

Desenvolupament d'una aplicació web per al micro-aprenentatge (μ Learning)

Elliot Ribas Garcia

Resum– Davant de l'ús creixent de tecnologies a les aules, l'ensenyament es troba en la necessitat d'evolucionar amb l'ajut de noves eines i metodologies que s'adaptin al context actual. En aquest projecte s'ha desenvolupat μ Learning, una aplicació web per al micro-aprenentatge a partir de la resolució de qüestionaris sobre els continguts de les assignatures d'un pla d'estudis universitari. Amb el desenvolupament d'aquest projecte, doncs, s'ha obtingut una aplicació funcional que es presenta com una eina docent innovadora, complementària a les classes presencials. μ Learning ofereix als professors la possibilitat de validar i avaluar l'assoliment dels coneixements explicats a les seves assignatures, i als alumnes una manera de seguir el transcurs d'una assignatura i millorar el seu aprenentatge més enllà de la universitat.

Paraules clau– Desenvolupament Web, Aplicació Web, Disseny Responsiu, Java, Spring Framework, Thymeleaf, Ludificació, Innovació Docent.

Abstract– Due to the increasing use of technologies in classrooms, teaching is in the need to evolve with the help of new tools and methodologies that can adapt to the current context. In this project, μ Learning has been developed, a web application for the micro-learning based on the resolution of quizzes about the contents of the subjects in the syllabus of a university course. Thus, with the development of this project, a functional application has been obtained, presented as being an innovative teaching tool, complementary to face-to-face classes. μ Learning offers both teachers the possibility to validate and evaluate the knowledge achieved at their subjects, and students a way to follow the course of a subject and improve their learning outside the university.

Keywords– Web Development, Web Application, Responsive Design, Java, Spring Framework, Thymeleaf, Gamification, Teaching Innovation.

1 INTRODUCCIÓ

EN els darrers anys, és un fet que les tecnologies han anat prenent gran presència a les aules. Els professors fan ús de projectors per mostrar els recursos teòrics de l'assignatura que imparteixen, i en alguns casos pissarres digitals interactives, on hi fan aclariments i exercicis. Els alumnes, d'altra banda, relleven el paper i llapis pels documents digitals i dispositius com tauletes, portàtils o *smartphones*.

Davant d'aquest nou context, les eines d'ensenyament no poden romandre estàtiques i tradicionals. Per això, sorgeixen

noves tècniques docents com la *ludificació*. Aquestes resulten molt enriquidores ja que en el cas de la ludificació es reforça la idea de que s'aprèn fent (*Learn by doing*) i que l'aprenentatge ha de ser quelcom progressiu, on s'utilitzin coneixements anteriors per assimilar-ne de nous, de manera que la motivació i la retroalimentació entre professors i alumnes siguin les bases de l'ensenyament [1]. Aquest Treball de Final de Grau (TFG) consisteix en el desenvolupament d'una aplicació web per al micro-aprenentatge, basant-se en la ludificació com a tècnica d'innovació docent. Així doncs, aquesta memòria presenta μ Learning, una aplicació web a utilitzar per part de la docència com a complement a les classes magistrals de les diferents assignatures d'un pla d'estudis universitari.

L'aplicació consisteix en una sèrie de qüestionaris creats pels professors sobre els conceptes teòrics, o també pràctics, que s'hagin ensenyat a les darreres sessions d'una assignatura. Quan els qüestionaris són publicats pels professors, els alumnes els poden respondre, rebent un *fe-*

- E-mail de contacte: elliott.ribas@e-campus.uab.cat
- Menció realitzada: Tecnologies de la Informació
- Treball tutoritzat per: Joan Bartrina Rapesta (Departament d'Enginyeria de la Informació i les Comunicacions)
- Curs 2018/19

edback sobre la resposta a cada pregunta un cop les han respost totes.

