A Systems Engineering Approach to Curriculum Design

An Engineering Case Study

Carmen García Berdonés, Francisco David Trujillo Aguilera

Departamento Tecnología Electrónica
Universidad de Málaga
Málaga, España (Spain)
{berdones, fdtrujillo}@uma.es

Juan Carlos Tójar Hurtado

Departamento Métodos de Investigación e Innovación Educativa
Universidad de Málaga
Málaga, España (Spain)
jc_tojar@uma.es

Abstract— **This paper deals with the main problems when designing the engineering curriculum taking into account generic competences, and how a Systems Engineering approach can help in overcoming them. Here, the curriculum is considered as a system and a preliminary version of SysML model is proposed for it.**

Keywords-component; Education; EHEA; Bologna Process; Generic Competence; MBSE; SysML

I. INTRODUCTION

New adaptation demands towards the European Higher Education Area (EHEA) [1] suggest two innovative parameters for designing of degrees: European Credit Transfer System, (ECTS) and competences. The first one sets the learning student time as the new standard for measuring degree length. The second one moves the object of the process of learning since to show contents by professor to foster competences by students [2]. Those competences are usually divided into two groups: specific and non-specific ones. The former one depends on the specific degree while the latest one might be acquired by all university students, but they should reach different levels depending on their respective degrees. This paper will be focused on non-specific competences, hereinafter called Generic Competences (GC) although they can be found at literature as transversal, key or transferable competences or skills, depend on features of each kind of GC [3].

The transition from traditional learning to competence based learning is not being an easy process. In fact, a decade past since the process began, there are still some unclear aspects, first of all, the term "competence". A unique meaning cannot find neither in its original area, Human Resources [4], nor in EHEA area, where different formulations of this term are shown in two significant EHEA papers: Tuning project [5] and European Qualifications Framework (EQF) [6]. In EQF framework, the different levels of qualifications must be described in terms of knowledge, skills and competences. Here, competences are related to responsibility and autonomy. However, Tuning states that competences represent a dynamic combination of knowledge, understanding, skills, abilities and attitudes. Anyway, both of them share the so called Learning Outcomes (LO). LO describe what a learner is expected to know, understand and be able to do after successful completion of a process of learning [2]. So, if you work is based on LO, you will can specify well defined evaluation criteria and chose right learning activities for student. However, Tuning meaning instead LO will be used here, because Spanish curriculum competences have been stated with that meaning.

Other non-well defined issue related to design based on competences is the choices which must be done for each competence. For instance, it must be define the expected level of proficiency of student (domain levels) and it must be scheduled a progressive develop along degree by using the proper learning methodologies and assessment tools. Additional previous issue must be considered in the particular case of GC: their usually inaccurate description. The competence "team work" is a good example. When you say "student must be able to work in groups", you are talking about a set of skills (e.g. conflict´s resolution, leadership, oral expression) what can be non-well define for all curriculum design actors (professor, students, employers). European [7-9] and non-European [10-11] authors have addressed these issues. All of them agree that, to improve the design, large competences must be split in more simplex ones and all of them must be related to LO.

As outlined above, curriculum design can be very difficult, because curriculum is a very complex system that involves a lot of competences (requirements to satisfy), a lot of customs (students, employers, and society), a lot of actors (professors, student, and administrative staff) and, maybe, a lot of designers. This paper proposes a Systems Engineering (SE) approach which can help to tackle the main problems related to curriculum design in the engineering area. Special attention will be paid to GC. This paper presents the first steps of a research, carried out in the School of Telecommunications Engineering (ETSIT) at the University of Málaga, focused on enhancing the develop and evaluation of GC included in engineering curriculum.

Next section talks about the role of SE at the system design stage together with SysML™, a language for SE; SE point of view of the curriculum and some examples of this approach are

described in section III. Last sections are dedicated to extract preliminary conclusion from using SysML™ model proposed here and to suggest a way to extend it.

II. SYSTEMS ENGINEERING AND SYSML

A. Basic concepts

As INCOSE (International Council on Systems Engineering) states "SE is an interdisciplinary approach and means to enable the realization of successful systems. It focuses on defining customer needs and required functionality early in the development cycle, documenting requirements, then proceeding with design synthesis and system validation" [12]. Hence, the aim of SE is to take into account every system parameters (e.g. operation, cost, schedule, performance) since the beginning of the process in order to get an optimal final system that complains customer needs.

