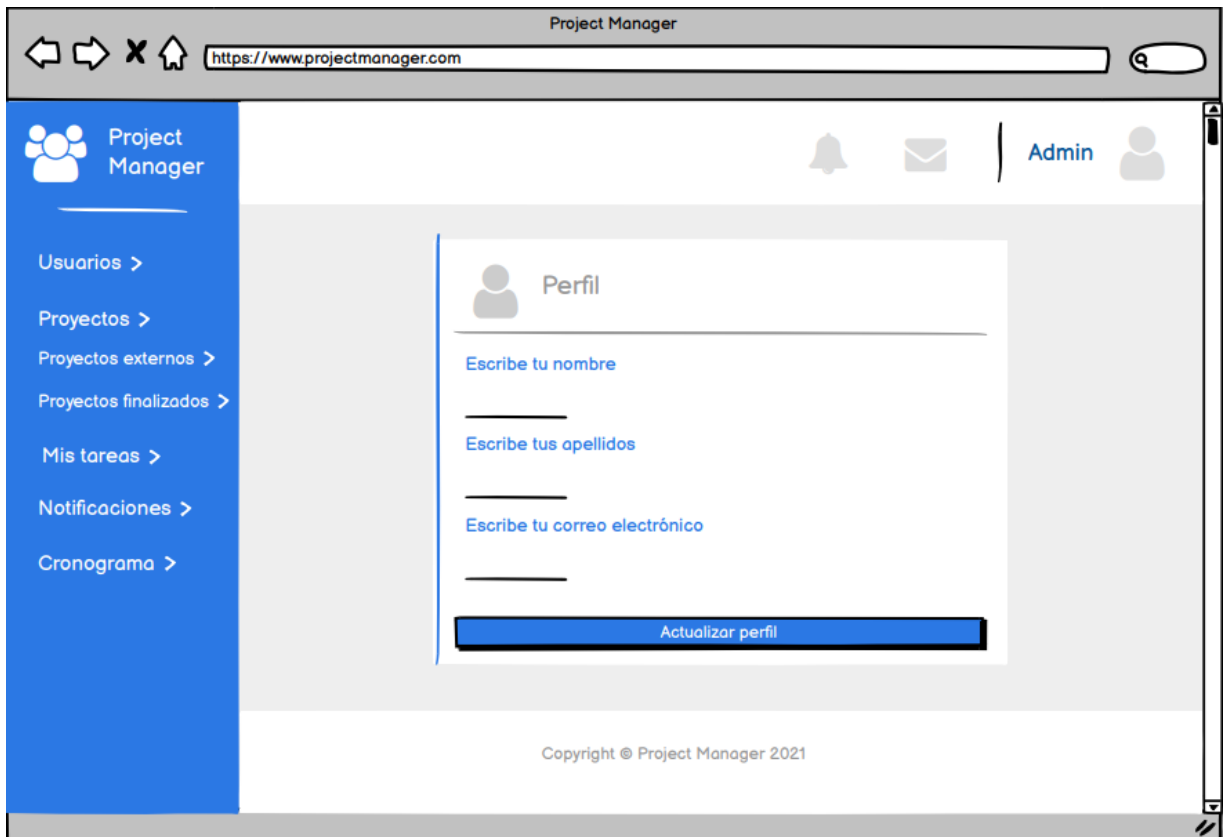


Figura 46 Anexo MIU: Maqueta de interfaz de usuario "Editar perfil"

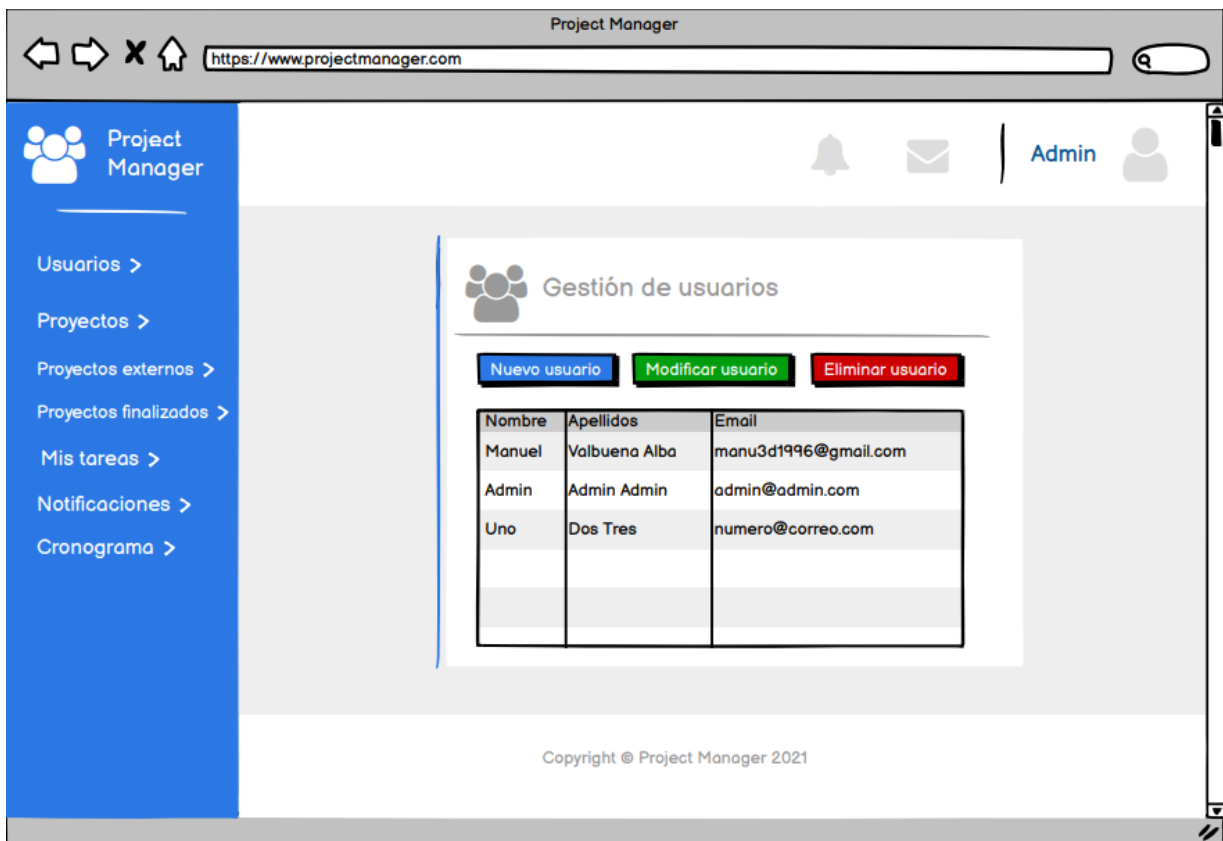


Figura 47 Anexo MIU: Maqueta de interfaz de usuario "Gestión de usuarios"

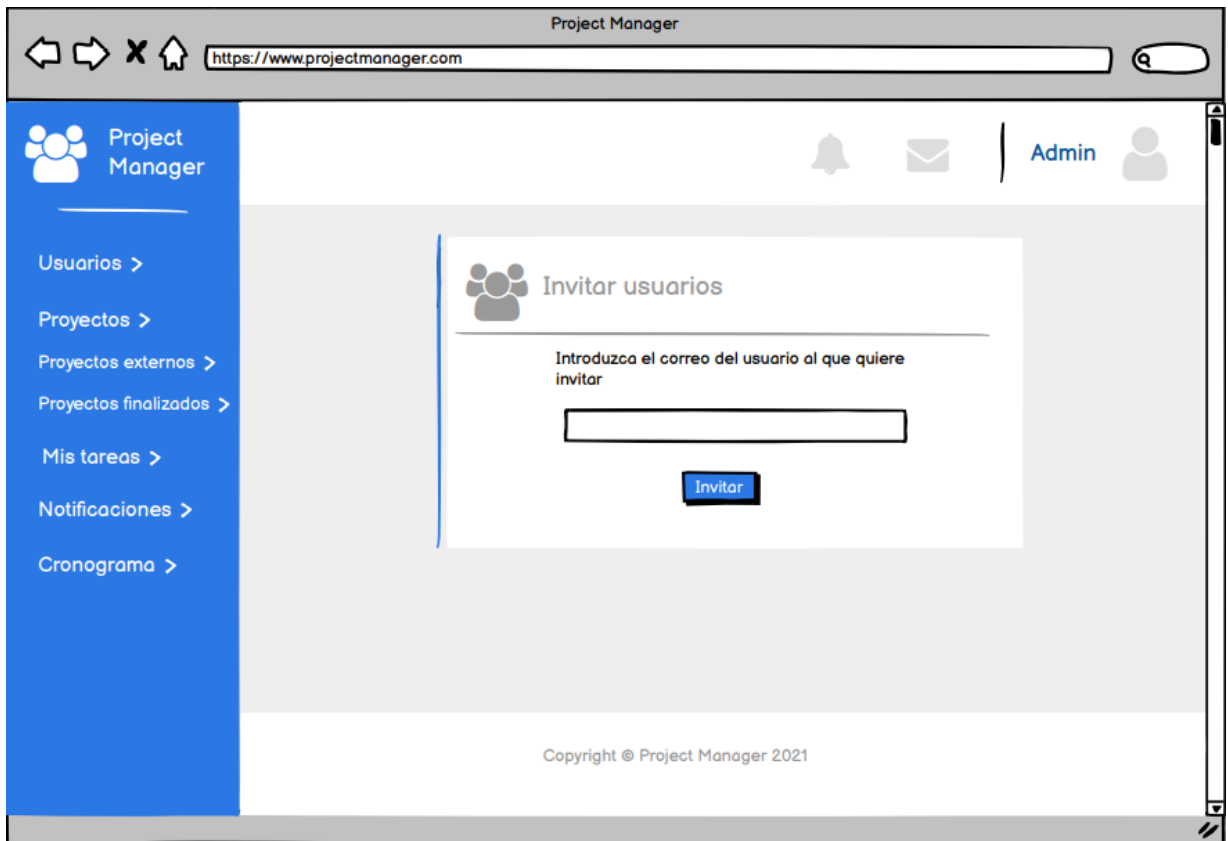


Figura 48 Anexo MIU: Maqueta de interfaz de usuario "Invitar usuarios"

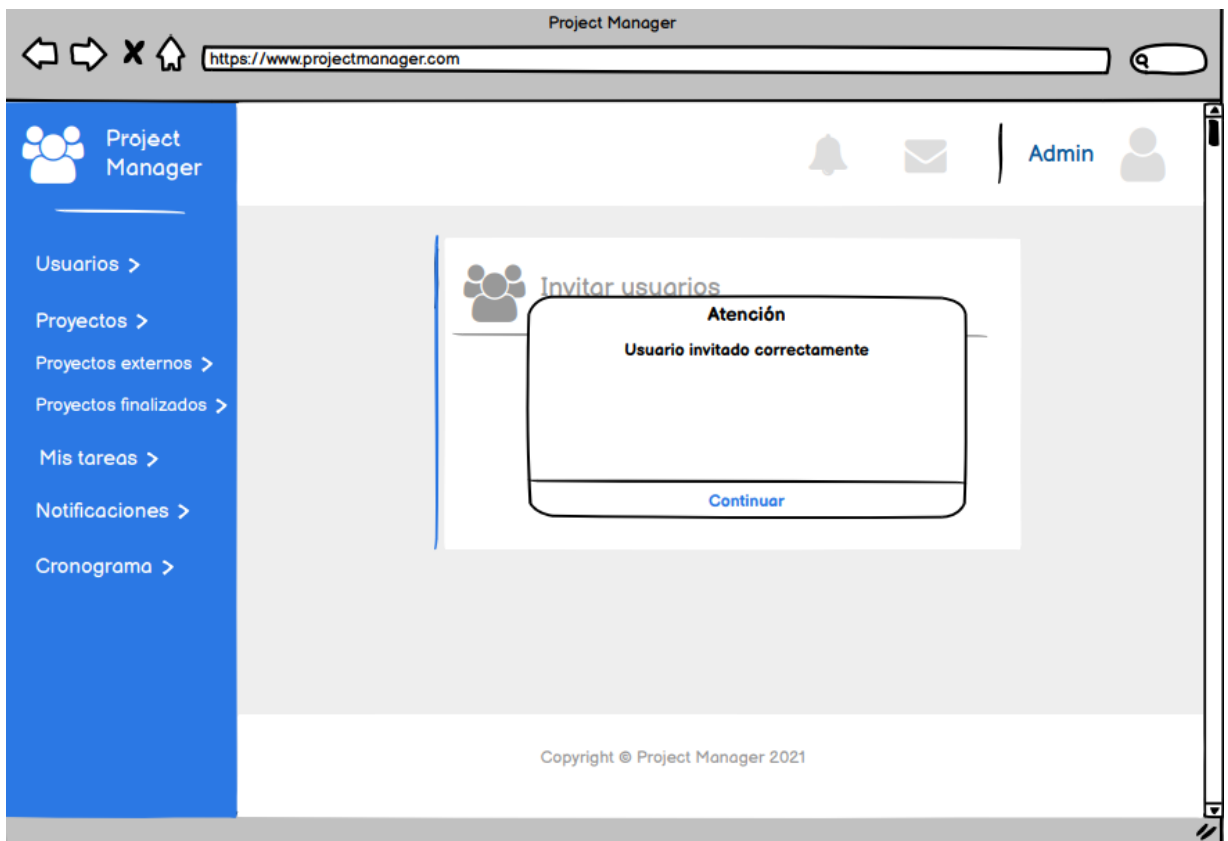


Figura 49 Anexo MIU: Maqueta de interfaz de usuario "Usuario invitado"

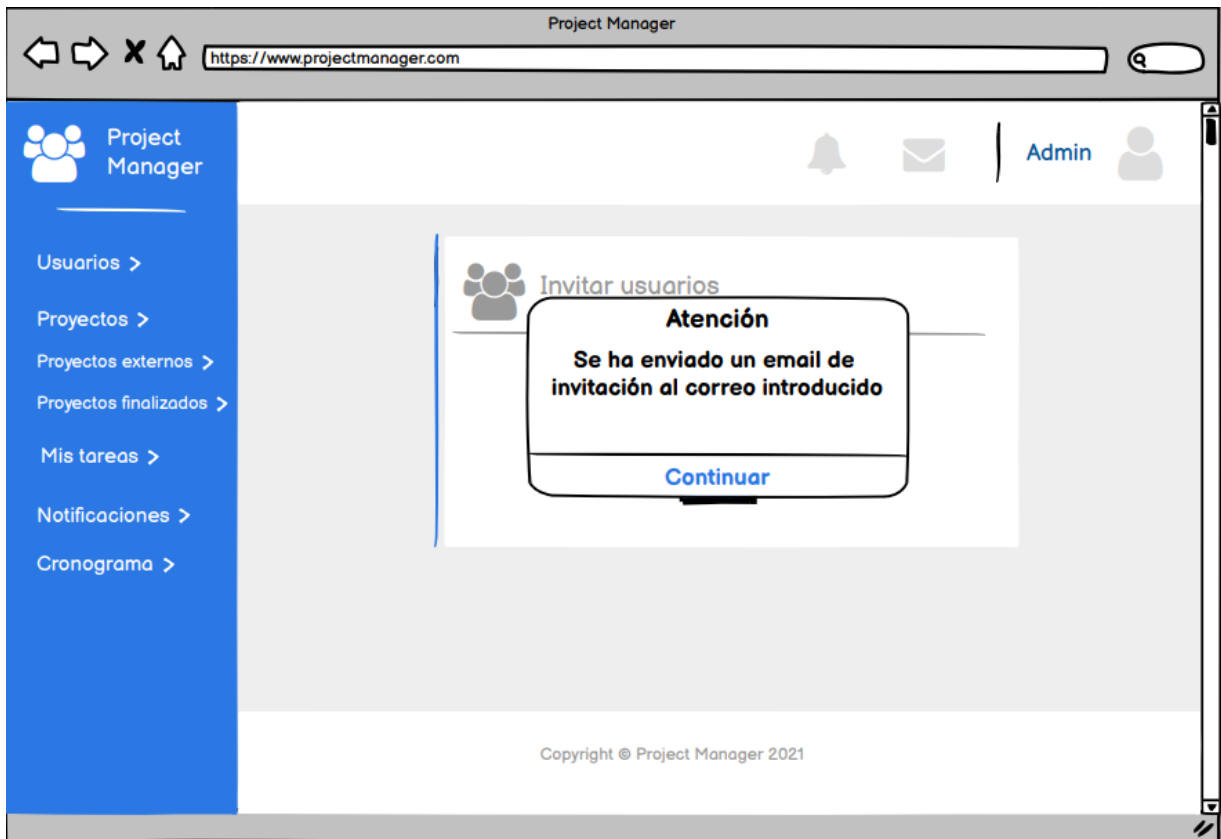


Figura 50 Anexo MIU: Maqueta de interfaz de usuario "Correo de invitación enviado"