El fet que μ Learning sigui una aplicació web facilita que els estudiants hi puguin accedir des de qualsevol dispositiu en qualsevol moment, potenciant així el seguiment de les assignatures més enllà de les aules. Així doncs, l'objectiu final d'aquest projecte és oferir als professors una manera de validar que els alumnes hagin assolit els coneixements explicats a les sessions de les assignatures llençant-los qüestionaris periòdicament. Per una altra banda, els estudiants disposen de la possibilitat de seguir el transcurs d'una assignatura responnent a preguntes que els hi seran útils per millorar l'assoliment dels coneixements.

La resta del document està estructurat de la següent manera. La Secció 2 analitza l'estat de l'art. La Secció 3 presenta els objectius del treball i la Secció 4 la metodologia i planificació seguida durant el seu desenvolupament. Seguidament, es detallen els aspectes tècnics del desenvolupament a la Secció 5, des de la especificació de requisits fins a la implementació web. Finalment, a la Secció 6 es fa un anàlisi dels resultats obtinguts, per realitzar una posterior valoració i tancar amb les conclusions del projecte a la Secció 7.

2 ESTAT DE L'ART

Abans d'iniciar aquest projecte es va realitzar un anàlisi de l'estat de l'art, per tal de valorar-lo i aportar un nou enfocament. En aquest anàlisi, s'ha trobat *Kahoot!* com l'eina més significativa i amb més presència a les aules actualment [2]. Aquesta aplicació permet als professors crear ràpidament un qüestionari de preguntes multi-resposta. Els alumnes s'hi poden unir mitjançant un codi QR o enllaç web i respondre el qüestionari, amb un temps limitat, quan el professor iniciï la partida.

En l'àmbit de la Universitat Autònoma de Barcelona (UAB) es troben dues plataformes que impulsen l'aprenentatge i la motivació dels alumnes, respectivament. La primera és la plataforma Moodle del Campus Virtual de la Universitat, que permet al professorat crear qüestionaris complexos per avaluar als alumnes durant l'avaluació continuada d'una assignatura. La segona és el Top Enginyeria [3], desenvolupada pel Departament d'Enginyeria de la Informació i les Comunicacions. Aquesta és una plataforma purament de ludificació però d'un caràcter més passiu que les mencionades anteriorment. Ofereix als alumnes la possibilitat de guanyar mèrits realitzant activitats acadèmiques addicionals a banda de les regulars dins d'una assignatura, o bé obtenint bons resultats a les proves individuals o pràctiques. Els mèrits posicionen als alumnes en un rànquing, de tal manera que la plataforma promou una competitivitat sana entre els alumnes per superar-se entre ells alhora que milloren individualment.

μ Learning es presenta com un híbrid format per les característiques principals de les tres plataformes introduïdes anteriorment. Això vol dir que permet limitar el temps de resposta dels qüestionaris dinamitzant la seva resolució, un dels punts clau de *Kahoot!*, que també té el caràcter avaluador pels professors i enriquidor pels alumnes, com els qüestionaris del Campus Virtual, i que motiva un vincle entre els estudiants i les assignatures que cursen més enllà de les activitats regulars, tal i com fa el Top Enginyeria.

3 OBJECTIUS

Els objectius que aborda aquest TFG i que s'esperen aconseguir un cop μ Learning s'hagi desenvolupat completament, de major a menor prioritat, són:

- O-1** Oferir al professorat una eina d'ensenyament complementària a les classes magistrals.
- O-2** Oferir a l'alumnat una eina amb la que millorar l'assoliment dels coneixements.
- O-3** Incrementar la motivació dels estudiants al seguiment de les assignatures que cursen.
- O-4** Millorar els resultats acadèmics dels estudiants que cursin les assignatures que utilitzin μ Learning.
- O-5** Millorar la valoració per part dels alumnes a les enquestes de professorat i assignatures.

Els tres primers objectius són els més essencials, ja que són la conseqüència directa del desenvolupament de μ Learning. L'objectiu O-1 destaca la innovació docent que suposa l'aplicació, mentre que O-2 i O-3 són la conseqüència directa que s'espera al aplicar μ Learning a la docència.