Model-Based Systems Engineering (MBSE) is the formalized application of modeling to SE. Although a model is a simplification of reality, it is important to model when you need to identify complexity, to increase understanding about the system or to communicate in an unambiguous (or as unambiguous as possible) manner [13]. In spite of MBSE is especially useful for complex systems, nowadays there is not a standard for MBSE. SysML™, an extension of the well-known UML (Unified Modeling Language), is a general purpose graphical modeling language which could become a standard in the near future [14].

B. SysML ™ based methodology for system modeling

At an early design stage, SysML™ based methodology approaches the system modeling by splitting it up into three concurrent sub-models. Elements, relationship and diagrams for each sub-model are briefly described next. It can be found some examples in section III.

- *Use sub-Model* (UM) allows keeping all customer and stakeholder needs into the system model (by using "use case" elements). Examples of use cases, scenarios, can be defined in order to clarify these needs. Inaccurate requirement are usually obtained from this sub-model in first design stages, the so called "natural language" requirements. UM includes also relationships elements. Dependences between use cases can be modeled (by using "include" and "extend" relations) and between system and non-system (actors and context) elements. Use Case Diagrams (UCD) can be drawn to represent different views of UM.

- *Requirement sub-Model* (RM) allows keeping all system requirements expressed in an unambiguous way and information about them (by using "requirement" elements). So, if you want build RM, you will have to re-write carefully "natural language" requirements and to describe a set of parameters for each one. At least, you have to describe the reasons to achieve it-rationale –, the way to test it - verification – and the restrictions to achieve it-constrains-. This work is easier if you decompose complex original requirements ("high level") into new simpler ones

("low level"). Shared "low level" requirements can be discovered besides. Requirements hierarchy is kept into model by using relationship elements ("belong" and "derive" for share elements). Requirement Diagrams (RD) can be draw to represent different views of RM.

- *Context sub-Model* (CM) allows keeping the functional partition of the system in more simplex systems (by using "blocks" elements). Information flows that are interchanged between blocks can be modeled by adding port elements to blocks. Like UM, relationship elements are using to express interactions among system (blocks), context and actors. Block diagram (BD) and Internal Block Diagram (iBD) can be drawn to represent different views of CM.

You need to make sure that all customer needs are right covered by unambiguous requirement, and that there is a system part that is able to satisfy each requirement. Three sub-models concurrent (and coherent) construction helps to get a good design. So, SysML™ lets you model relationship between sub-models elements (e.g. "use case" is "refined by" a "requirement" what is "satisfied by" a "block") and represent them by mixing several sub-model elements in one sub-model diagram.

III. DESIGNING CURRICULUM WITH SYSML

The degree curriculum can be treated as a system, since a system is defined as "a set of functional elements organized to satisfy user´s needs". [15] These needs are requested by customs and stakeholders of the system. So, the degree curriculum can be considered as a system which has to satisfy student formation needs. Academic resources (i.e. professors, and classrooms) and academic activities (i.e. lectures, homework and assessments) will be the elements to be organized. The whole society behaves as the curriculum custom and then, it must define requirements for meeting student formation needs. That is to say, it must define which competences students have to reach in addiction to requirements related with time (ECTS) and cost. Finally, competences will be elicited too from the main stakeholders, social groups (i.e. employers, politicians, academic staff and the students themselves) with a legitimate interest in the system and its final product: the higher education graduate.

This paper focus is not decision criteria about design of based on competences curriculum, but how SE can help to make these decisions. So, next paragraphs describe only a SysML™ based methodology supports to carry out the three actions involved with design of curriculum: to choose competences, to plan activities for their develop and to define criteria for their evaluation [16].

A. Use sub-model to classify candidate competences

From SE view, custom use cases are the engineering curriculum compulsory competences stated by Spanish legislation [17-18]. Stakeholder use cases can be elicited from reports that describe the industrial needs for Engineering. Tuning work [5] or Reflex Project [19] are good examples of employer GC sources. It can also be helpful to look for

competences chosen by other Spanish universities that offer engineering degree, for instance Polytechnic University of Valencia [16] or Polytechnic University of Cataluña [20], as well as non-Spanish universities, like [11]. Finally, it must be considered the opinion of several authors who show a critical view about using employer needs like unique source to define education needs [21-22].