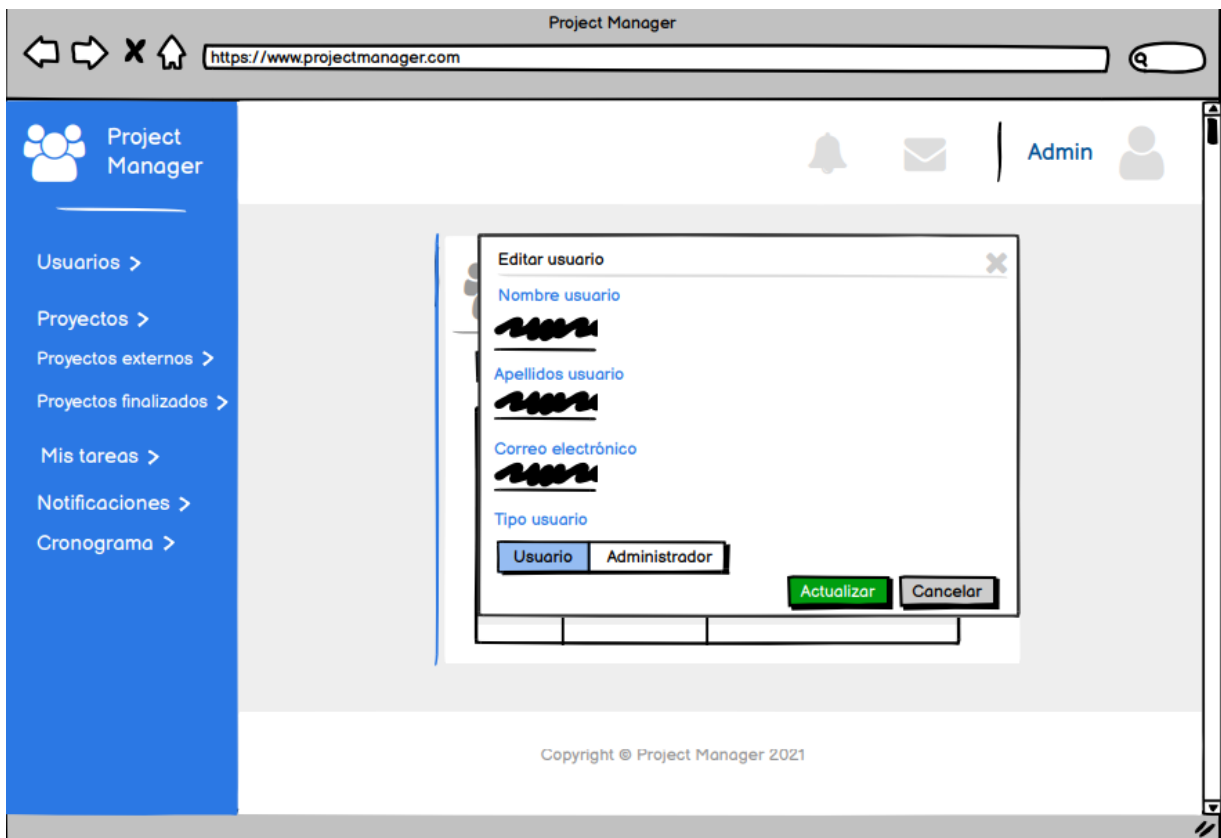


Figura 51 Anexo MIU: Maqueta de interfaz de usuario "Editar usuario"

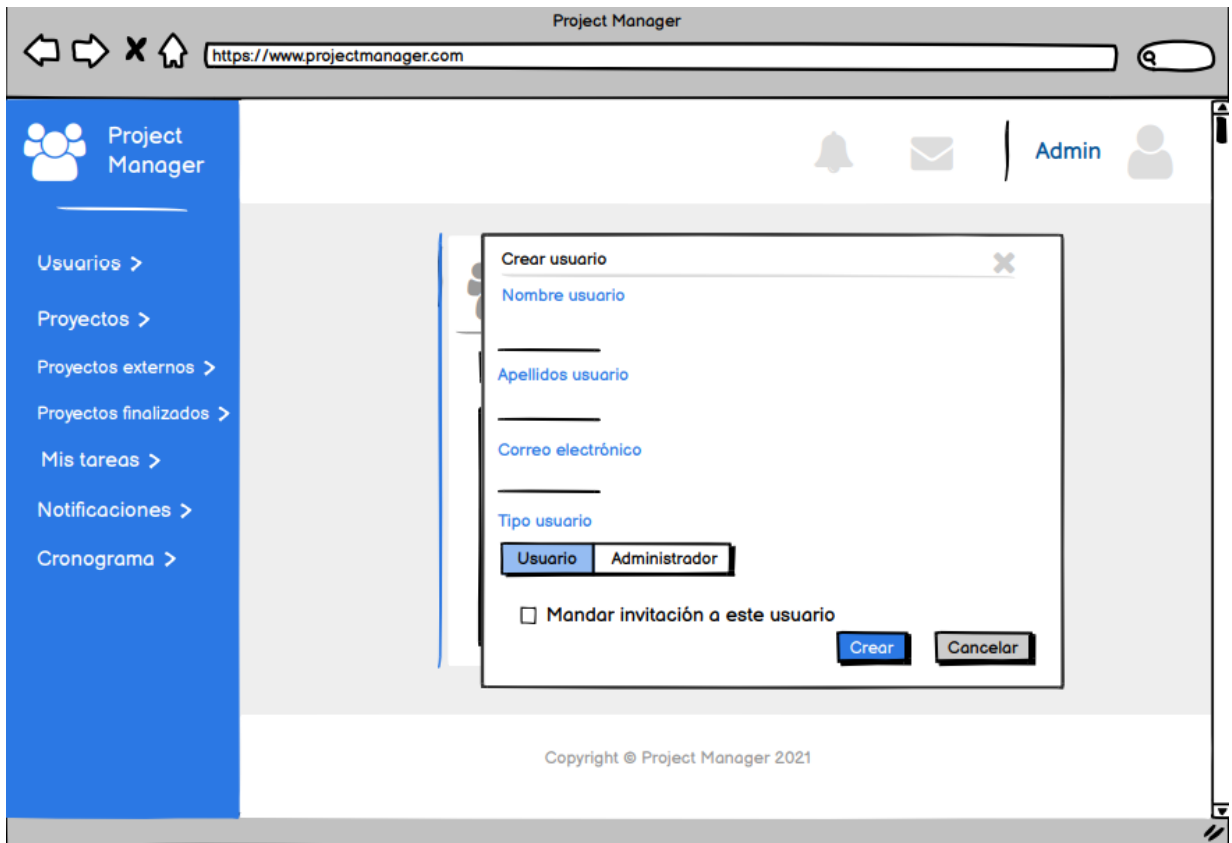


Figura 52 Anexo MIU: Maqueta de interfaz de usuario “Crear usuario”

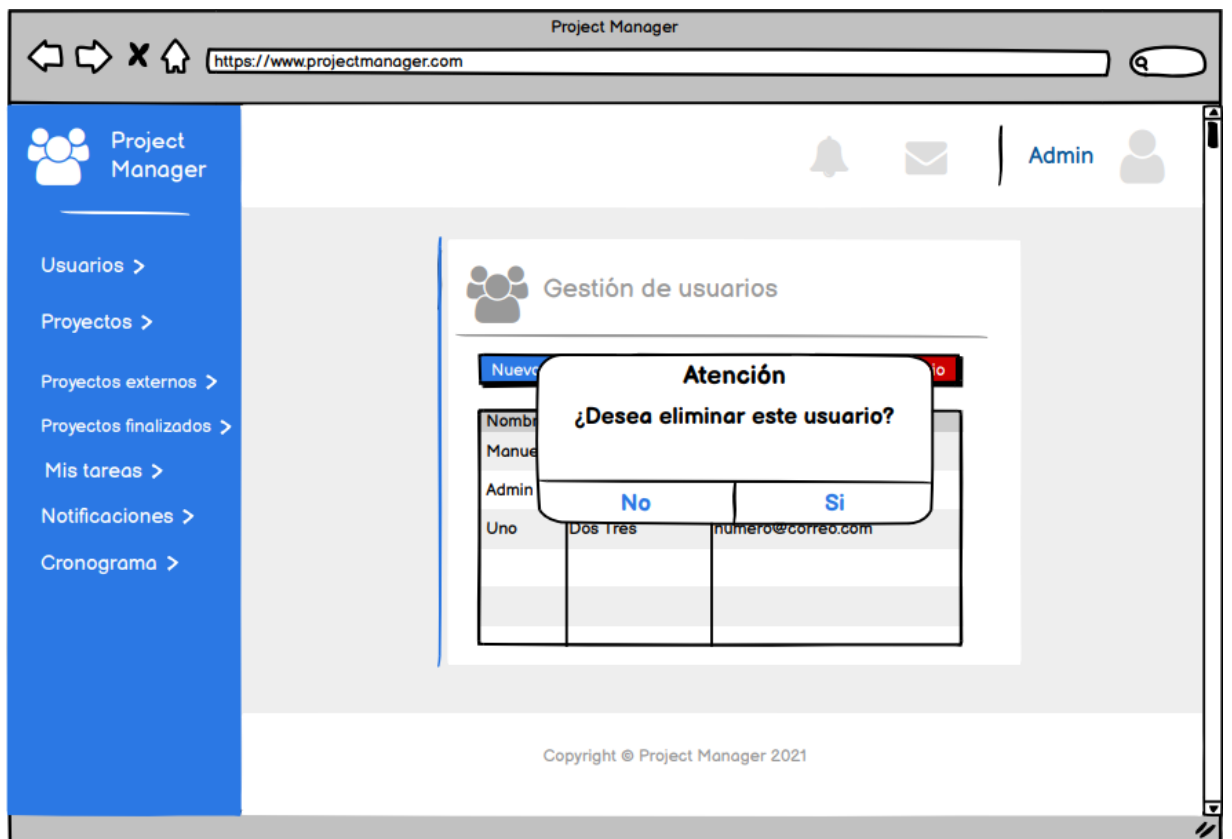


Figura 53 Anexo MIU: Maqueta de interfaz de usuario “Eliminar usuario”

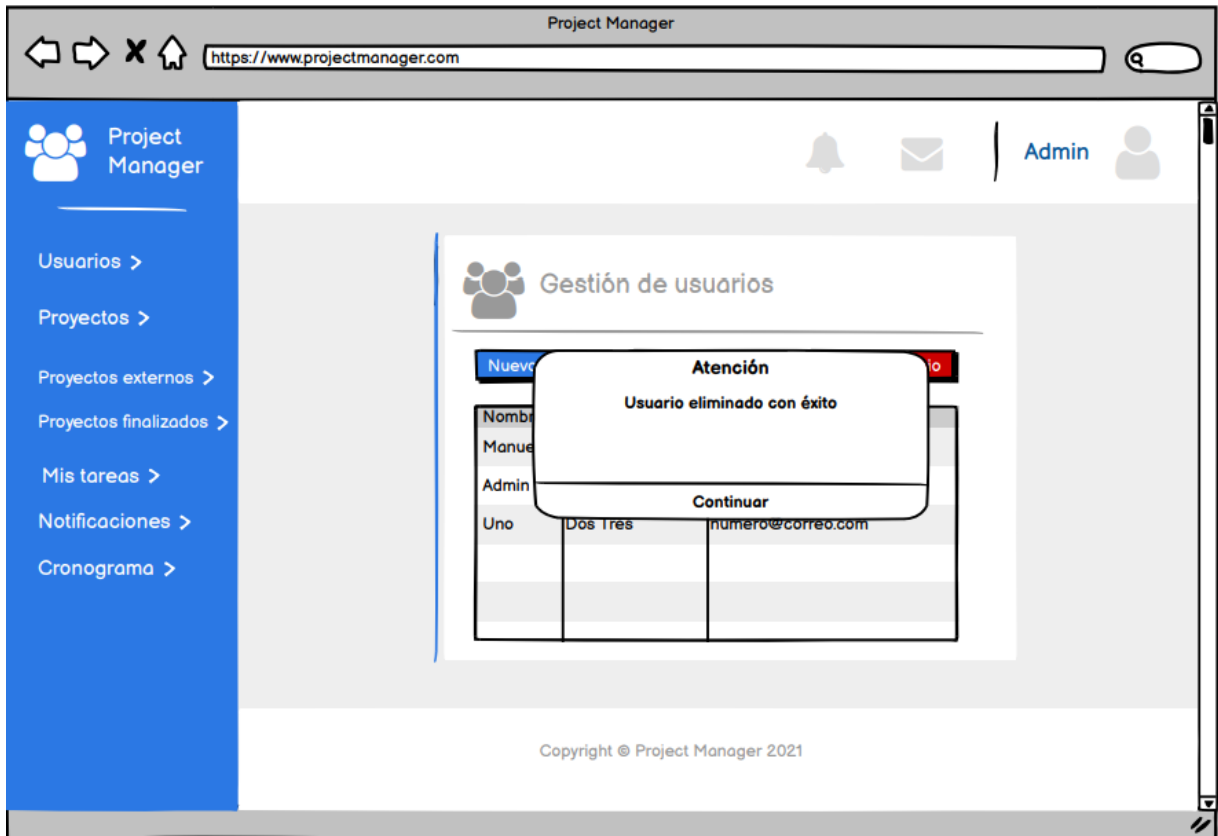


Figura 54 Anexo MIU: Maqueta de interfaz de usuario "Usuario eliminado"

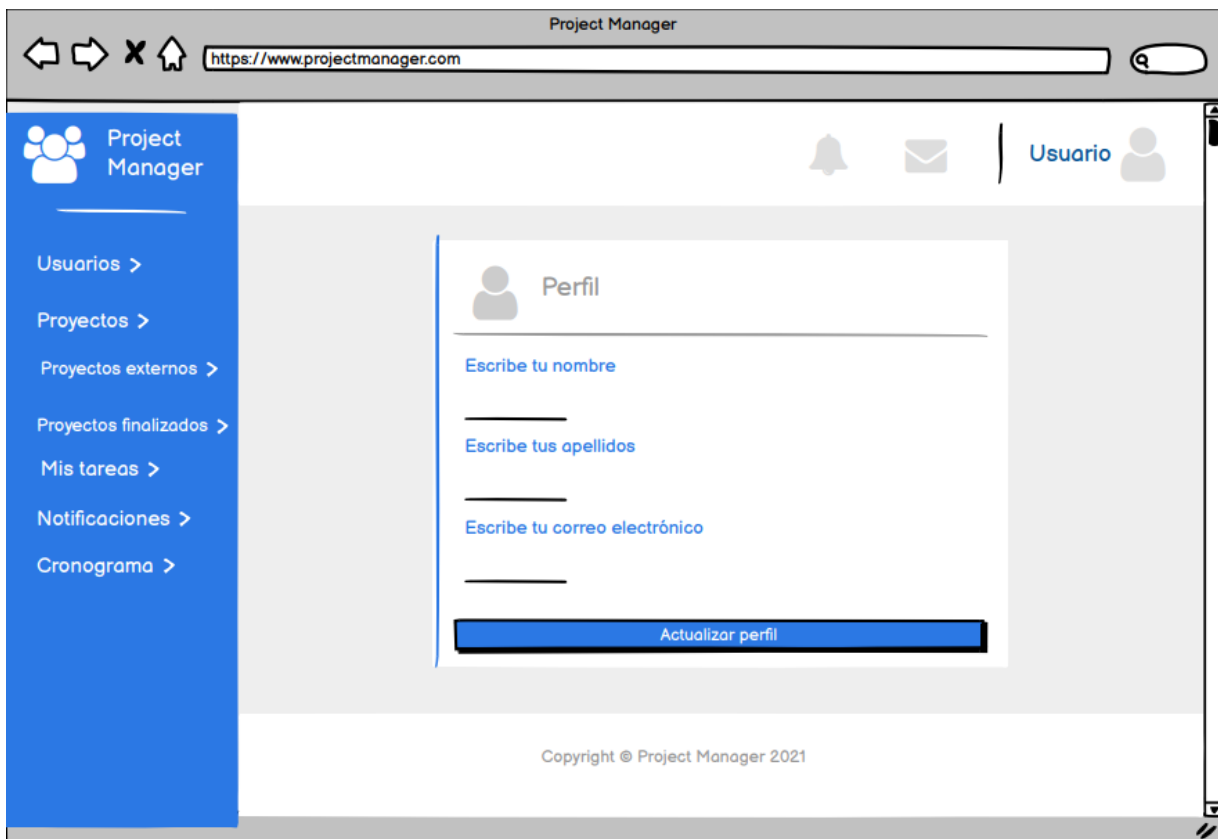


Figura 55 Anexo MIU: Maqueta de interfaz de usuario "Mi perfil"