En un pla més secundari però també interessant d'aconseguir, es trobarien els dos darrers objectius, l'assoliment dels quals ve determinat per la realització dels anteriors. Així doncs, s'espera que l'ús de l'aplicació porti als estudiants a millorar els seus resultats acadèmics (Objectiu O-4) i conseqüentment això es vegi reflectit positivament en la seva valoració a les enquestes sobre les assignatures i el professorat (Objectiu O-5).

4 METODOLOGIA I PLANIFICACIÓ

El desenvolupament del software d'aquest treball s'ha realitzat de manera *àgil*. Aquest enfocament segueix un model iteratiu i incremental, amb cicles curts de retroalimentació per facilitar l'adaptació a canvis, i l'objectiu d'obtenir un software de qualitat, entre d'altres principis [4]. Concretament, s'ha seguit la metodologia *Scrum* [5]. El motiu pel qual s'ha escollit aquesta metodologia àgil ha estat l'experiència prèvia, obtinguda de la realització d'altres projectes durant el grau i al lloc de treball.

Les característiques de *Scrum* aplicades en aquest projecte han estat la organització dels paquets de treball basats en requisits i la revisió periòdica de la realització dels mateixos. Així, s'ha adoptat una flexibilitat i reacció àgil al canvi, necessàries per afrontar tant la inestabilitat que presentava el projecte, al tractar-se del desenvolupament d'una aplicació web completament des de zero amb requisits canviants, com la limitació temporal que suposa el fet d'estar emmarcat dins d'un TFG.

Els paquets de treball del projecte han sorgit dels requisits definits, que es veuran a la següent secció, on correspon cada paquet a un o varis requisits. Tots aquests paquets (d'ara endavant *tiquets*), s'han agrupat dins l'anomenat *Product Backlog*. Després, a l'inici de cada iteració (*Sprint*) s'han anat escollint petits conjunts de tiquets per passar al *Sprint Backlog* segons la seva durada i prioritat, des d'on partia el seu cicle de vida, que pot canviar entre els estats *In Progress*, *Blocked* o *Done Current Sprint*. Al final de

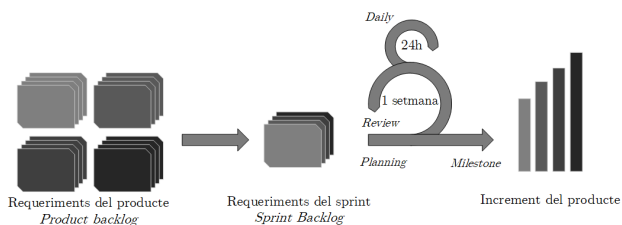


Fig. 1: Workflow de Scrum

cada Sprint havent passat el temps d'una setmana, definit així per tal de poder assolir progressos significatius, s'han revisat els tiquets actius i s'ha valorat si s'havien completat i podien passar a l'estat *Done* o si, contràriament, no s'havien pogut completar a temps i calia retornar-los al *Sprint Backlog*. La gestió dels tiquets s'ha realitzat mitjançant un tauler a la plataforma *Trello* [6], seleccionada per la seva simplicitat d'ús i per les característiques que ofereix, com etiquetar i classificar els tiquets, assignar-los *deadlines* per adaptar-los a la planificació, o enllaçar el repositori GitHub del projecte al tauler.

Per les revisions periòdiques mencionades s'han emprat les reunions de Scrum. Aquestes reunions han estat completament individuals i s'han utilitzat amb dues finalitats. La primera, a mode de revisió diària sobre els avenços fets el darrer dia treballat i el que es pretenia fer aquell dia, en el que s'anomena un *Daily Meeting*, i la segona, a mode d'avaluació global de la darrera iteració i planificació de la vinent, en les reunions de *Sprint Review* i *Sprint Planning*, respectivament. A la Figura 1 es mostra un diagrama que representa aquest flux de treball (*workflow*) complet.