In Fig. 1, UCD is used to show a first competence classification according to competences sources mentioned above. Curriculum must complain Spanish legislation (main use case). Legal competences can be extended with another ones found in other sources. First, it must be identified legal Spanish competences include in Non´-Spanish Universities and industrial reports (several use case has been drawing for keeping this fact). The knowledge about how many competences are shared for industrial stakeholders (report-shared use cases) or for universities (University-shared use cases) can be interesting in order to make the choice.
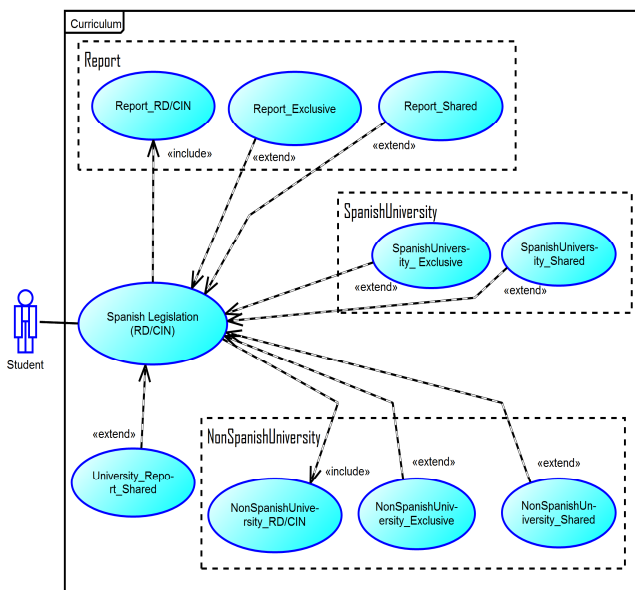

Figure 1. Example of Use Case Diagram: Classification of competence sources

So, UCD, showing relationships between use cases coming from all these different sources, can be used to make a preliminary sorting before choosing candidate competences.

B. Requirement sub-model to clarify and to choose competences

Competences referred above are usually formulated in very general sense, like "natural language requirement". So, you have to re-write chosen use cases to build RM look for more simplex competences ("low level" requirements) and set their hierarchy. An example, based on two Spanish legal generic competences (LGG) from [18], is outlined below. The translation of the original text for LGC 12 and LGC 17 are shown in the next paragraphs.

LG12: "Ability to solve problems with initiative, creativity, making decision, and for communicate and transmit knowledge, skills and abilities, understanding ethical and professional responsibility inherent to telecommunication engineering activity"

LG17: "Ability to work in multidisciplinary team and in multilingual environments, and to communicate, in oral and write way, knowledge, procedures, results and ideas related to telecommunication and Electronic".

RD for the two LGC is shown in Fig. 2. First, more simplex competences inside the original texts have been found, and then, they have been represented like nested requirement using belong relations. This first trial lets you see duplicated competences which can be deleted from model in order to simplify it. But "belong" relation means that each "son" requirement belongs only to its "father" requirement, and for the most of the sons in Fig. 2 probably it won't be true. For instance, LG12_4 (oral and written communication) it is necessary for LG17_1 (teamwork).
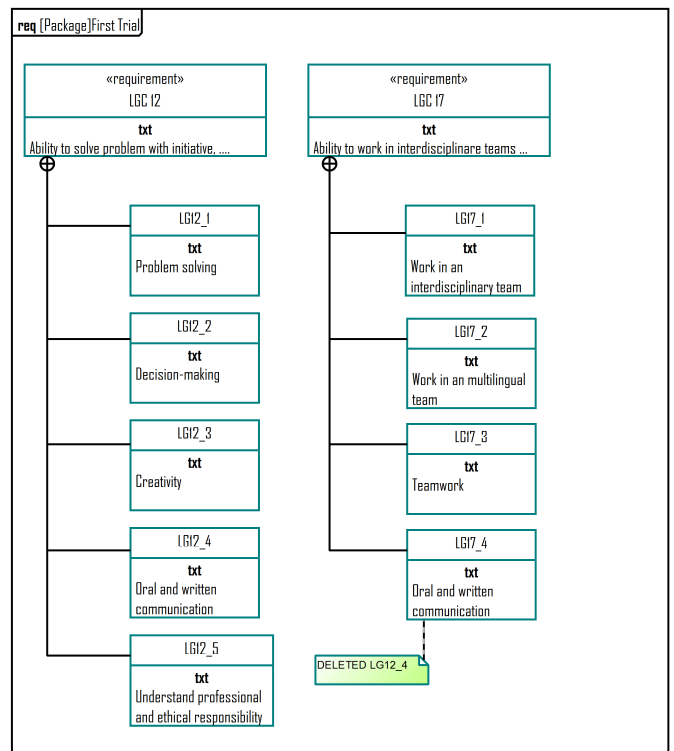