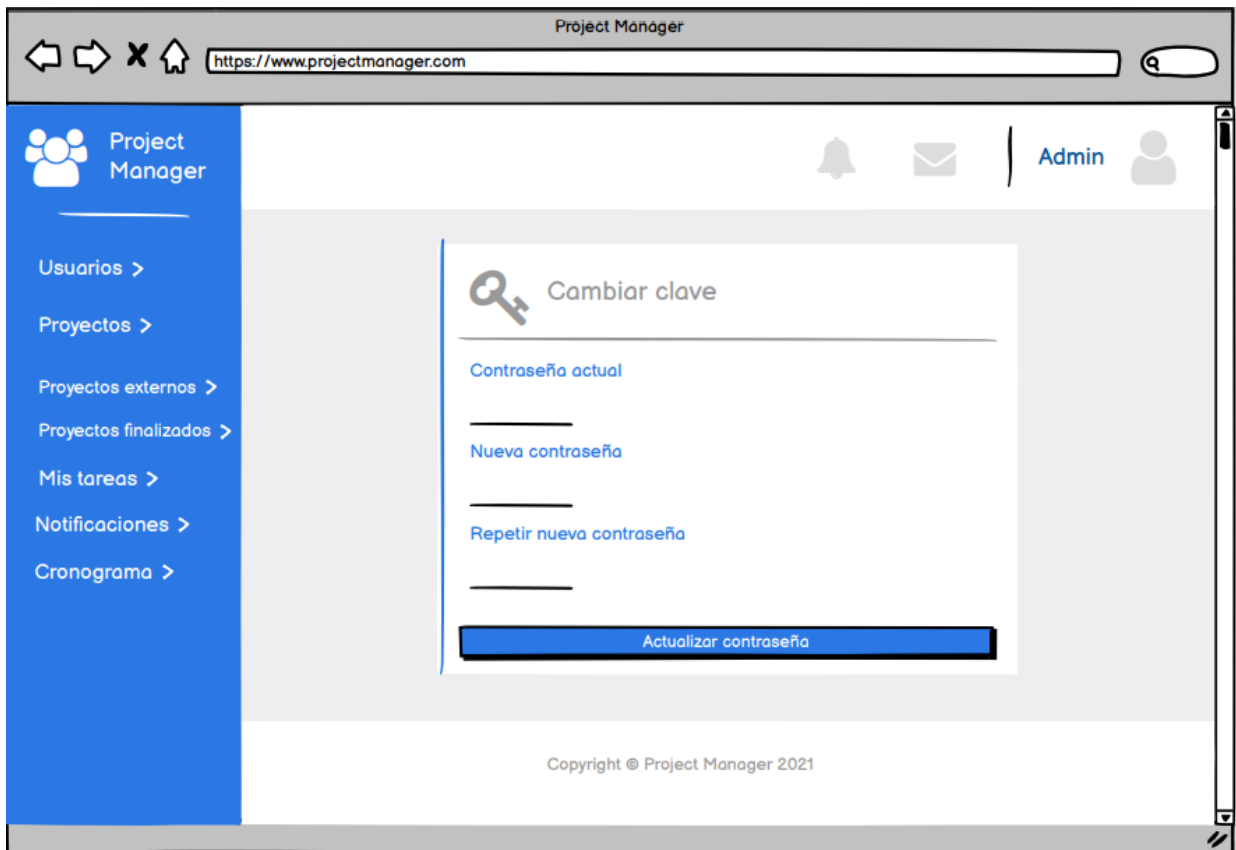


Figura 56 Anexo MIU: Maqueta de interfaz de usuario “Cambiar contraseña”

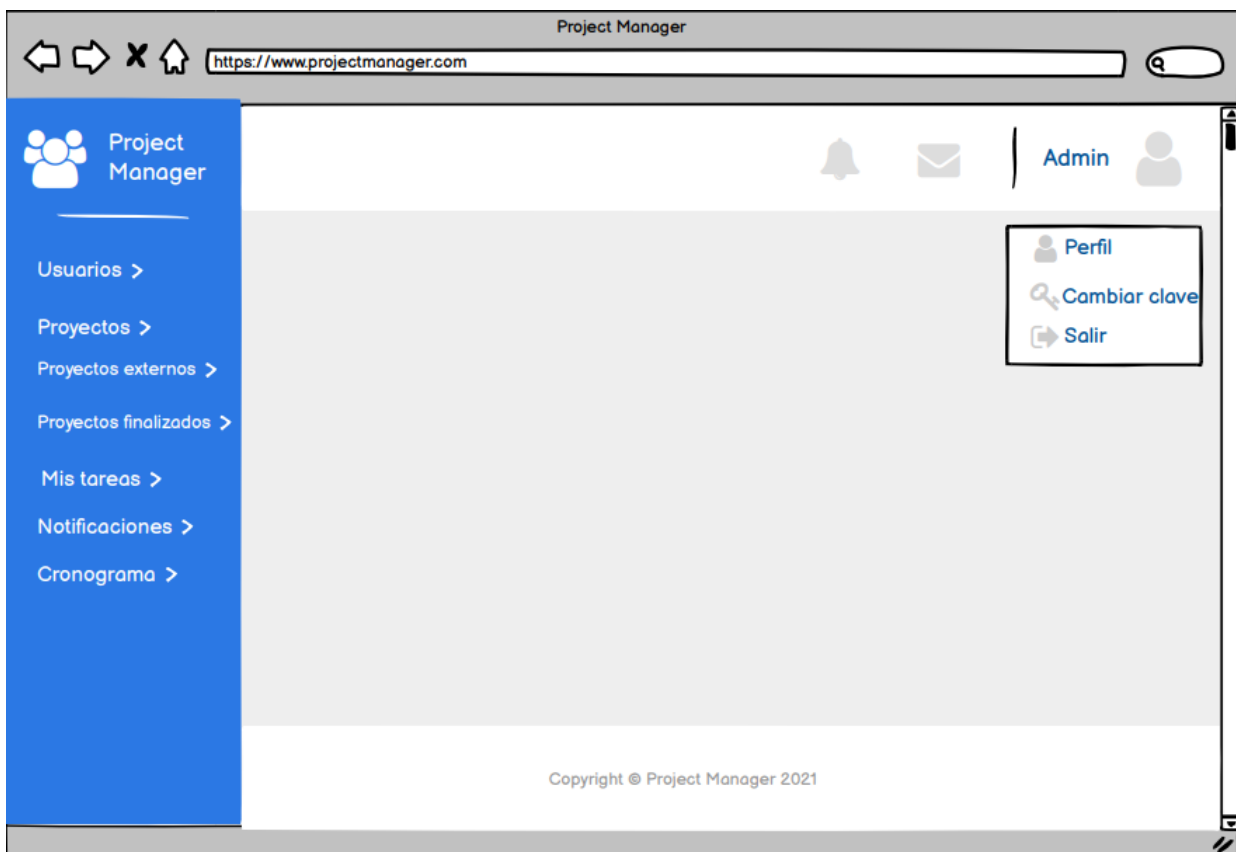


Figura 57 Anexo MIU: Maqueta de interfaz de usuario “Opciones de un usuario”

Anexo

Diagramas de secuencia

CRUD Usuarios

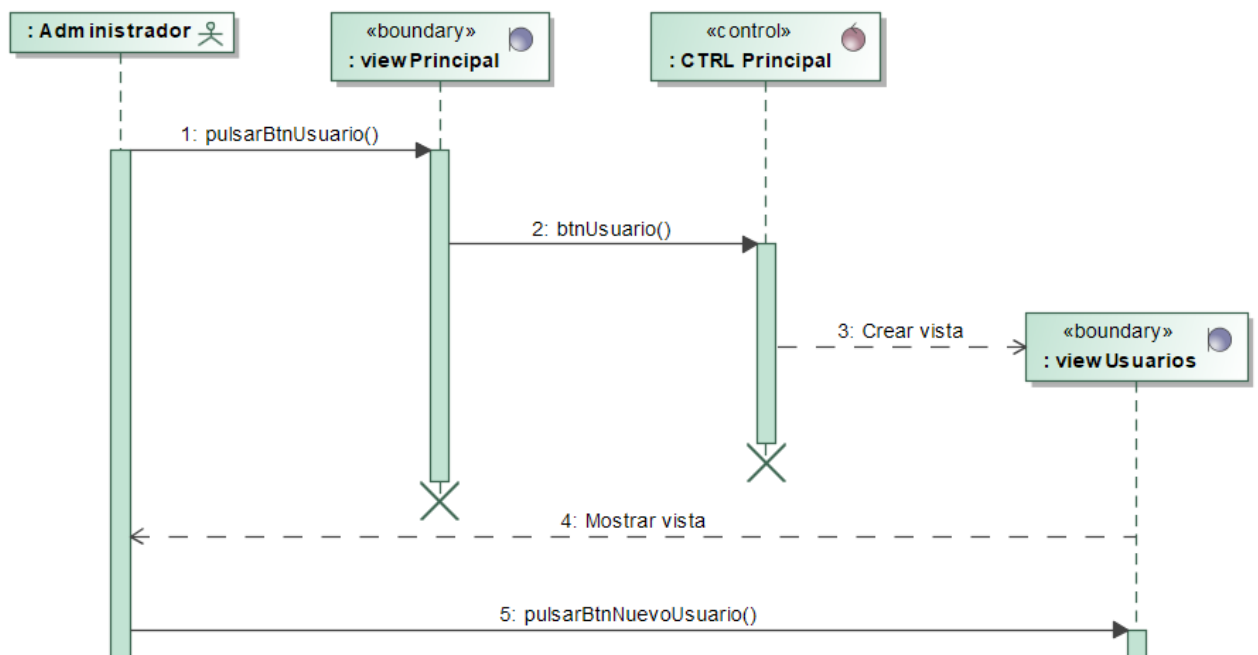


Figura 1 Anexo DS: Crear usuario (Parte 1)

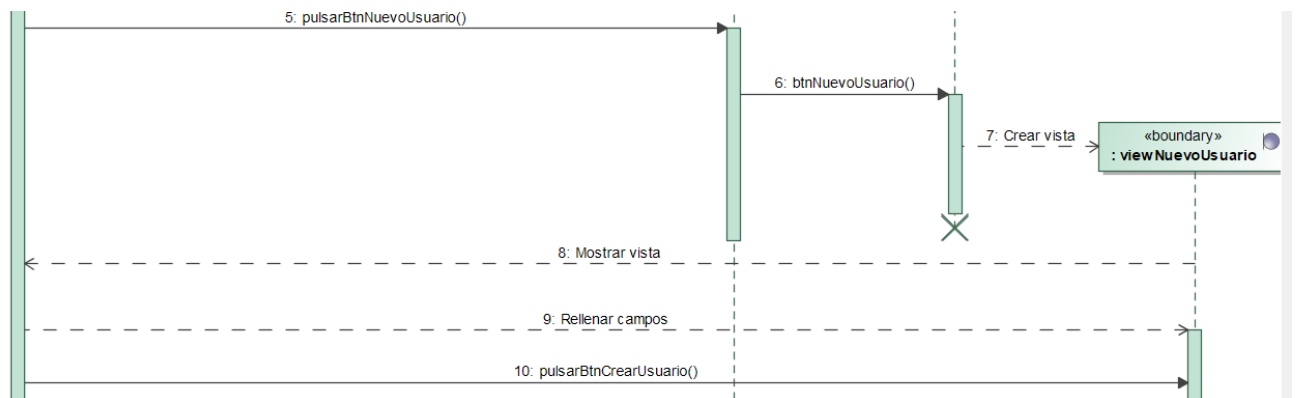


Figura 2 Anexo DS: Crear usuario (Parte 2)

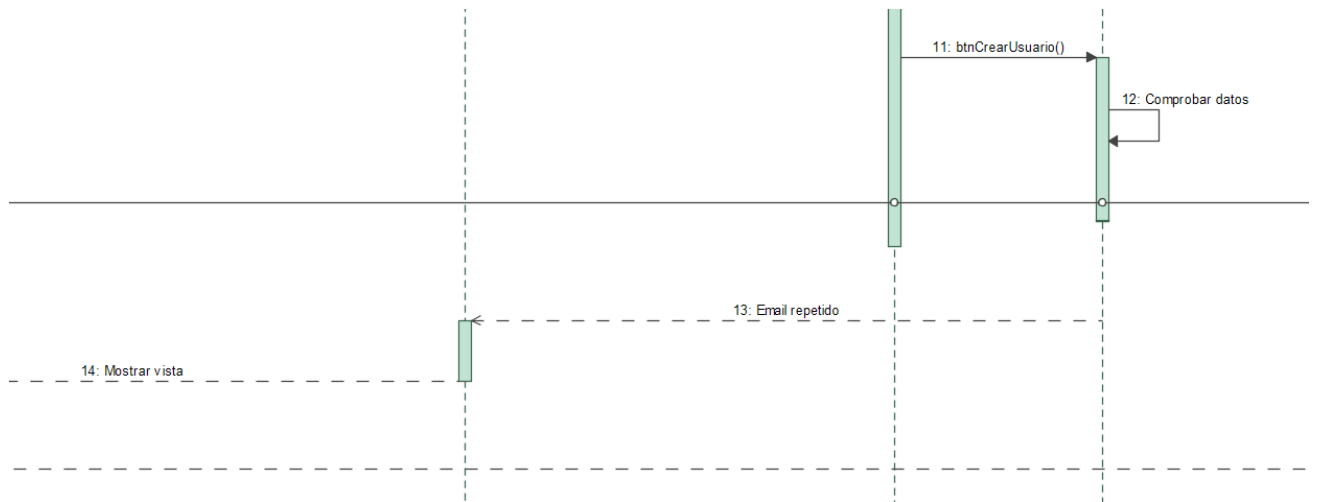


Figura 3 Anexo DS: Crear usuario (Parte 3)

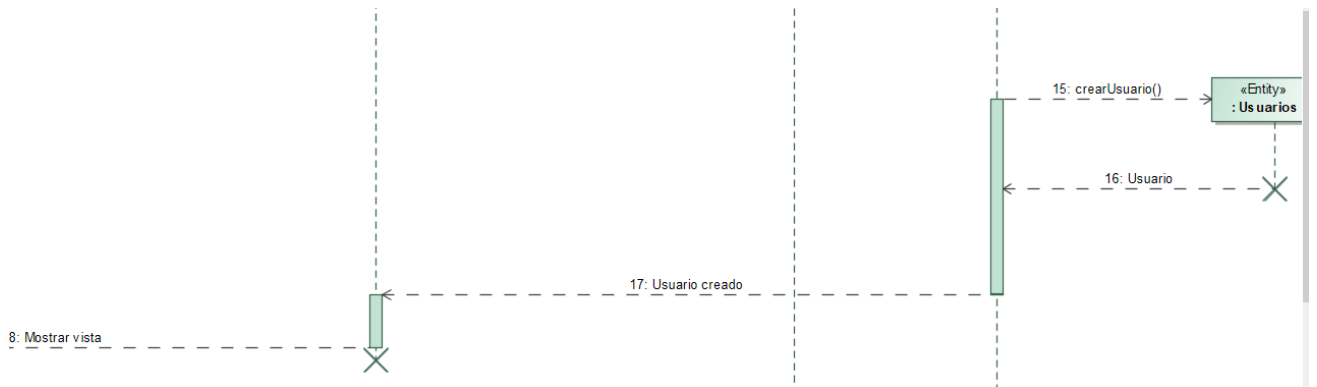


Figura 4 Anexo DS: Crear usuario (Parte 4)