La planificació del treball s'ha organitzat en fases on cadascuna ha inclòs activitats amb una durada assignada, la suma total de les quals representa el temps disponible per la realització d'aquest TFG, 300 hores. Aquestes hores s'han distribuït uniformement al llarg dels 18 Sprints del projecte, corresponents a les 18 setmanes compreses entre les dates de l'11 de febrer i el 16 de juny, aproximadament. Per tant, a cada Sprint s'han realitzat entre 16 i 17 hores de treball, tenint en compte els 7 dies de la setmana com a dies laborables. Ara bé, dins del total d'hores disponibles, el temps que s'ha estimat destinat a desenvolupament ha estat de 260 hores, mentre que les 40 hores restants s'han estimat per la realització de tasques de documentació de seguiment. De totes maneres, finalment el temps total real de desenvolupament ha estat de 239 hores i el dedicat a documentació s'ha allargat fins a les 61 hores.

La Taula 1 desglossa la planificació completa, indicant el temps planificat per cada activitat i el temps real invertit en la seva realització. La taula també inclou els quatre *milestones* que s'han definit, els quals representen els punts de control on s'ha assolit una part important del desenvolupament del projecte, com el disseny, les funcionalitats crítiques o el lliurament de documentació. Sobre l'activitat de desenvolupament de la app, aquesta s'ha planificat dividint l'aplicació en dues bandes, la de professors i la d'alumnes. Això ha permès la realització de tasques de les dues bandes de manera concurrent, les quals s'han fusionat entre elles en determinades ocasions degut a dependències d'implementació. Tot i així, algunes de les tasques planificades no s'han pogut dur a terme finalment per falta de recursos temporals.

TAULA 1: DISTRIBUCIÓ TEMPORAL DE LES FASES, ACTIVITATS I MILESTONES DEL PROJECTE

Fases i activitats	Temps estimat	Temps real
Inici	10 h	10 h
Definició d'objectius	4 h	4 h
Captació de requisits	6 h	6 h
Anàlisi	6 h	6 h
Estat de l'art	4 h	4 h
Elecció de metodologia	2 h	2 h
Disseny	18 h	18 h
Diagrama casos d'ús	2 h	2 h
Diagrama d'arquitectura	4 h	4 h
Diagrama EER de la BD	12 h	12 h
Milestone: Aplicació dissenyada (Sprint 3)		
Implementació	190 h	196 h
Implementació de la BD	2 h	2 h
Creació de l'entorn	8 h	2 h
Desenvolupament de la app	180 h	192 h
Milestone: Aplicació finalitzada (Sprint 15)		
Proves	21 h	6 h
Casos d'ús d'usuari	12 h	6 h
Casos d'ús del sistema	9 h	-
Milestone: Aplicació testada (Sprint 17)		
Documentació i tancament	15 h	3 h
Manual d'usuari	6 h	-
Manual de referència	6 h	-
Milestone: Documentació finalitzada (Sprint 18)		
Conclusions i línies de futur	3 h	3 h

Les causes sobre aquest aspecte i altres desajustaments respecte la planificació original seran detallades en seccions posteriors.

5 DESENVOLUPAMENT

Aquesta secció divideix les fases que conformen el desenvolupament en diverses subseccions. A la subsecció 5.1 es llisten els requisits de l'aplicació definits durant l'activitat de captació. Tot seguit, a la subsecció 5.2 es mostra l'arquitectura dels components que formen l'aplicació, per després poder passar a definir el disseny de la base de dades (BD), a la subsecció 5.3. Per acabar, a les subseccions 5.4 i 5.5 es comenten els aspectes més rellevants del desenvolupament de µLearning i s'explica a alt nivell com s'han implementat el Back End i el Front End de l'aplicació, respectivament.