Figure 2. Example of Requirement Diagram: First attemp to address requirement descompose

A second trial RD is shown in Fig. 3. "Belong" relations have been replaced for "derive" relations, which means that a son requirement is necessary to complain more than one father requirement. In this RD you can see new relation between the requirements showed in Fig. 2. Besides, a new competence coming from Tuning project [5] has been included because it has been considered that it is necessary to complain LG17_1.

Finally, it can be said that the relationships between requirements and use cases shown in Fig.1 can be saved in the model by using "refine" relations. As RDs are simplified views

of the whole model, in Fig.3 is shown only a Tuning use case but all elements (requirements, use cases and their relations) are kept in model.
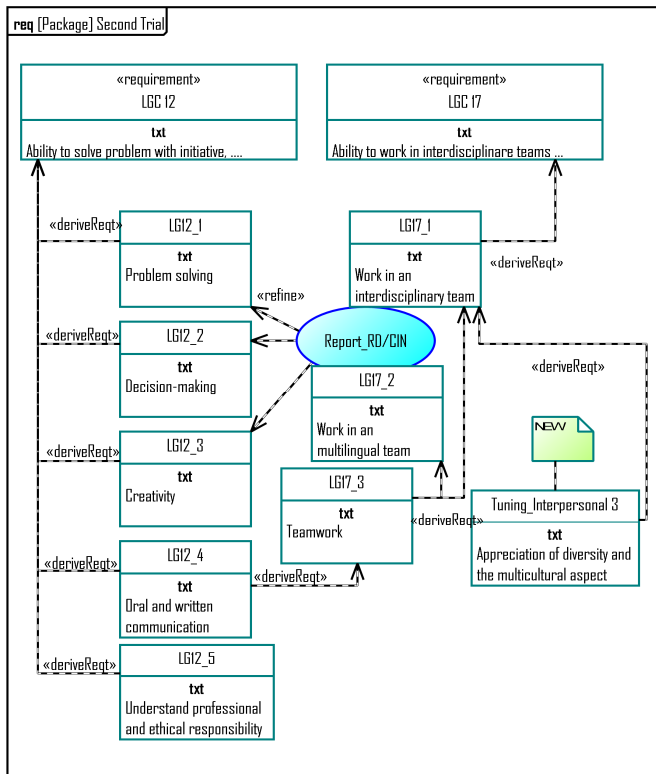

Figure 3.   Example of Requirement Diagram: Second attemp to address requirement descompose

This example has shown how competences decomposition and RD can be useful to identify duplicated information and to find low level competences which must be developed early in the curriculum. Besides, you can look for those low level competences that are essential elements because they hold a lot of high level competences. Finally, stakeholder competences can be included in the model as necessary low level competences or as new competences which will be candidates to be included in final curriculum.

As it was stated in the previous section, it must be defined for each requirement (competence) at least rationale, constrains and assessment. The former one can be filled with its source from UM and with its reason for choosing. But the characteristics of our requirements, especially when we talk about GC, make a non-trivial issue to assign the second and third parameters. We will talk about it in next section and will propose a deeper research as future work in the last section.

C.   Context sub-model to assign competences to curriculum

Curriculum can be divided into functional parts based on several criteria: academic activities (e.g. lecture, homework or assessment) and non-academic activities but related with (e.g. second language course, conference or workshop), timing (e.g. course or semester) or contents (e.g. subjects or modules). From SE point of view, all of these parts are blocks and it can

be defined in CM. A BD example for two Spanish degree curriculum, Electronic systems engineering (SE) and Telecommunication systems engineering (ST), is shown in Fig.4. Curriculum_SE and Curriculum_ST blocks represent each bachelor respectively. The iBD related to former one is represented in Fig. 5.