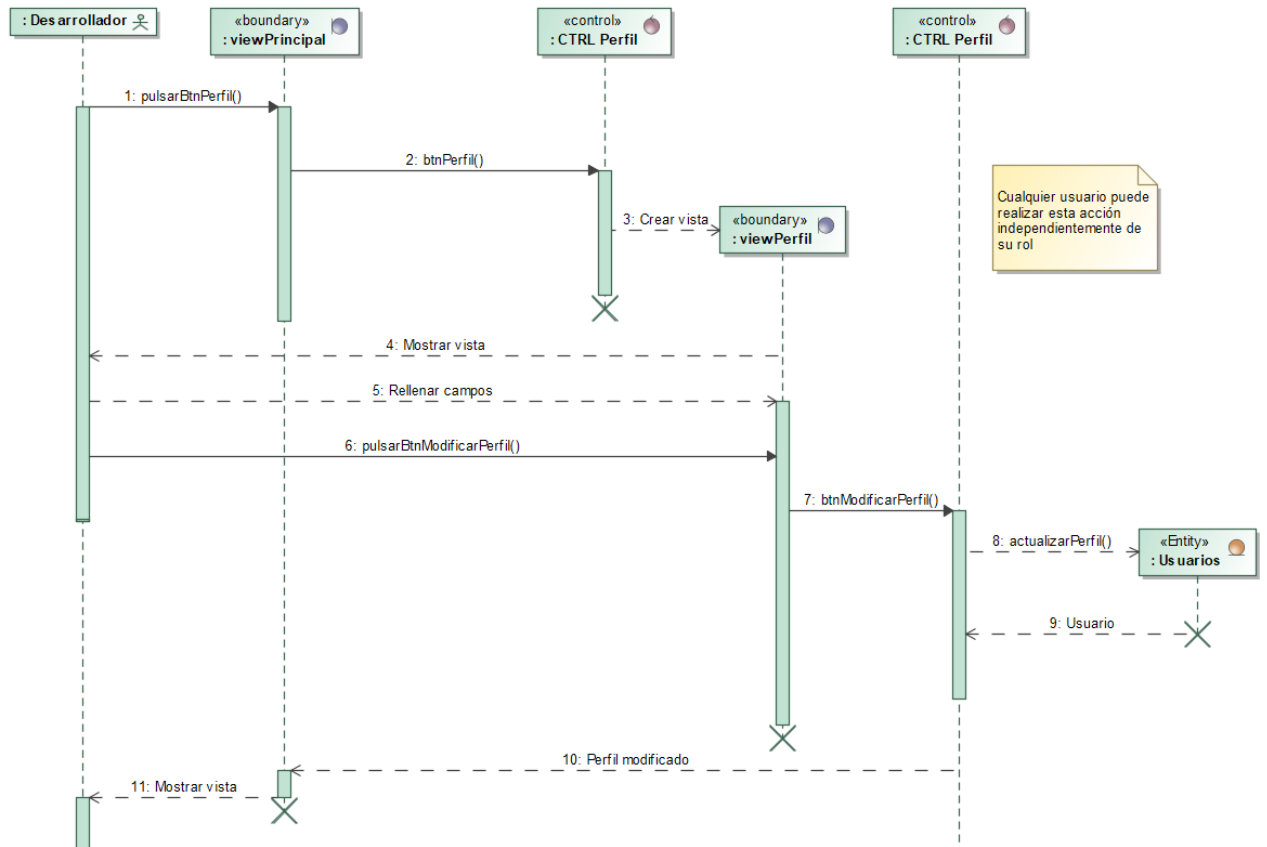


Figura 5 Anexo DS: Leer y actualizar usuario

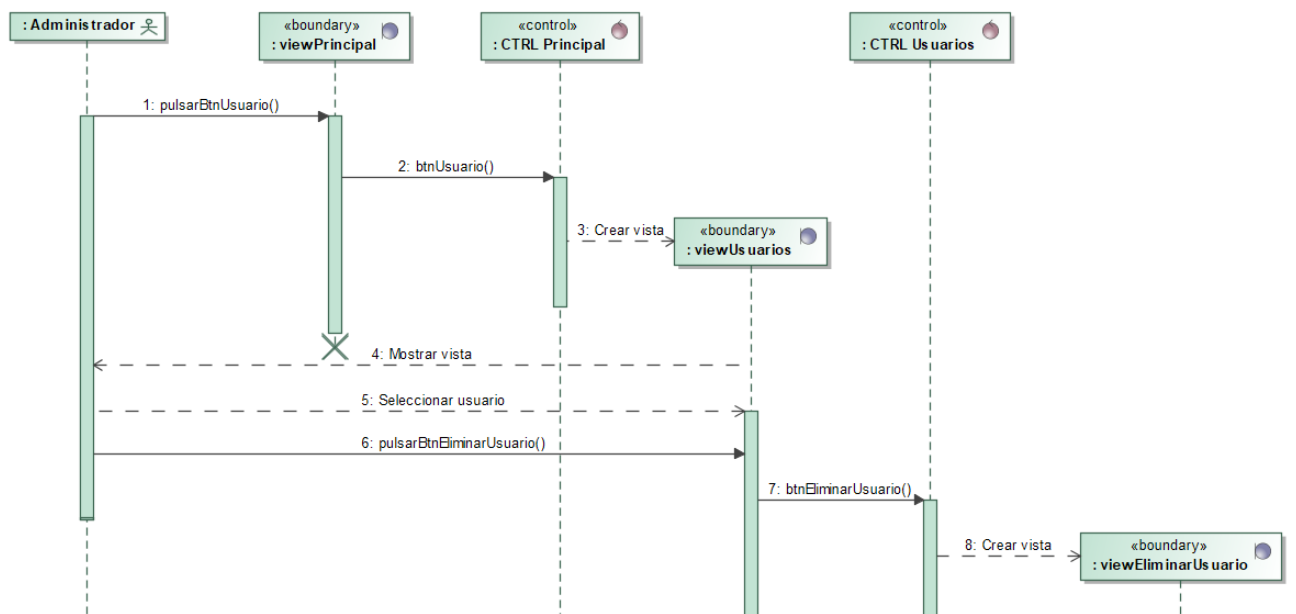


Figura 6 Anexo DS: Eliminar usuario (Parte 1)

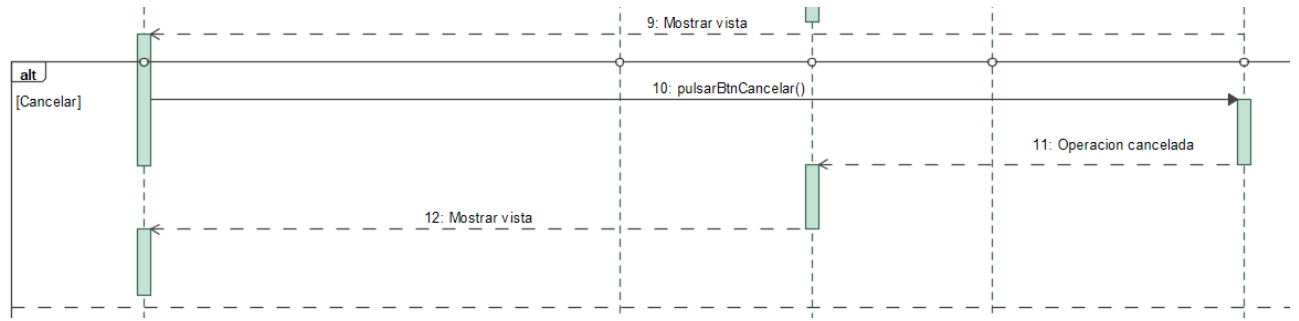


Figura 7 Anexo DS: Eliminar usuario (Parte 2)

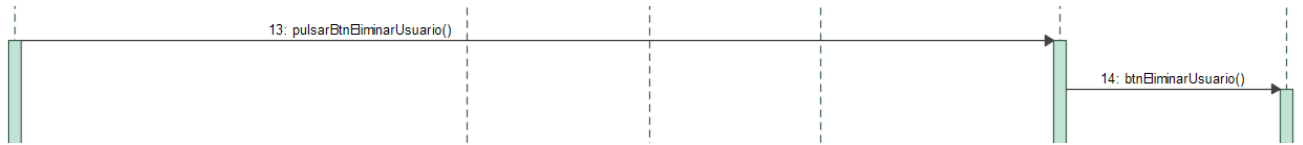


Figura 8 Anexo DS: Eliminar usuario (Parte 3)

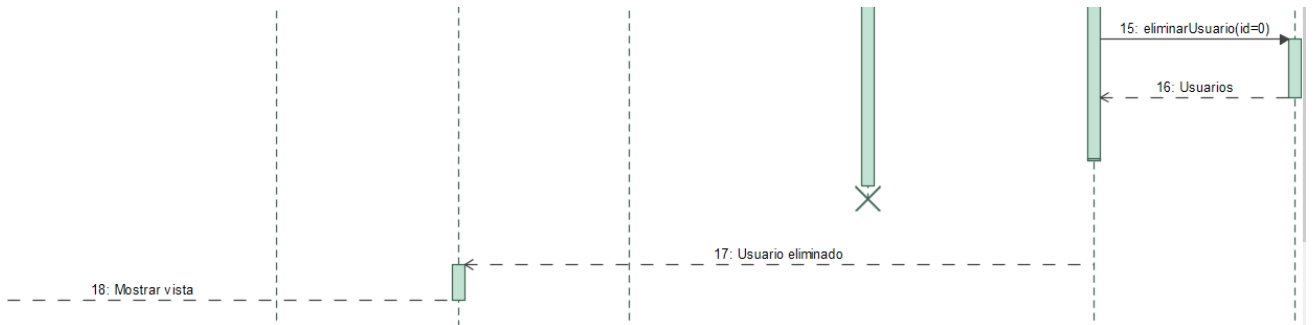


Figura 9 Anexo DS: Eliminar usuario (Parte 4)

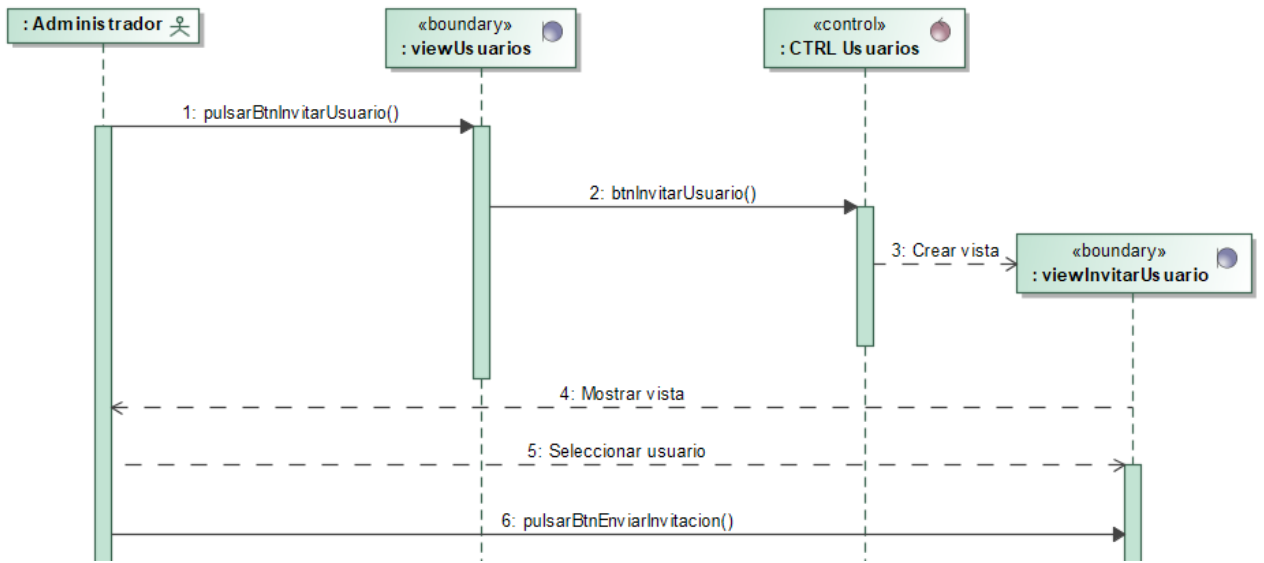


Figura 10 Anexo DS: Invitar usuario (Parte 1)

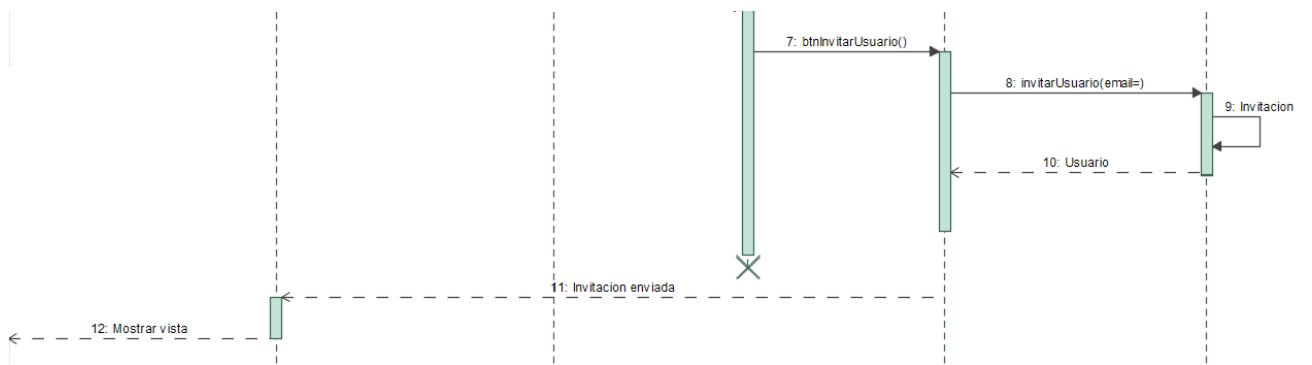


Figura 11 Anexo DS: Invitar usuario (Parte 2)

CRUD Proyectos

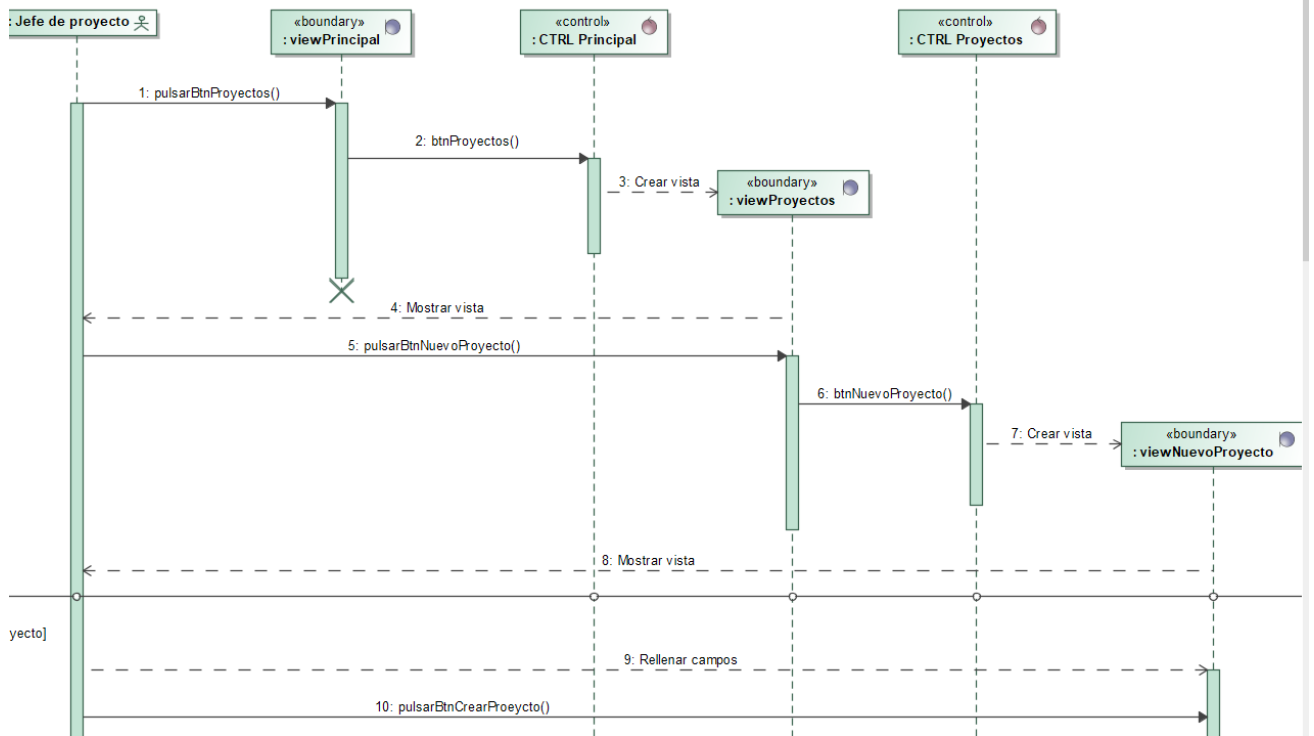


Figura 12 Anexo DS: Crear proyecto (Parte 1)

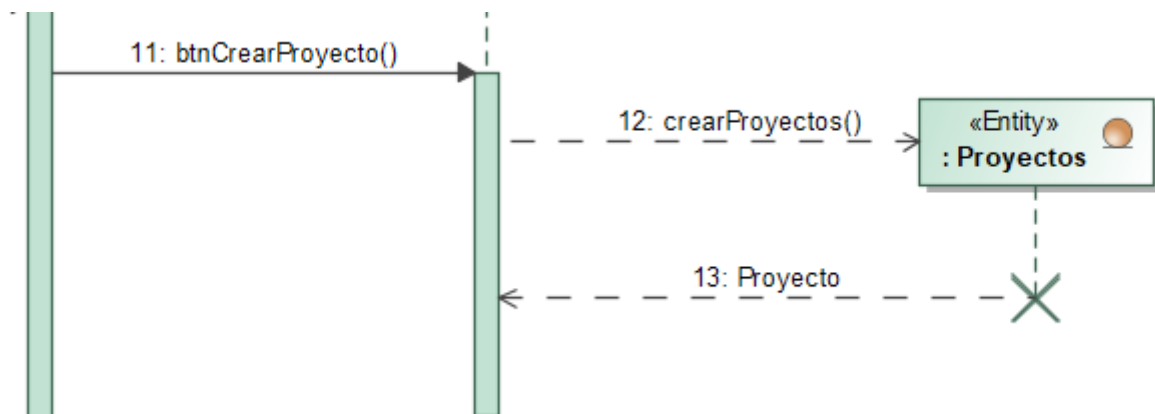


Figura 13 Anexo DS: Crear proyecto (Parte 2)

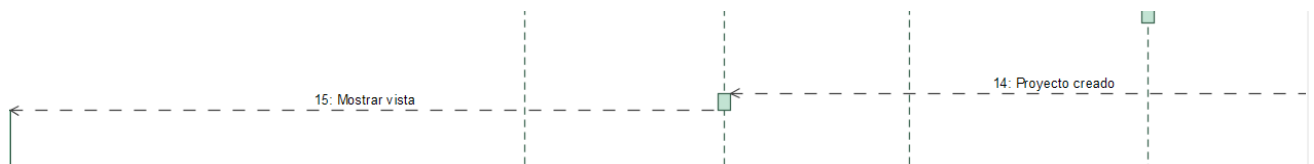


Figura 14 Anexo DS: Crear proyecto (Parte 3)

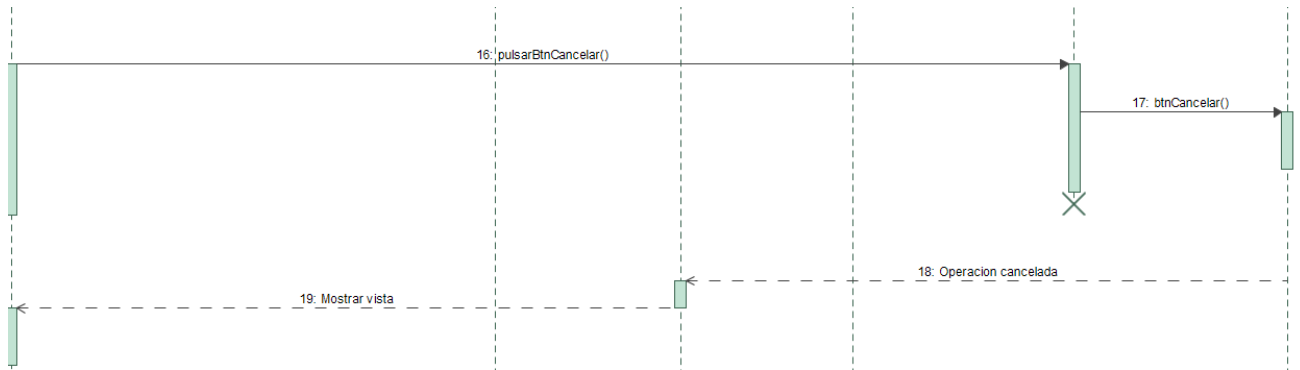


Figura 15 Anexo DS: Crear proyecto (Parte 4)