5.1 Requisites

Durant l'activitat de captació de requisits es van definir un conjunt de requisits funcionals i no funcionals que han condicionat posteriorment el desenvolupament. Els requisits funcionals tenen un risc d'assoliment associat indicant el seu impacte sobre el projecte, on el de grau Alt pot resultar bloquejant per la consecució del projecte i el Mitjà o Baix pot ser solucionat amb mesures de contingència. Per aquests darrers graus s'ha adoptat la mesura d'aplaçar la implementació del requisit i realitzar el següent més prioritari en la planificació, de manera temporal o permanent respectivament. A continuació es llisten els requisits funcionals:

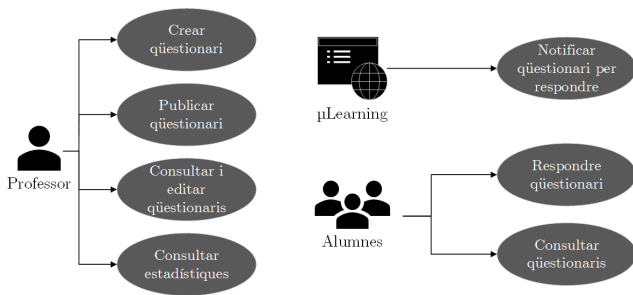


Fig. 2: Diagrama de casos d'ús de μ Learning

- RF-01** Els professors han de poder crear qüestionaris sobre un tema i subtema concrets dins la seva assignatura en un grau. Risc: Alt
- RF-02** Els professors han de poder fixar la data de tancament d'un qüestionari. Risc: Mitjà
- RF-03** Els professors han de poder habilitar un qüestionari que han creat anteriorment per a que pugui ser respost pels alumnes. Risc: Mitjà
- RF-04** Els professors han de poder definir la dificultat d'un qüestionari. Risc: Baix
- RF-05** Els professors han de poder afegir, a part de text si s'escau, imatges tant als enunciats com a les respostes de les preguntes dels qüestionaris. Risc: Mitjà
- RF-06** Els usuaris, professors i alumnes, han de poder consultar a quina assignatura pertany cada qüestionari. Risc: Baix
- RF-07** Els professors han de poder limitar el temps de resposta per cada qüestionari des que els usuaris hi accedeixen per primer cop. Risc: Alt
- RF-08** Els alumnes han de poder respondre qüestionaris amb preguntes tipus test de 4 opcions de resposta múltiple. Risc: Alt
- RF-09** El sistema ha de notificar als alumnes si tenen qüestionaris per respondre, tant recents com antics. Risc: Mitjà
- RF-10** Els professors han de poder consultar estadístiques sobre els qüestionari que han creat, com la quantitat d'alumnes que han respost, el temps mitjà de resposta, els encerts/errades mitjans, etc. Risc: Baix
- RF-11** El sistema ha de mostrar als alumnes un *feedback* per cada pregunta, si n'hi ha, un cop han respost un qüestionari. Risc: Baix
- RF-12** Els usuaris han de poder diferenciar entre estats dels qüestionaris. Els alumnes els han de poder veure entre els estats disponible, aprovat o suspès/expirat/tancat, mentre que els professors els han de poder veure en estat publicat o ocult. Risc: Mitjà

A partir dels requisits funcionals s'ha extret un diagrama de casos d'ús (veure Figura 2), on els actors i els casos d'ús que duen a terme corresponen a les definicions dels requisits d'aquest tipus. Tot seguit, es llisten els requisits de tipus no funcional:

- RNF-01** La implementació de l'aplicació ha de seguir el patró Model-Vista-Controlador.
- RNF-02** El Back End de l'aplicació ha d'estar implementat amb Java, concretament amb el *framework* Spring Boot.
- RNF-03** El servidor web on s'executarà l'aplicació ha de ser un Apache Tomcat 9, integrat dins de Spring Boot.
- RNF-04** La base de dades ha de ser relacional, utilitzant el sistema gestor MySQL.
- RNF-05** El Front End de l'aplicació ha d'utilitzar els llenguatges HTML (amb Thymeleaf), JavaScript (amb jQuery), i SCSS i Bootstrap.
- RNF-06** El Front End de l'aplicació ha de ser *responsive*, adaptant-se tant a dispositius mòbils com a monitors de sobretaula i portàtils.
- RNF-07** L'aplicació s'ha de mostrar en català, però ha d'implementar la possibilitat d'extensió al castellà i anglès.

En seccions posteriors s'especificarà l'acompliment de cadascun dels requisits definits, funcionals i no funcionals, mitjançant la notació (RF-*) i (RNF-*), respectivament.