BD lets define shared and specific components (and interfaces) for each block. A black diamond ended relations means belong relation, and an ended arrow relation means that one block contains, at least, the same elements than other block. For instance, in Fig. 4 is represented that first and second course are common for both bachelor, and that all bachelors must have a set of quality processes.
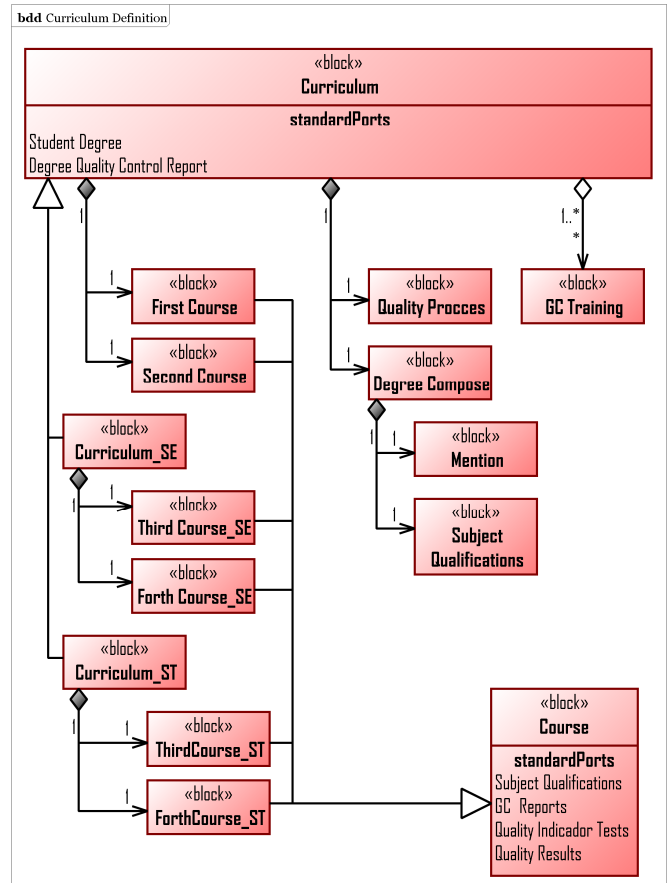

Figure 4.   Example of Block Diagram: Curriculum components

All courses must communicate student results in order to compose final student degree. This communication has been modeled by adding to block course a port for traditional subject qualifications, and another port for results of extra subject activities which could be included in student degree. Typically, some GC's are developed and assessed with these extra subject activities (e.g. second language). These extra subject activities could be also done not in courses but in parallel with them, so a block has been included for them (GC Training), but with white diamond ended relations. That means that this block can be used but its design has been done for other than curriculum

designer (e.g. university second language courses or extra university training on team work).

The port connections and the actors related with each block are represented in iBD. For instance, it can be seen in Fig. 5 that teachers from another institution can be involved with extra subject blocks, or the input and output from quality process.

It must be noted some issues about Fig. 4 y Fig. 5. In one hand, only the port and the block related to Generic Competences have been considered, but we could extend model for all kind of competences. In the other hand, quality processes will be active during all curriculum time, so the relations with all blocks have been modeled, but iBD relation with course blocks has been only shown for draw clarity. For the same reason, only a few of all modeled relations (between all courses blocks and degree compose block and between actors and blocks) are showed.

2 y Fig. 3). All competences chosen for curriculum must be covered at least for a block in a first design iteration. In successive iterations, the curriculum design will be completed by selecting a specific sub block (included in course, CG training or quality process blocks) to satisfy each competence. So, it must be done iBD for each block.