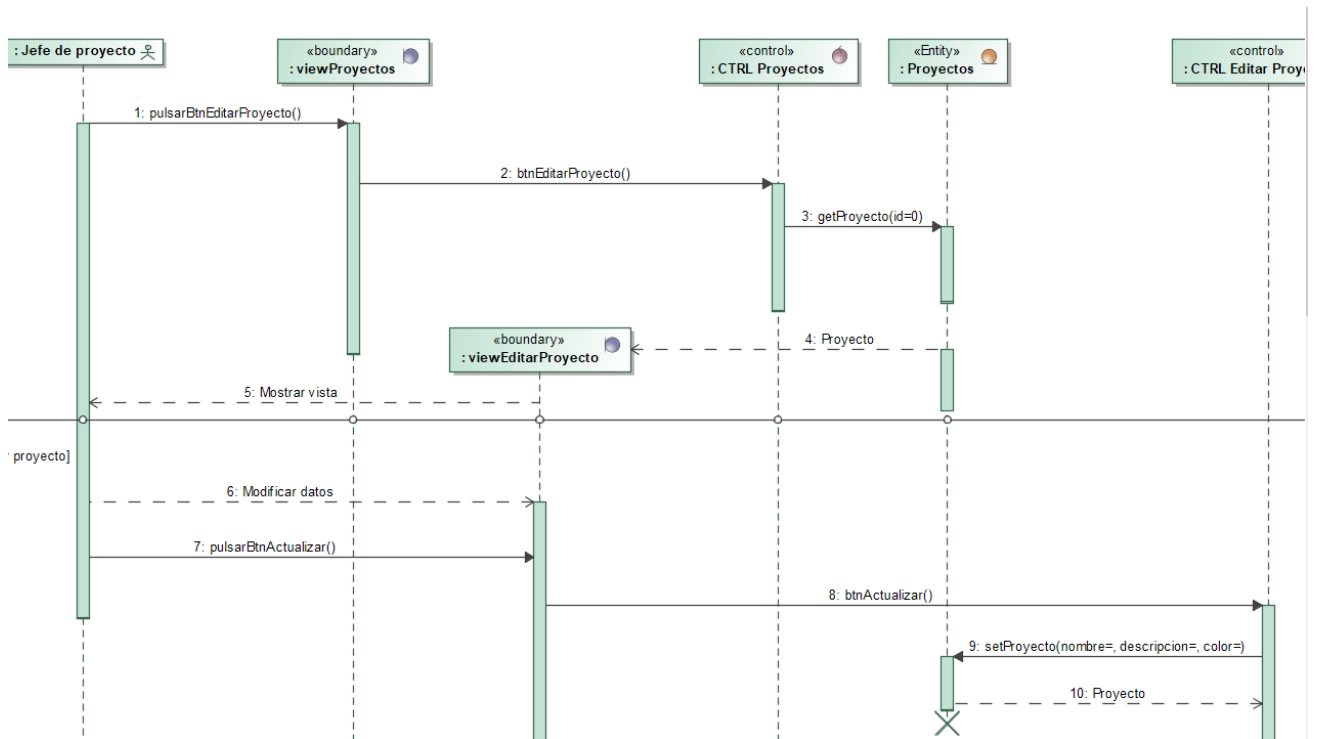


Figura 16 Anexo DS: Leer y actualizar proyecto (Parte 1)

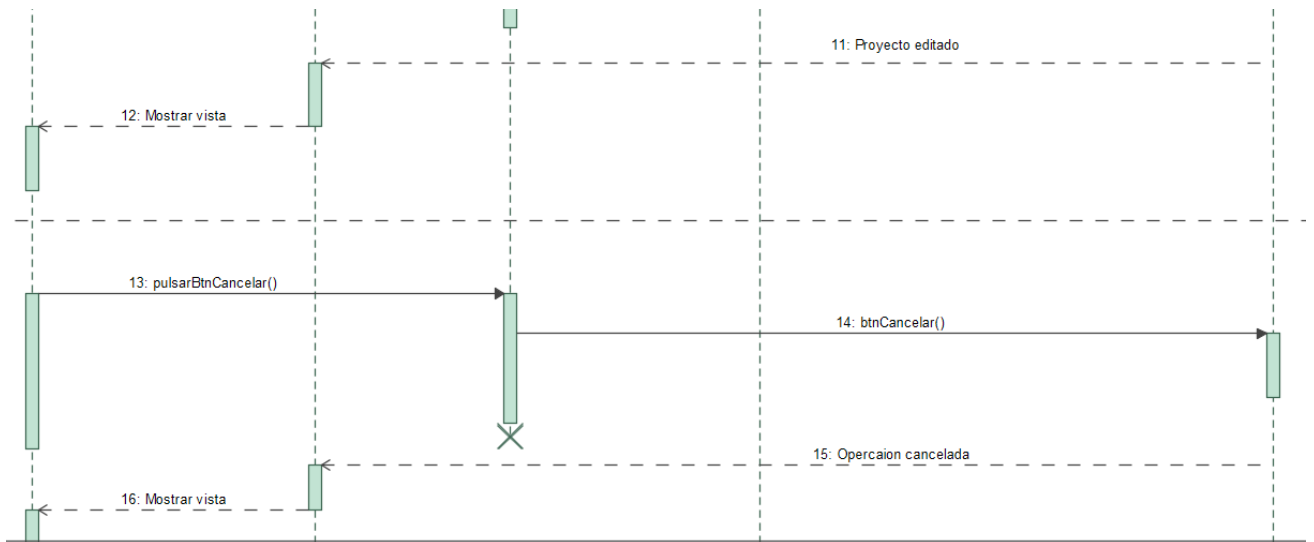


Figura 17 Anexo DS: Leer y actualizar proyecto (Parte 2)

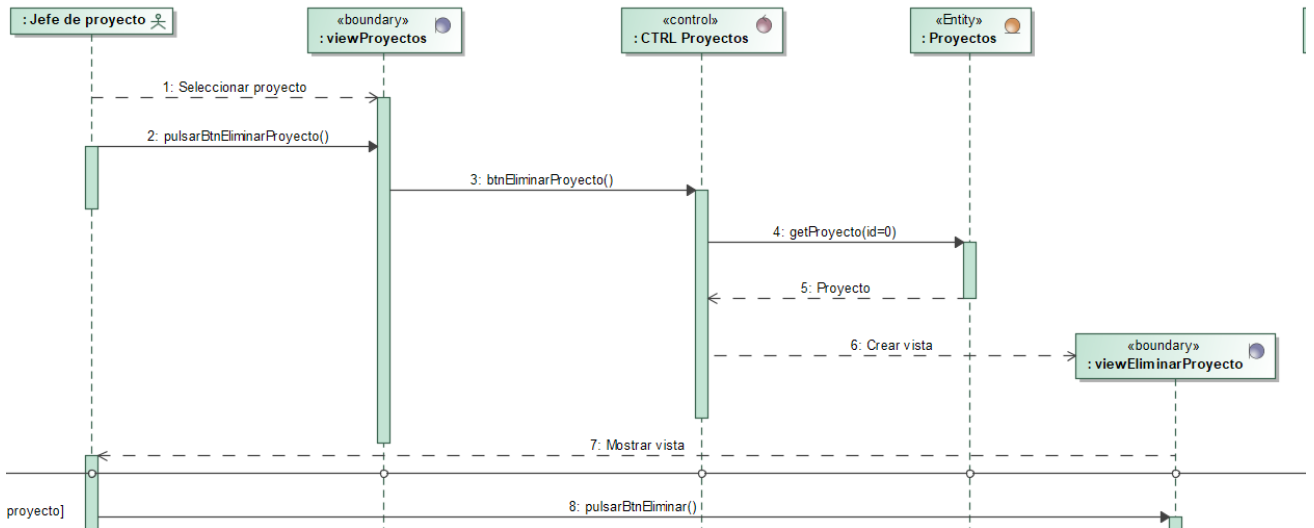


Figura 18 Anexo DS: Eliminar proyecto (Parte 1)

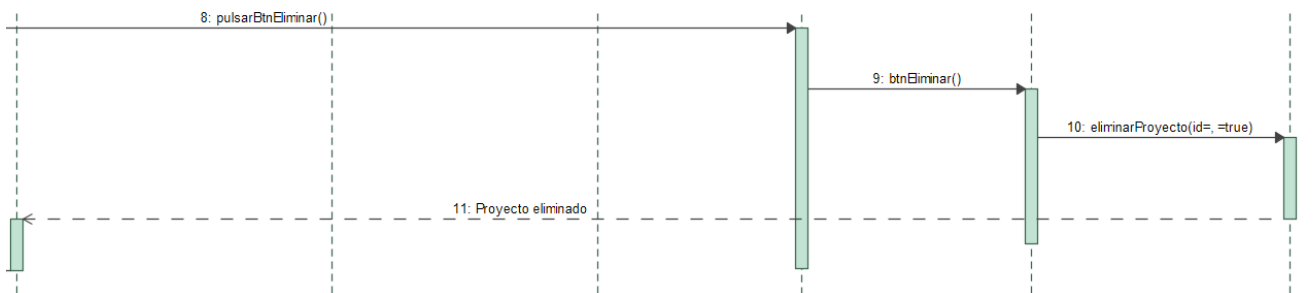


Figura 19 Anexo DS: Eliminar proyecto (Parte 2)



Figura 20 Anexo DS: Eliminar proyecto (Parte 3)

CRUD Sprints

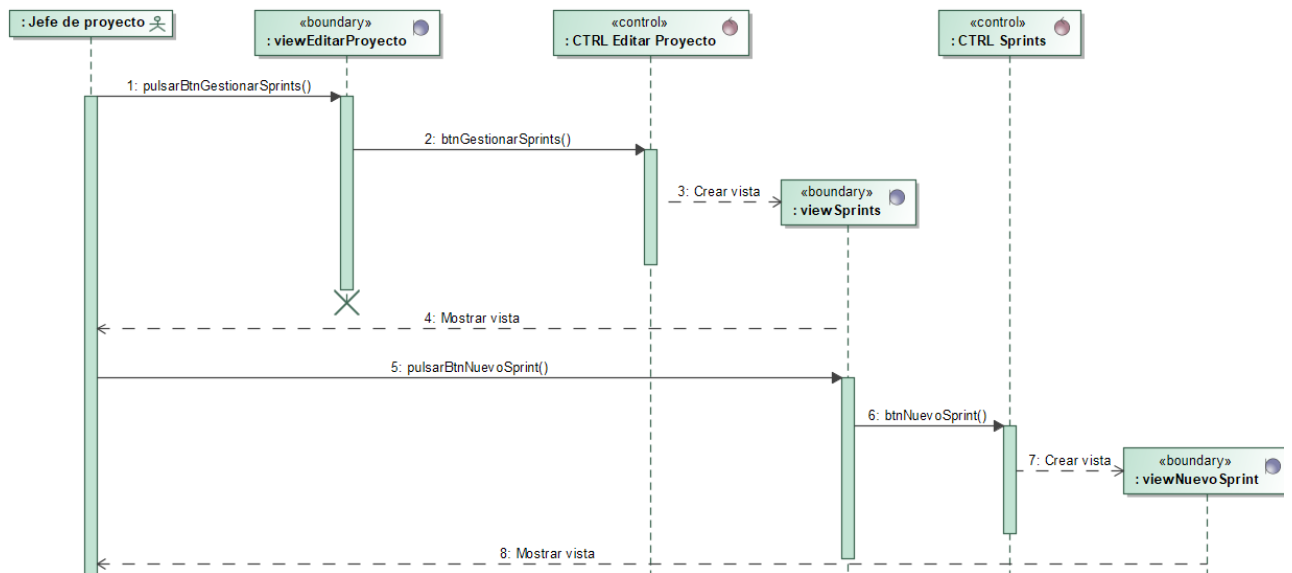


Figura 21 Anexo DS: Crear, leer y modificar sprint (Parte 1)

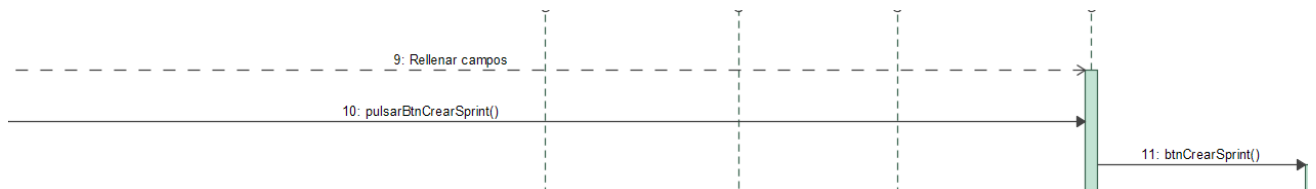


Figura 22 Anexo DS: Crear, leer y modificar sprint (Parte 2)

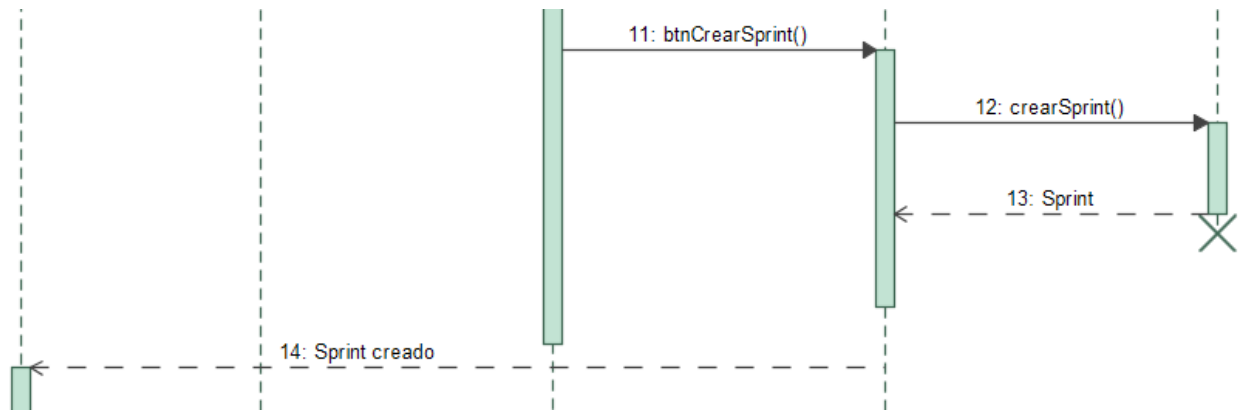


Figura 23 Anexo DS: Crear, leer y modificar sprint (Parte 3)

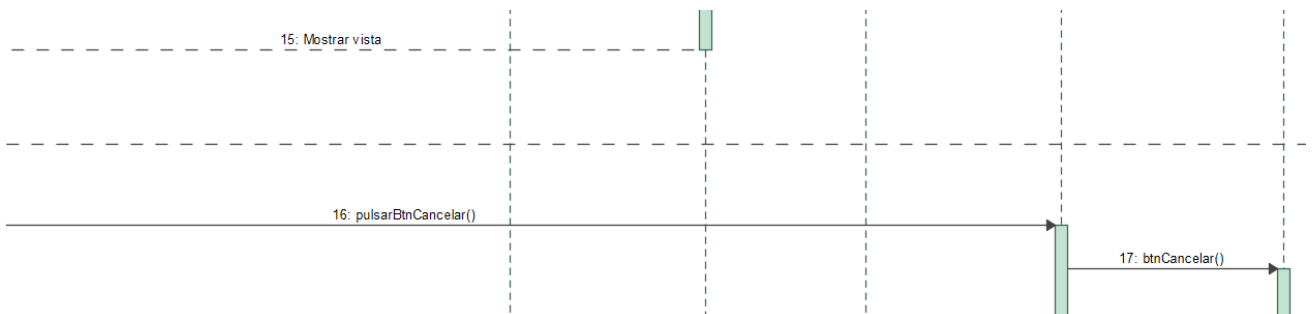


Figura 24 Anexo DS: Crear, leer y modificar sprint (Parte 4)

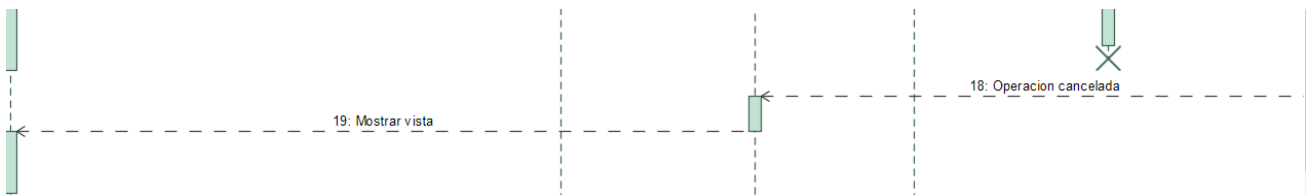


Figura 25 Anexo DS: Crear, leer y modificar sprint (Parte 5)