5.2 Arquitectura i tecnologies

Com s'ha mencionat anteriorment, l'arquitectura de l'aplicació s'ha dividit en dues bandes clarament diferenciades, la de professors i la d'alumnes. Des del punt de vista del desenvolupament, però, l'aplicació ha estat tractada com un mateix projecte indivisible, i per tant la BD s'ha compartit per les dues bandes. Ara bé, per molt que el projecte s'hagi tractat com un de sol, compartint recursos i tecnologies emprades per la seva implementació, els mòduls que el componen es troben força desacoblats entre sí depenent de si la lògica que implementen està orientada a professors o a alumnes. La distribució d'aquests mòduls segueix el patró Model-Vista-Controlador (MVC) de manera estricta (RNF-01), el que implica que la Vista només es comunica amb el Controlador i aquest amb el Model. Amb tot això, l'arquitectura que s'ajusta a l'enfocament descrit, i la que s'ha implementat finalment, és la que apareix a la Figura 3.

Pel que fa a les tecnologies, el Back End de l'aplicació s'implementa en Java, llenguatge orientat a objectes completament independent de la plataforma on s'executi gràcies a la *Java Virtual Machine*. Concretament, s'ha treballat amb Spring Boot [7], una distribució del *framework* Spring (RNF-02) que integra un conjunt de característiques essencials pel desenvolupament web classificades per mòduls dins del *framework*. Els mòduls utilitzats explícitament han estat Spring MVC, per seguir el patró MVC, Spring Session, per la gestió de sessions d'usuari, Spring Security, pel control d'accés i protecció contra atacs a la seguretat

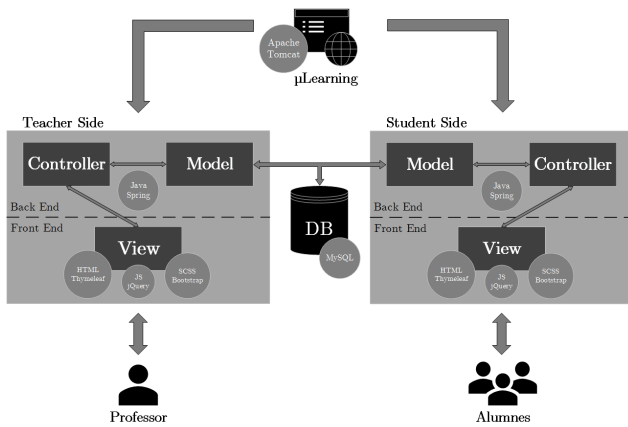


Fig. 3: Arquitectura de μ Learning

de l'aplicació, i Spring Data, per l'accés a dades i la seva persistència tant a BD com al model de l'aplicació. Spring Boot realitza una autoconfiguració d'aquests mòduls per facilitar la seva integració, però també permet configurar-los mitjançant classes Java i paràmetres en un fitxer de tipus *properties*, que pot ser flexible depenent de l'entorn on es trobi l'aplicació (desenvolupament, test, o producció) i més segur per ocultar configuracions sensibles, enlloc de definir-les dins del codi. A més, el *framework* s'encarrega de gestionar les dependències important totes les llibreries que necessita el projecte automàticament, sempre i quan se li indiqui el seu tipus, que en aquest cas ha estat Maven. En darrer lloc, Spring Boot consta d'un servidor Apache Tomcat integrat (RNF-03), el que permet desplegar aplicacions web com μ Learning com si es tractés d'aplicacions Java *stand-alone*, en format JAR. Per una altra banda, la BD de l'aplicació és relacional, ja que les dades que emmagatzema estan fortament relacionades entre elles, i s'implementa en el sistema gestor MySQL (RNF-04).