Sub-blocks proposed for one course (FourthCourse_SE) are represented in the iBD shown in Fig. 6. As in previous diagrams, only a few relations have been drawn. You can see four kinds of subject proposed by Spanish legal (CO, TFG, Opt and TE) and an iBD for one of them (SubjectCO). It has been decided to group the subjects according to the compulsory way, but it can choose other better criteria to make an easier assignment. Multiplicity for each block (i.e. how many sub blocks could be belonged to a block) can be seen in the right corner of each block. It must be noticed that some blocks are optional (symbol "*" means zero or more blocks). For instance, activities and assessment of GC are often made into specific competence activities, but blocks for separated CG actions have been foreseen.
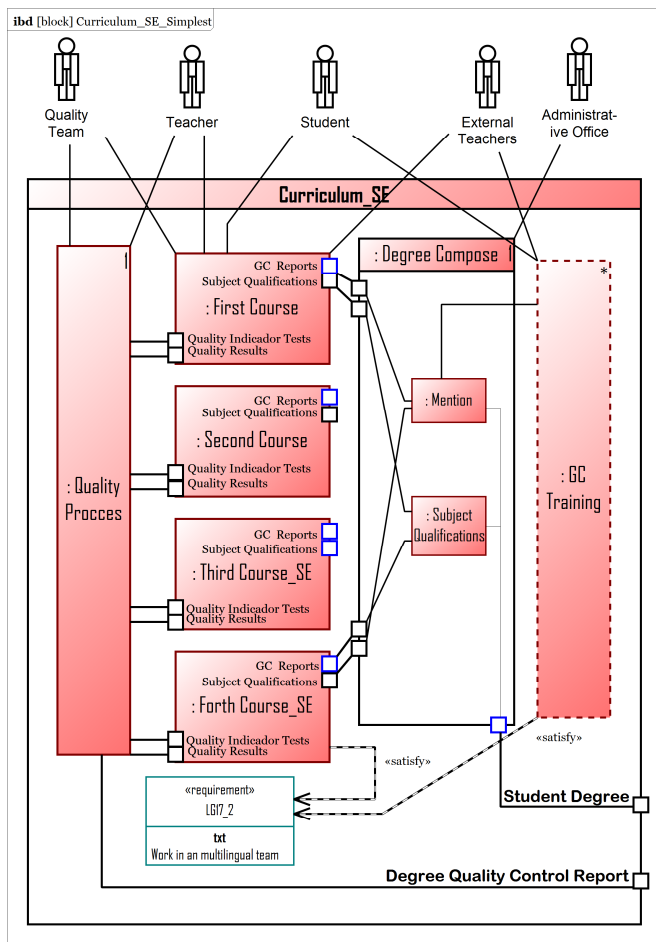

Figure 5. Example of Internal Block Diagram: Curriculum components, requirement and their relationships

When the curriculum architecture has been set, it must be assigned all system requirements to blocks. You can assign a competence to a several blocks using "satisfy" relation. It means that activities to develop and to assessment it must be done in this blocks. In Fig. 5 are shown two "satisfy" relations for Requirement LG17_3 (same element model showed in Fig.
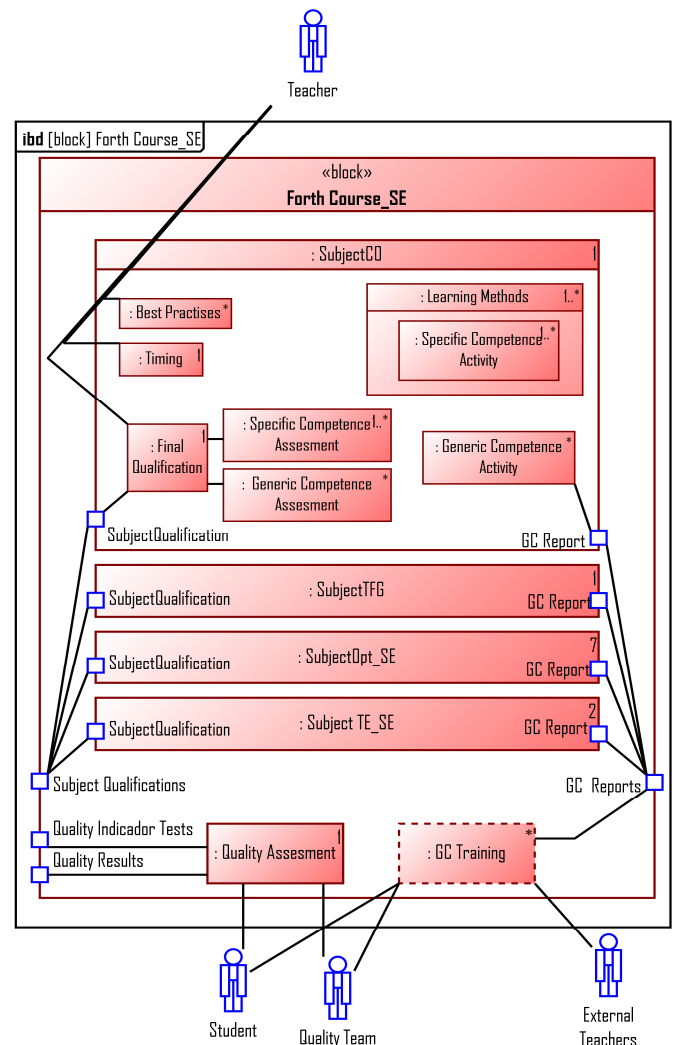