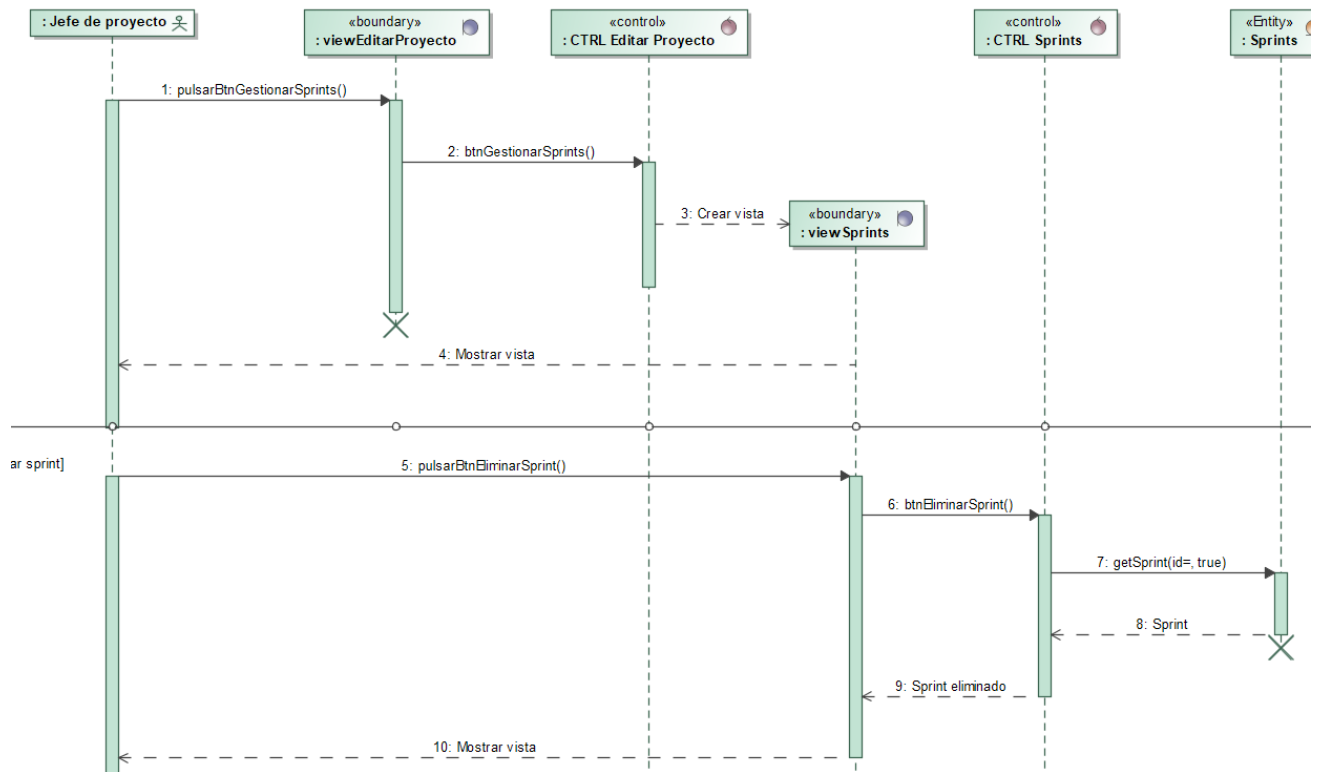


Figura 26 Anexo DS: Eliminar sprint (Parte 1)

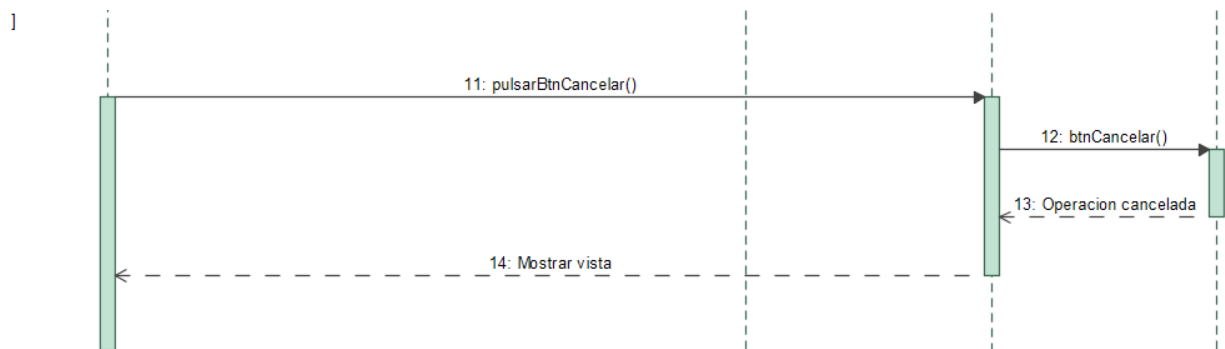


Figura 27 Anexo DS: Eliminar sprint (Parte 2)

CRUD Historias de usuario

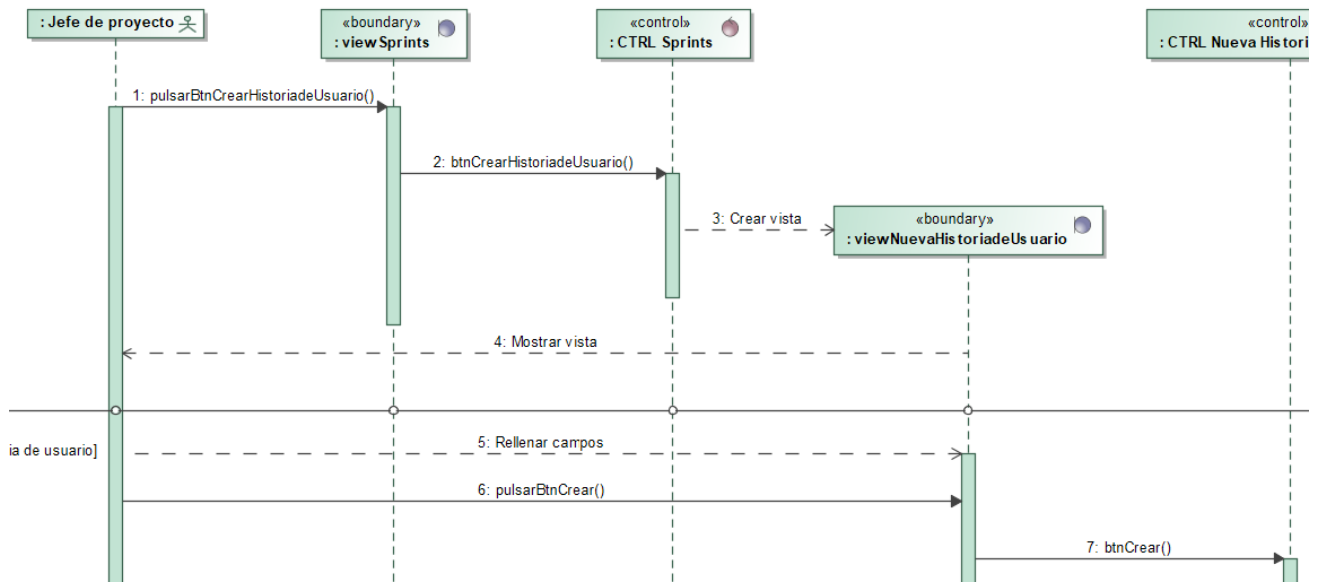


Figura 28 Anexo DS: Crear historia de usuario (Parte 1)

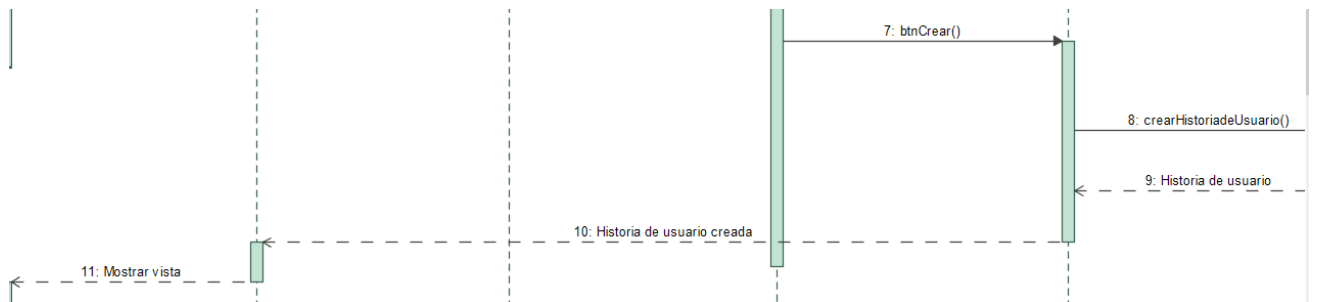


Figura 29 Anexo DS: Crear historia de usuario (Parte 2)

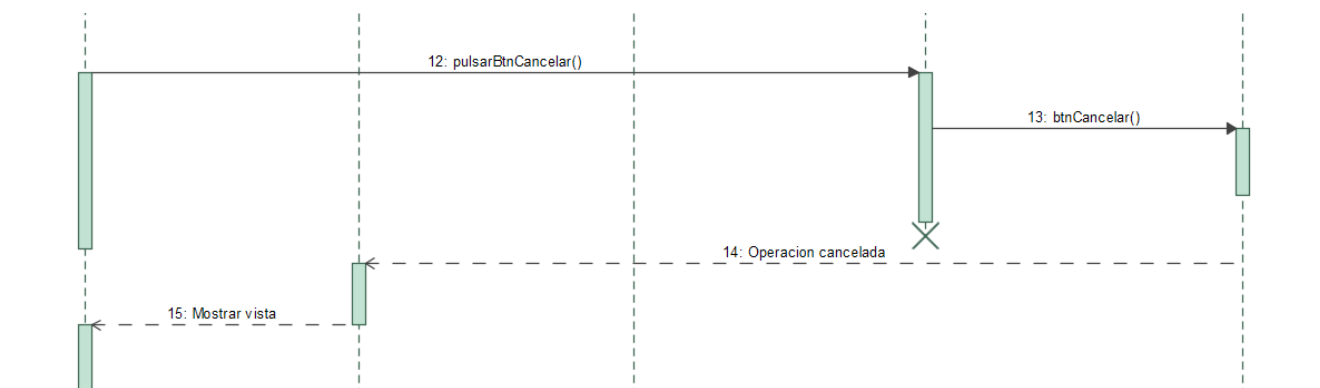


Figura 30 Anexo DS: Crear historia de usuario (Parte 3)

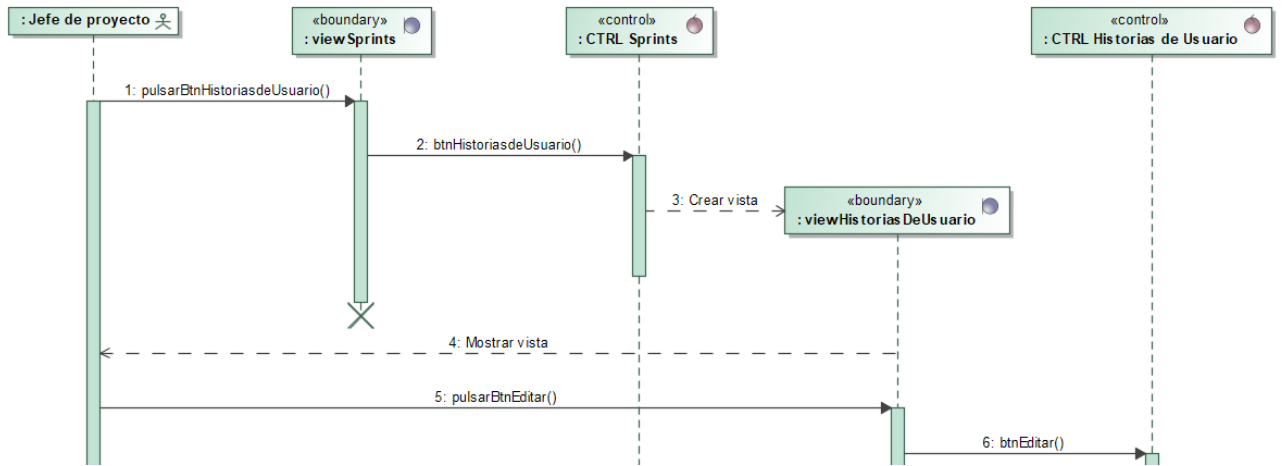


Figura 31 Anexo DS: Leer y modificar historia de usuario (Parte 1)

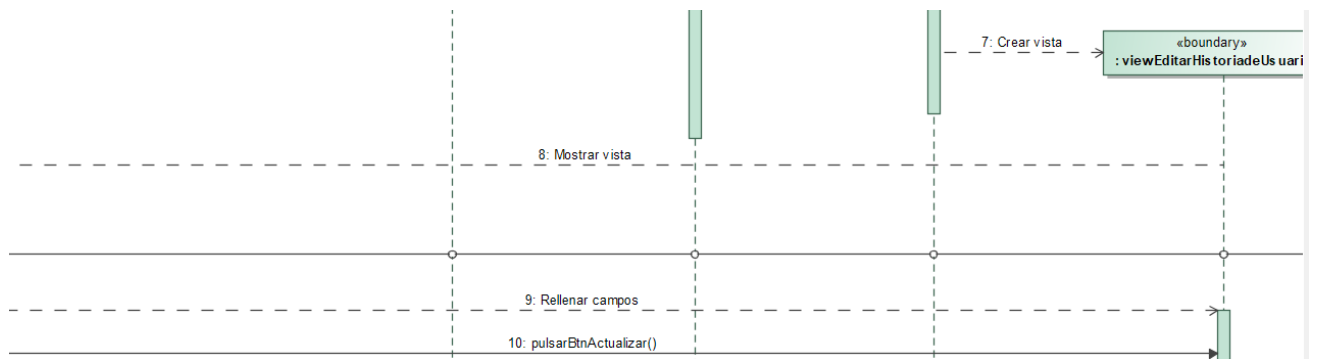


Figura 32 Anexo DS: Leer y modificar historia de usuario (Parte 2)

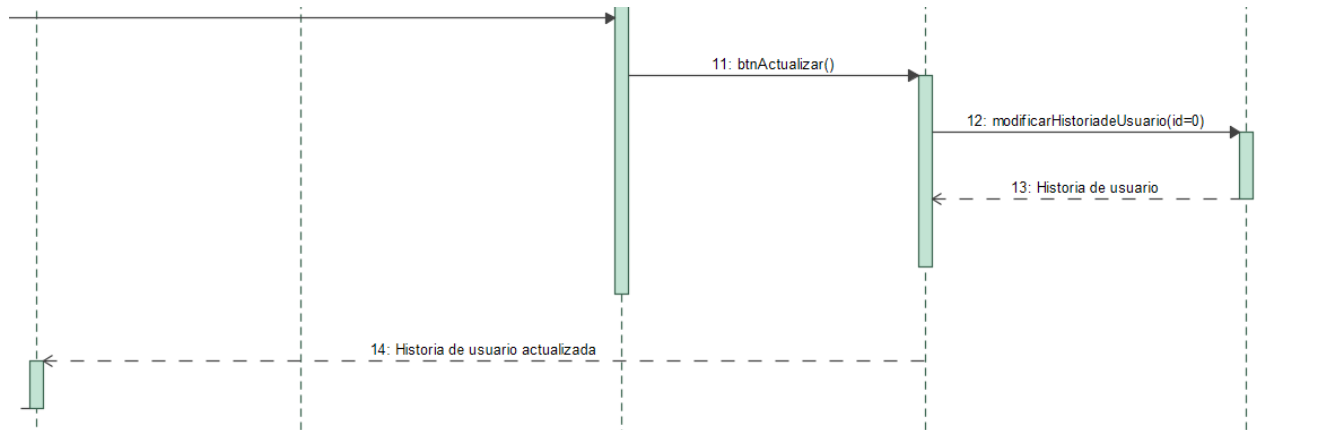


Figura 33 Anexo DS: Leer y modificar historia de usuario (Parte 3)

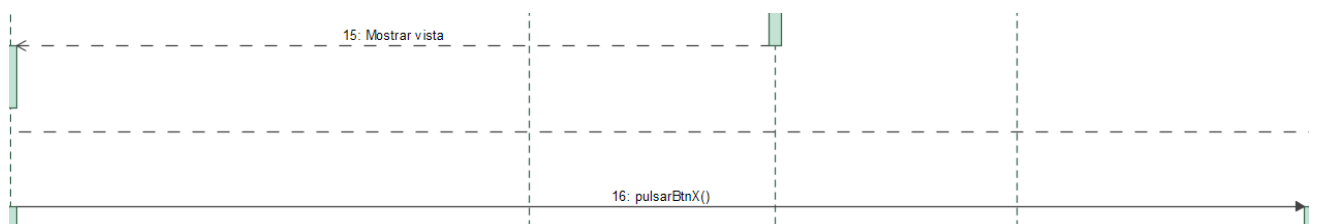


Figura 34 Anexo DS: Leer y modificar historia de usuario (Parte 4)