Pel desenvolupament front-end de l'aplicació s'han emprat els llenguatges HTML, JavaScript (JS) i SCSS, on aquest darrer és una sintaxi del llenguatge Sass, més avançat que el llenguatge CSS pur [8]. Per la creació de les vistes s'ha utilitzat la llibreria Thymeleaf [9], que es tracta d'un motor de plantilles HTML5 que permeten treballar amb Spring MVC sense haver d'utilitzar JavaServer Pages (JSP) i utilitzar enlloc seu HTML solament. Thymeleaf també consta d'un conjunt de dialectes, expressions i operadors que faciliten la construcció de plantilles web, mitjançant l'enllaç de dades (*data binding*) provinents dels controladors. Altres llibreries emprades en el desenvolupament front-end han estat jQuery, per simplificar la manipulació del DOM i el codi JS, i Bootstrap (RNF-05), per implementar estils responsius que facin les vistes adaptables a les pantalles de múltiples dispositius (RNF-06).

5.3 Disseny

Un cop l'arquitectura ja s'ha establert, s'ha dissenyat el model relacional de la base de dades MySQL de l'aplicació. Per la realització d'aquesta tasca s'ha emprat l'eina de modelatge MySQL Model, integrada dins de l'entorn MySQL Workbench. Sobre les decisions preses pel que fa als camps de les taules, aquests han estat afegits d'acord a la informació que s'ha cregut rellevant per ser emmagatzemada, la

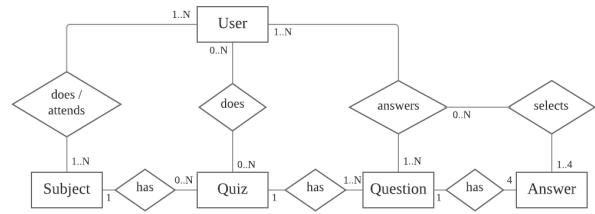


Fig. 4: Diagrama ER de la base de dades

definició del projecte i els requisits definits al seu inici. Cal dir, però, que el disseny ha estat subjecte a canvis al llarg de tota la fase de desenvolupament degut a decisions d'implementació, i per aquest motiu ha estat necessària la incorporació d'alguna nova taula o la modificació/eliminació d'alguns camps concrets. De totes maneres, les taules que s'han mantingut immutables des d'un inici i han representat les entitats principals de l'aplicació han estat: *User*, on es contenen les dades dels usuaris (alumnes i professors); *Subject*, que guarda les assignatures que usen l'aplicació; *Quiz*, que conté la informació sobre els qüestionaris creats pels professors; i *Question* i *Answer*, on s'emmagatzema la informació de les preguntes i respostes dels qüestionaris, respectivament.

Entre aquestes taules base n'han sorgit tres més, representant les relacions entre algunes d'elles. La primera és *User_Subject*, la taula que permet identificar, mitjançant una cardinalitat N-N, quins usuaris estan relacionats amb quines assignatures i quin rol hi desenvolupen, professors o alumnes. En aquesta taula a més, cal destacar que també s'hi emmagatzemen els casos en que un usuari pugui ser alhora alumne i professor, de dues assignatures diferents, és clar. La següent taula relació és *User_Quiz*, que sorgeix de la necessitat de guardar la informació que genera un alumne quan interactua amb un qüestionari des que hi accedeix per primer cop fins que l'envia, si ho fa. La cardinalitat d'aquesta taula es justifica com que un alumne pot respondre N qüestionaris i un mateix qüestionari pot ser respost per N alumnes. La darrera taula sorgida de la relació entre dues entitats és *User_Question*. Com el seu nom indica, en aquesta taula s'hi emmagatzema simplement la única relació que mantenen un alumne i les preguntes d'un qüestionari, és a dir, si l'alumne ha respost correctament cada pregunta. Com la taula anterior, *User_Question* també sorgeix d'una relació N-N, on un alumne pot respondre N preguntes i una pregunta pot ser resposta per N alumnes. Per últim, relacionada amb aquesta taula i la entitat *Answer*, es troba la taula *User_Question_Answer*. La incorporació d'aquesta taula al disseny s'ha degut a la necessitat de fer persistents les respostes seleccionades per un alumne a cada pregunta d'un qüestionari. D'aquesta manera, un cop enviat el qüestionari, es pot mostrar a l'alumne les respostes que ha marcat com a correctes per cada pregunta i si les ha encertat o no, oferint-li una correcció més acurada fins i tot dies després d'haver resolt el qüestionari.