Figure 6. Example of Internal Block Diagram: Course and subject components

In short, it has been outlined two main advantages of using CM. First, it improves communication and coordination between curriculum designer professors, because a lot of characteristics of curriculum can be seen quickly and BD and iBD help them to define common parameters and interfaces for curriculum subject. Second, it lets in an iterative way look for the best blocks to satisfy each requirement, taking into account the desirable order for developing it.

IV. CONCLUSIONS

A MBSE approach to design a curriculum based on competences has been shown. Curriculum has been treated as system and competences as system requirements. So, you can use this approach to set a list of clear system requirements (curriculum competences) and to make possible an unambiguous (or as unambiguous as possible) communication between curriculum designer team. Both actions are specially required when you talk about GC.

The example of SysML™ based methodology presented here has pointed to use UM for preliminary sorting of candidate competences. Then, it has been suggested to build RM to identify duplicated information and to find low level competences which must be developed early in the curriculum. Finally, CM can be used to facilitate the necessary coordination between curriculum designer professors and to look for the best curriculum block to develop each competence. That is, three sub-models use can help to define and to schedule a progressive competence develop along degree.

It is important to note that concurrent construction of the three sub-models can help to get a good design. In fact, when you are thinking about one of them, you are normally reviewing the other two. For instance, the new requirement shown in Fig. 3 was coming from Report_Shared use case (Fig. 1), but when RM was making, we thought that this new requirement was included in other one coming from Spanish_Legislation use case. So, it was necessary to review UM in order to change the new requirement source from Report_Shared use case to Spanish_Legislation use case. The inclusion of Quality blocks in CM was done too in a CM rewrite, when RM model was been thought.

Finally, it must be noticed that when a requirement is assigned to block, characteristics of this block are being constrained to comply the requirement. From curriculum SE point of view, this means that competence characteristics force the curriculum activities. Therefore, a subject professor mustn't plan any learning activities or any evaluations, but only those coming from assigned competences. That is only a different way to say that subject must be competence based.

So, more extensive characterization of competences than the presented here must be done, in order to assign activities to CM blocks. From SE point of view, that means that requirement description has not been complete in this paper, it must be defined requirement assessment and requirement constrains, as we have stated in previous section.

V. FUTURE RESERCH

We propose for future work to build a competence SYSML model (ComM). ComM will extend requirement elements shown in previous section (description and rationale) by means of defining a sort of LO for each domain level of a competence. As it was said in the Introduction of this paper, a LO allows you to specify well defined evaluation criteria and to set which activities can be useful for student to get it. Time spent or necessary resources to do these activities become competence constrains. Therefore, competences can be assigned to block in Fig. 6 in design preliminary iterations, as it has been suggest in this paper. But competences must be replaced for their LO in successive iterations. In this way, we can get the final curriculum because the activities and assessment of each block in Fig. 6 will be the activities and assessment that were set for the LO assigned to this block.

After all the aforesaid, we propose the next future step to design a specific curriculum. First, ComM will be proposed. A parametric diagram, typical SysML™ diagram not shown here, will be used to take into account requirement constrains to get the final whole model. Second, in order to help to design the curriculum, research will be focus on found a right sort of LO for each domain level of chosen competences. So, literature will be deeply revised to look for different sorts of LO for generic competence. [9, 10] are two examples of this kind of literature. Finally, we will look for a set of right activities and assessment tools for these LO by reviewing the work from authors like [23] or [24]. Both explain different learning methodologies and their relations with generic competences.

REFERENCES

[1] The European Higher Education Area, Bologna Declaration, 1999.