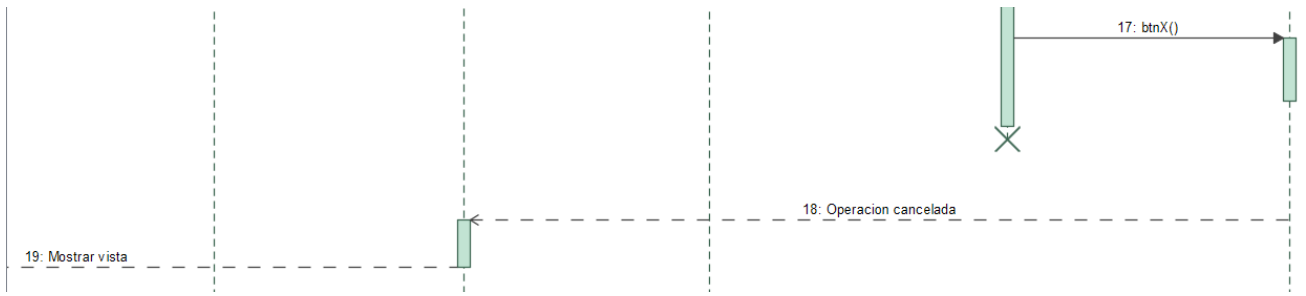


Figura 35 Anexo DS: Leer y modificar historia de usuario (Parte 5)

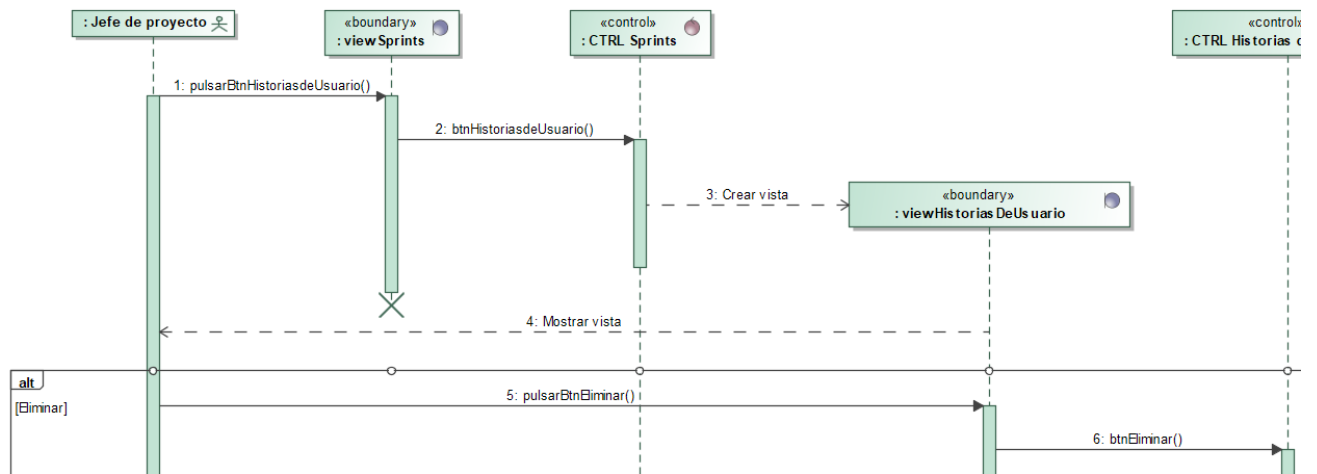


Figura 36 Anexo DS: Eliminar historia de usuario (Parte 1)

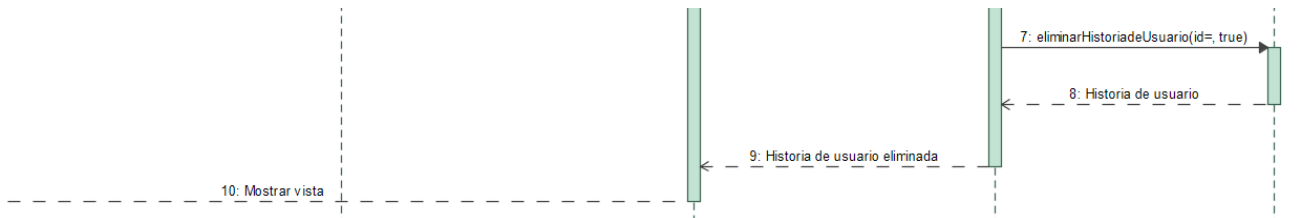


Figura 37 Anexo DS: Eliminar historia de usuario (Parte 2)

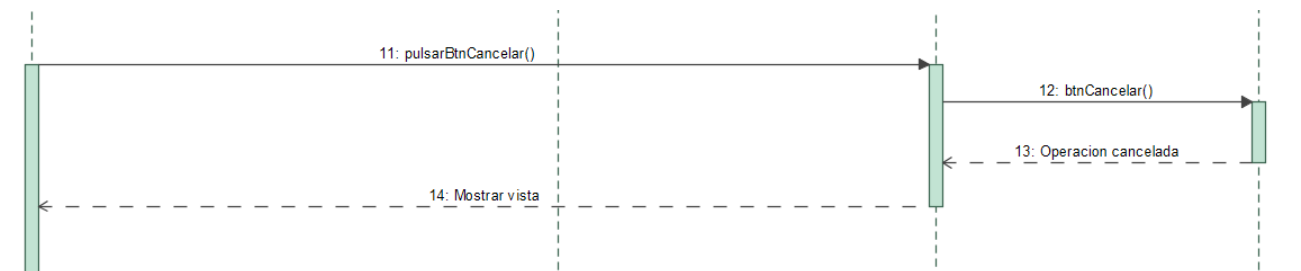


Figura 38 Anexo DS: Eliminar historia de usuario (Parte 3)

Adjuntar archivos a historias de usuario

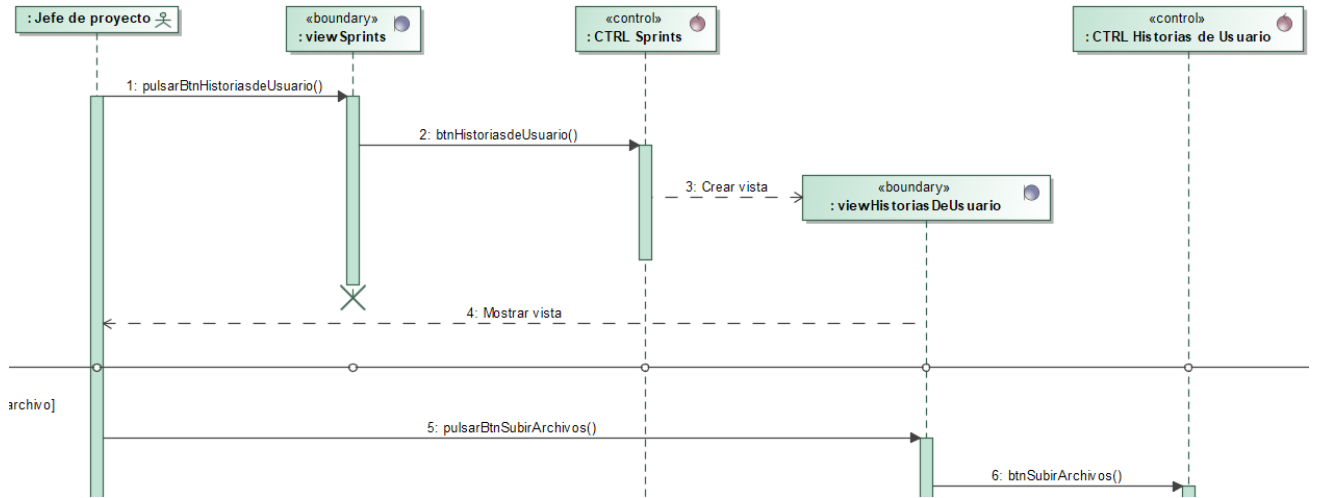


Figura 39 Anexo DS: Adjuntar archivos a historias de usuario (Parte 1)

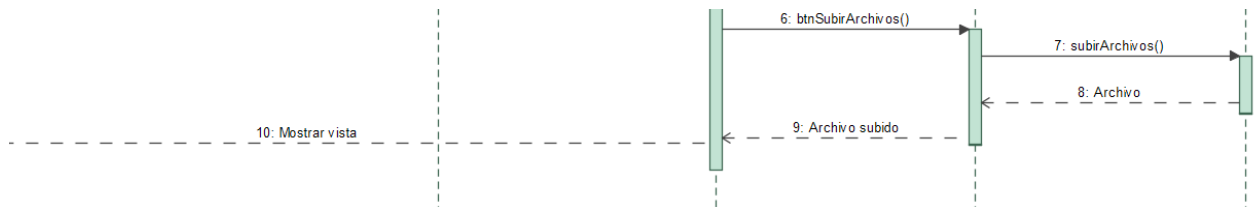


Figura 40 Anexo DS: Adjuntar archivos a historias de usuario (Parte 2)

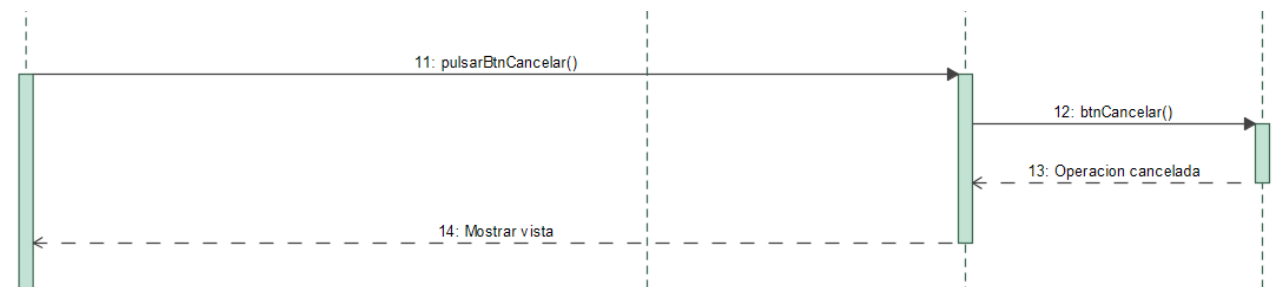


Figura 41 Anexo DS: Adjuntar archivos a historias de usuario (Parte 3)

Asignar responsable a historia de usuario

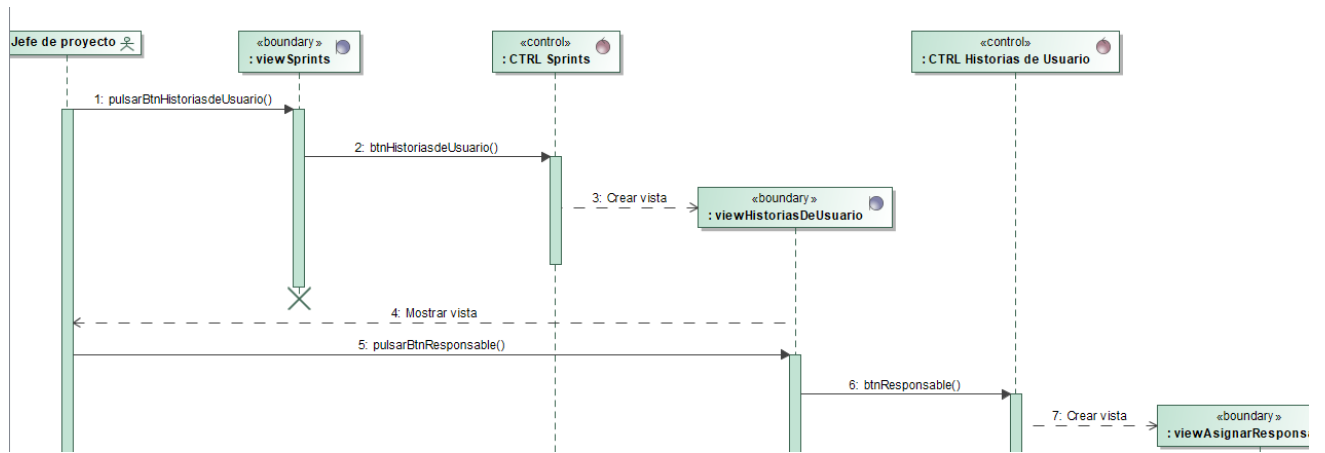


Figura 42 Anexo DS: Asignar responsable a historia de usuario (Parte 1)

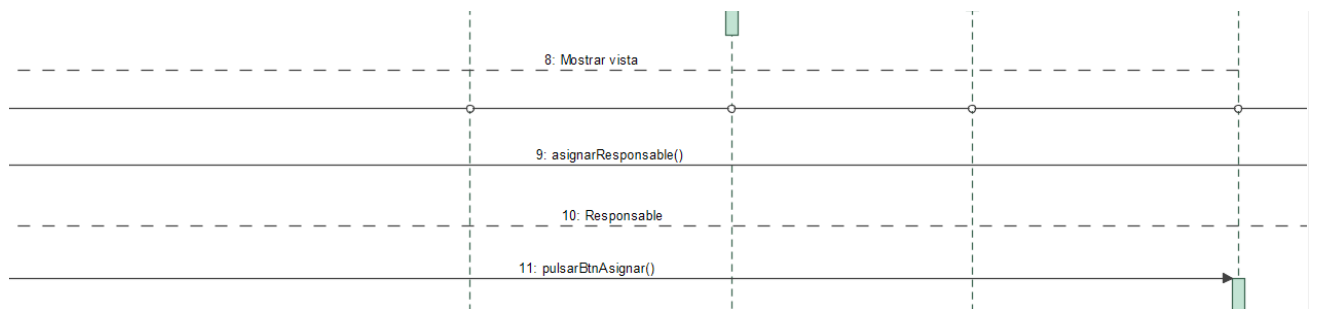


Figura 43 Anexo DS: Asignar responsable a historia de usuario (Parte 2)

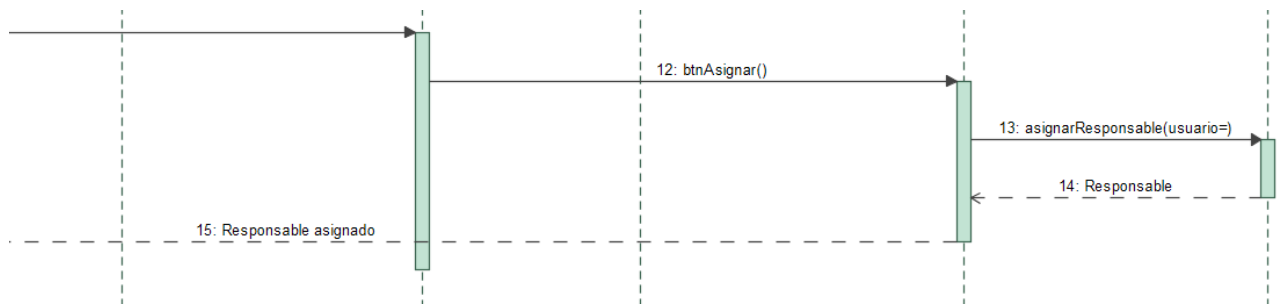


Figura 44 Anexo DS: Asignar responsable a historia de usuario (Parte 3)

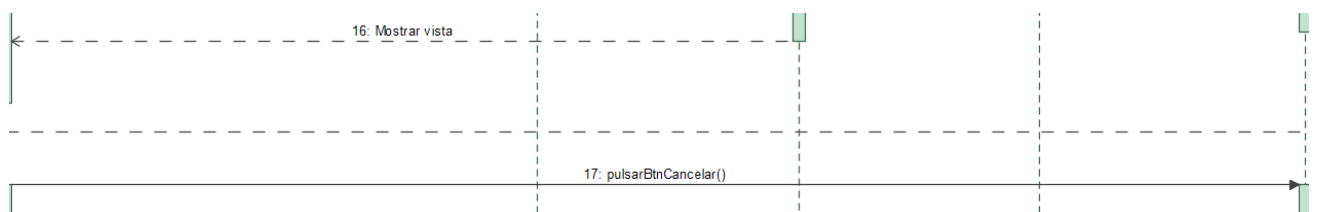


Figura 45 Anexo DS: Asignar responsable a historia de usuario (Parte 4)

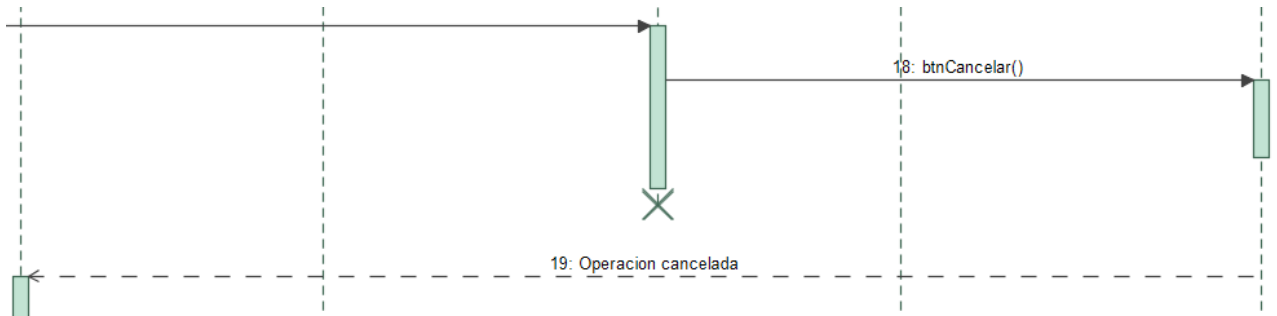


Figura 46 Anexo DS: Asignar responsable a historia de usuario (Parte 5)