[2] European Union, "ECTS user guide". Available, 2012: http://ec.europa.eu/education/lifelong-learning-policy/doc/ects/guide_en.pdf

[3] A. Kelly, "The evolution of key skills: Towards a tawney paradigm", Journal of Vocational Education & Training, vol. 53(1), pp. 21-36, 2001.

[4] F.D. Le Deist and J. Winterton, "What is competence?", Human Resource Development International, vol. 8(1), pp. 27-46, 2005.

[5] J. González and R. Wagenaar, "Universidad de Deusto, Universiteit Groningen and Comisión de las Comunidades Europeas", Tuning Educational Structures in Europe: Informe Final, 2004-2006.

[6] European Union, "The European Qualifications Framework for Lifelong Learning (EQF)". Available, 2012: http://ec.europa.eu/education/pub/pdf/general/eqf/broch_en.pdf

[7] M. García García, M. Terrón López, and Y. Blanco Archilla, "Desarrollo de Recursos Docentes para la Evaluación de Competencias Genéricas", ReVisión, vol. 3(2), pp. 265-272, 2010.

[8] M. Pallisera i Díaz, J. Fullana Noell, A. Planas Lladó, and A. Valle Gómez, "La adaptación al espacio europeo de educación superior en España. Los cambios/retos que implica la enseñanza basada en competencias y orientaciones para responder a ellos", Revista Iberoamericana de Educación, vol. 52(4), pp. 2-13, 2010.

[9] A. Villa Sánchez and M. Poblete Ruiz, Aprendizaje basado en competencias: una propuesta para la evaluación de las competencias genéricas (3ª ed.), Bilbao: Ediciones Mensajero, 2010.

[10] M. Besterfield-Sacre, L.J. Shuman, H. Wolfe, C.J. Atman, J. McGourty, R.L. Miller, and G.M. Rogers, "Defining the outcomes: A framework for EC-2000", IEEE Transactions on Education, vol. 43(2), pp.100-110, 2000.

[11] E.F. Crawley, J. Malmqvist, S. Ostlund, and D. Brodeur, Rethinking Engineering Education: The CDIO Approach., New York: Springer Verlang, 2007.

[12] C. Haskins, "SE Handbook Working Group and International Council on Systems Engineering", Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities, 2011.

[13] J. Holt and S. Perry, SysML for Systems Engineering, United Kingdom: The Institution of Engineering and Technology, 2007.

[14] International Council on Systems Engineering, "SE Vision 2020". Available, 2012:
http://www.incose.org/ProductsPubs/products/sevision2020.aspx

[15] IEEE, P1220 Standard for Application and Management of the Systems Engineering Process, 1994.

[16] M. Edwards, L.M. Sanchez-Ruiz, and C. Sanchez-Diaz, "Achieving competence-based curriculum in engineering education in Spain", Proceedings of the IEEE, vol. 97(10), pp. 1727-1736, 2009.

[17] Boletín Oficial del Estado, Real Decreto 1393/2007, vol. 260, pp. 44037-44048, 2007.

[18] Boletín Oficial del Estado, Orden CIN/352/2009, vol. 44, pp. 18150-18156, 2009.

[19] J. Allen and R. van der Velden, The Flexible Professional in the Knowledge Society, Dordrecht: Springer Netherlands, 2011.

[20] R. Bragos, "Proceso de inversión de competencias genéricas en los nuevos planes de estudios de grado de la ETSETB de acuerdo con el modelo CDIO", Congreso de Tecnologías Aplicadas a la Enseñanza de la Electrónica, pp. 1-9, Madrid, 2010.

[21] J. Walther and D. Radcliffe, "The competence dilemma in engineering education: Moving beyond simple graduate attribute mapping", . Australasian Journal of Engineering Education, vol. 13(1), pp. 41-51, 2007.

[22] E. J. Diez Gutiérrez, "Bolonia: ¿racionalización de la educación o modelo neoliberal?", Aula de Innovación Educativa, vol. 186, pp. 70-73, 2009.

[23] M. De Miguel Díaz, Metodologías de enseñanza y aprendizaje para el desarrollo de competencias: orientaciones para el profesorado universitario ante el Espacio Europeo de Educación Superior (1ª ed.) Madrid: Alianza, 2006.

[24] R.M. Felder and R. Brent, "Designing and teaching courses to satisfy the ABET engineering criteria", Journal of Engineering Education, vol. 92(1), pp. 7-25, 2003.