Cronograma

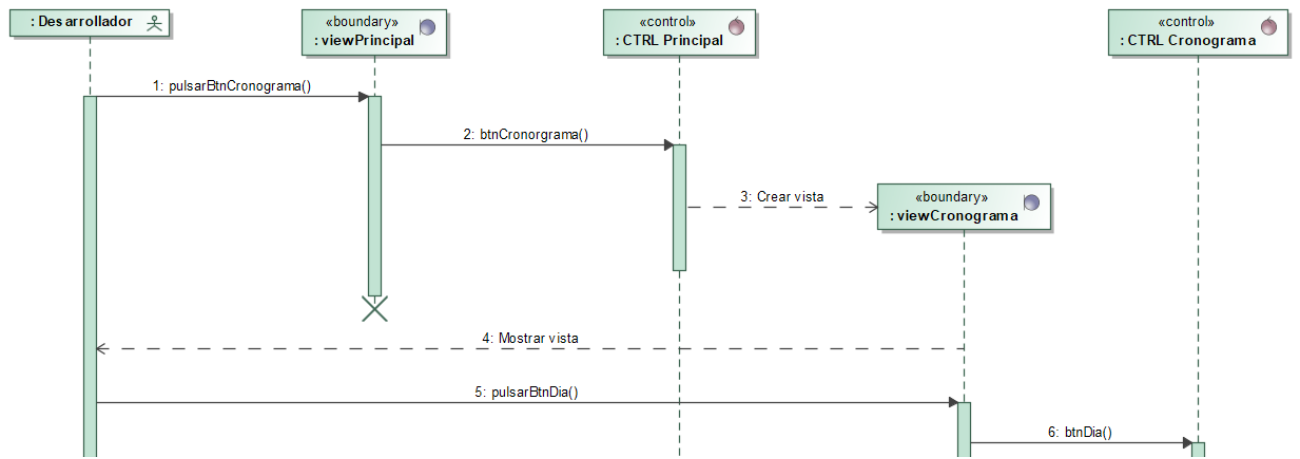


Figura 47 Anexo DS: Cronograma (Parte 1)

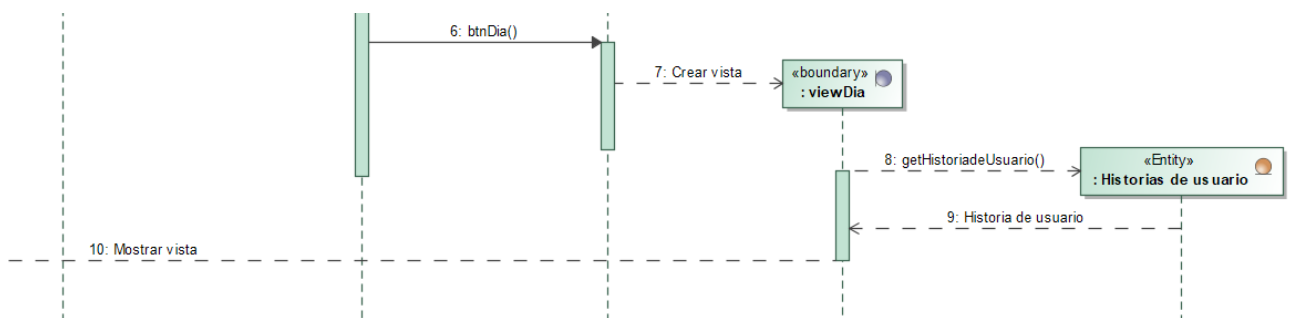


Figura 48 Anexo DS: Cronograma (Parte 2)

Log de historia de usuario

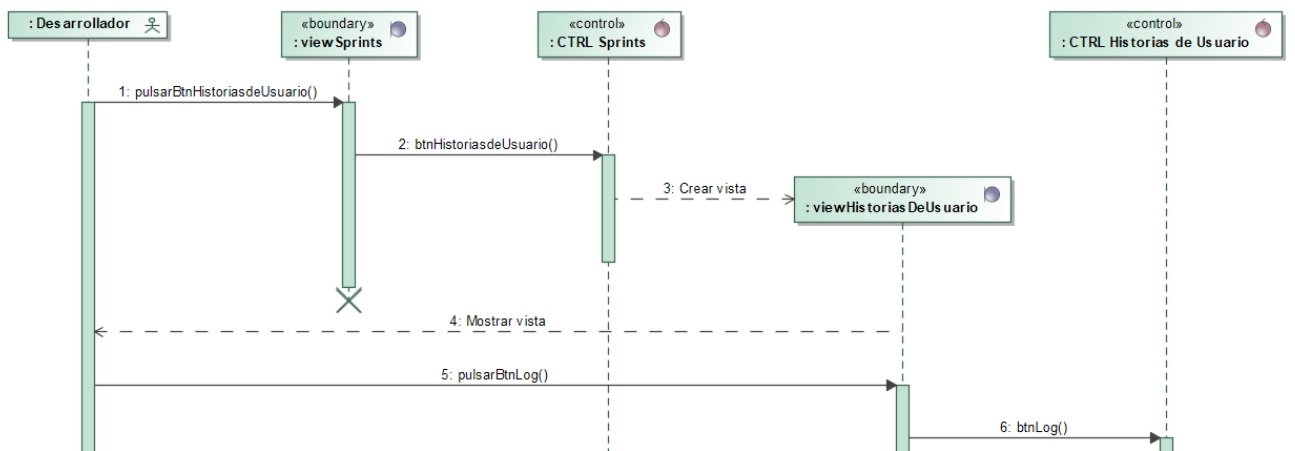


Figura 49 Anexo DS: Log de historia de usuario (Parte 1)

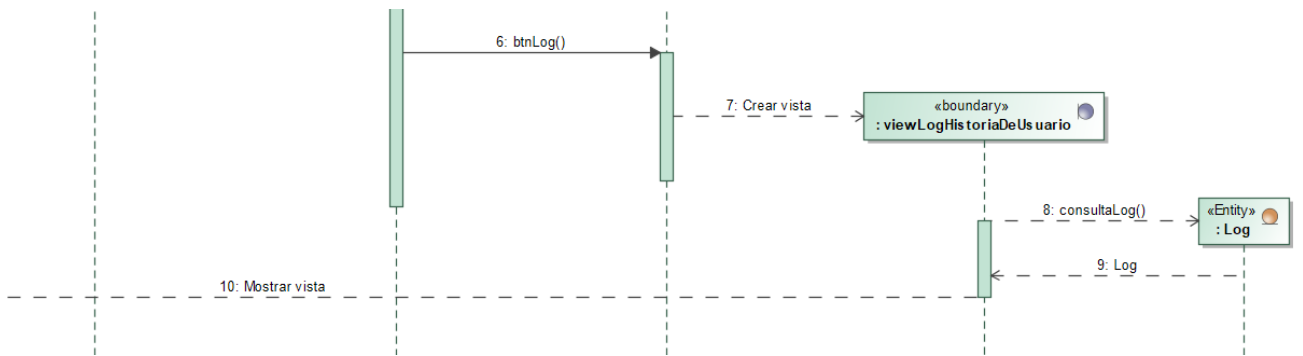


Figura 50 Anexo DS: Log de historia de usuario (Parte 2)

Validar acceso

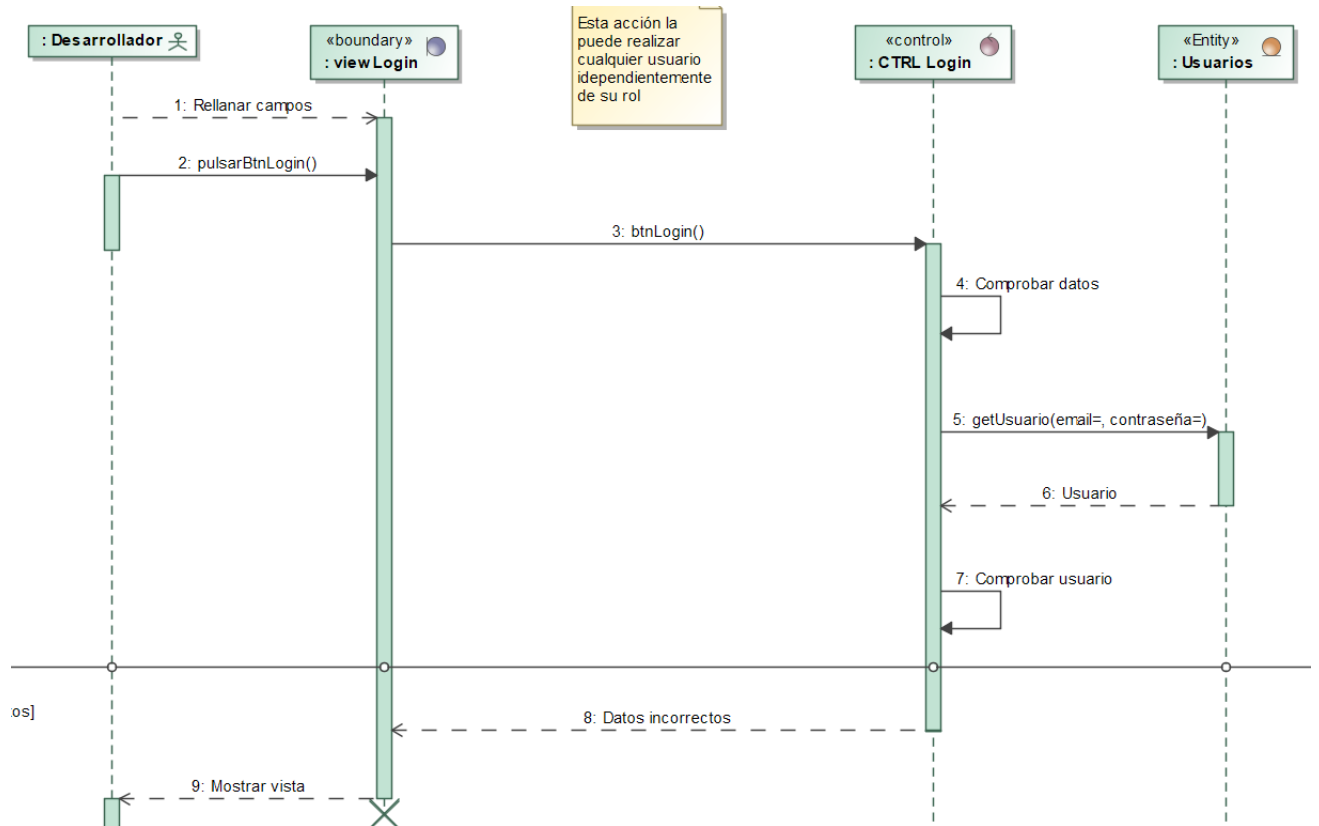


Figura 51 Anexo DS: Validar acceso (Parte 1)

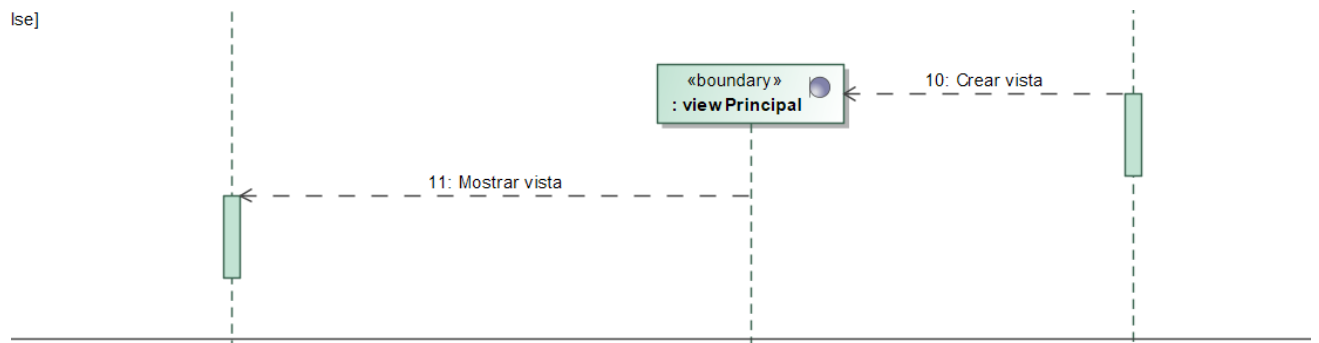


Figura 52 Anexo DS: Validar acceso (Parte 2)

Validar historias de usuario

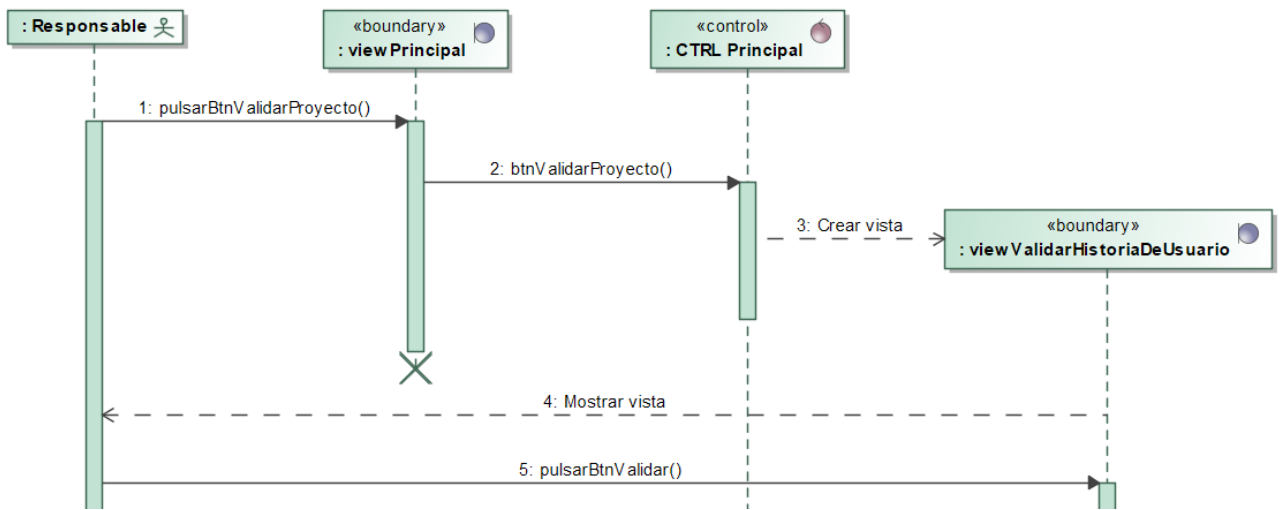


Figura 52 Anexo DS: Validar historia de usuario (Parte 1)

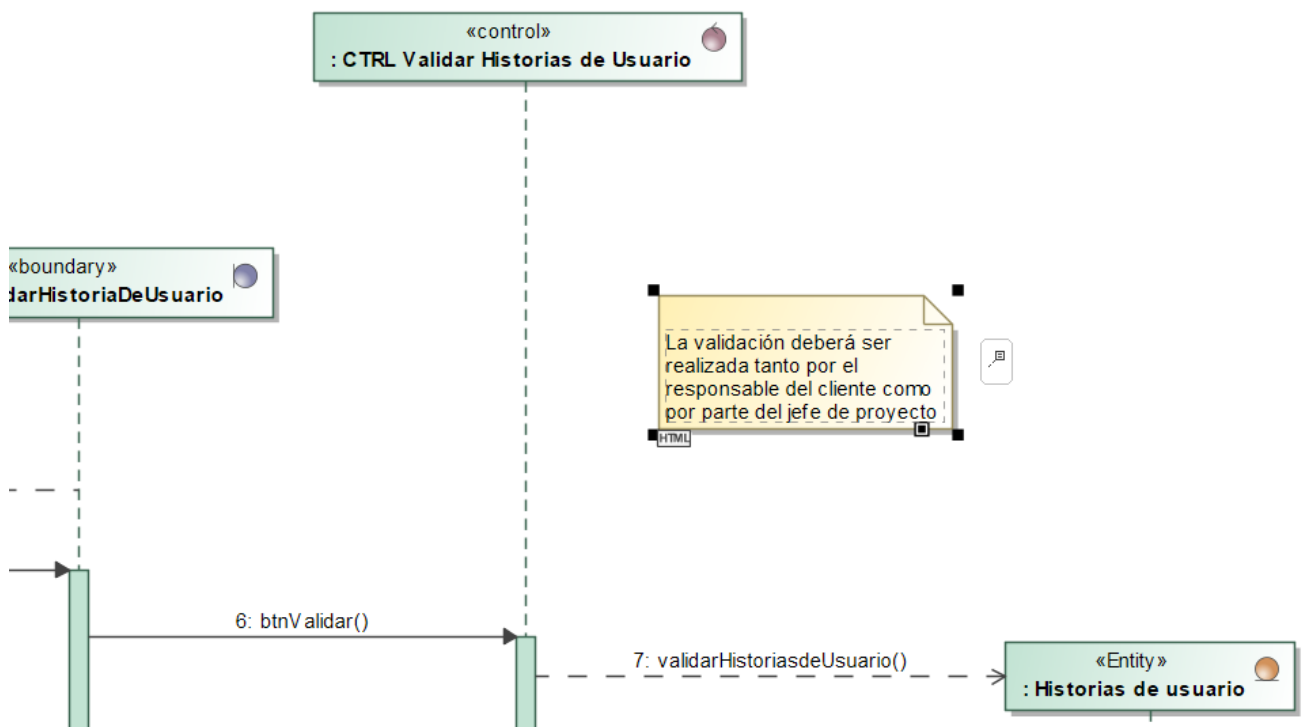


Figura 53 Anexo DS: Validar historia de usuario (Parte 2)



Figura 54 Anexo DS: Validar historia de usuario (Parte 4)



UNIVERSIDAD
DE MÁLAGA

| **uma.es**

E.T.S de Ingeniería Informática
Bulevar Louis Pasteur, 35
Campus de Teatinos
29071 Málaga

E.T.S. DE INGENIERÍA INFORMÁTICA