Així doncs, aquest conjunt de taules han conformat el disseny de la BD de μ Learning, el qual pot ser consultat com a diagrama Entitat-Relació (ER) a la Figura 4. Per simplicitat, en aquest diagrama s'han omès els camps de les entitats i de les relacions que les uneixen ja que manquen de

rellevància respecte al disseny descrit en aquesta subsecció. Per una visió més detallada del disseny, consultar la Figura A.1, on es mostra el model relacional de la BD.

La posterior implementació del model dissenyat ha estat mitjançant un script SQL. Aquest script ha estat generat automàticament a partir del model relacional dissenyat a l'eina de MySQL Model gràcies a l'eina Forward Engineering de MySQL Workbench. Utilitzant aquesta eina s'ha assegurat la correctesa i optimització del codi SQL, enlloc d'escriure'l manualment. La manera com s'ha realitzat la creació de la BD ha estat a través de migracions amb *Flyway* [10], importada com una dependència dins del projecte. Aquesta eina executa, cada cop que s'arrenca l'aplicació web, i de manera automàtica, les migracions necessàries a la BD especificada a la propietat *spring.datasource.url* del fitxer de configuració *application.properties*. *Flyway* executa per defecte tots aquells scripts SQL que es trobin sota el directori *src/main/resources/db/migration*, per ordre de versió. Per això, s'ha decidit utilitzar una notació de versions basada en la data, de tal manera que els scripts SQL s'executen cronològicament, sense perill de sobreescriure dades de versions més noves amb d'altres més antigues. Així doncs, per la implementació del model de la BD s'ha col·locat el fitxer *V<YYYYMMDD>_bd_init.sql* al directori mencionat, per a que *Flyway* la intenti crear automàticament cada cop que s'arrenqui l'aplicació, sempre i quan no estigui ja creada.

5.4 Back End

5.4.1 Implementació del Model i els Controladors

El Back End de l'aplicació s'ha compost principalment del Model i de diferents Controladors. Al Model s'ha utilitzat una de les implementacions més usades de la Java Persistence API (JPA) en projectes amb Spring, Hibernate. Aquesta implementació es basa en el mapeig de model relacional a objectes (*Object Relational Mapping*) [11]. La organització del Model s'ha dividit en una jerarquia de tres capes, que són, de la més baixa a la més alta en nivell d'abstracció, les entitats, els repositoris, i els serveis. Les entitats representen les taules de la BD, i els seus atributs mapegen els camps i relacions de cadascuna. Per exemple, la taula *User* s'ha mapejat com una classe entitat que té per atributs els camps propis NIU, nom i contrasenya, i també una llista d'objectes *Subject*, corresponent a les N assignatures amb les quals aquell usuari té relació. Pel que fa als mètodes, les entitats únicament tenen els accessors als seus atributs i algun mètode addicional que els manipuli, a part del constructor, si s'escau.

Per tal que la JPA sigui capaç de comunicar-se amb la BD, Hibernate ofereix una sèrie de mètodes per realitzar operacions de creació, lectura, actualització i eliminació (CRUD) sobre la BD. L'ús de les operacions CRUD s'ha implementat als repositoris, encarregats d'interactuar directament amb la BD recuperant o emmagatzemant entitats de la BD al model de l'aplicació o viceversa. En el nivell més alt de la jerarquia del Model es troben els serveis, els responsables d'executar mètodes dels repositoris quan un Controlador requereix algun conjunt de dades. Als serveis s'hi ha efectuat un pre-processament de les dades recuperades pels repositoris, per tal que al Controlador hi arribin exactament les dades que requereixi del Model. A més de les

classes d'aquesta jerarquia, també han estat necessaris dos tipus de classes auxiliars del Model, els accessors i els *Data Transfer Objects* (DTOs). Els primers han estat necessaris quan els controladors han requerit conjunts de dades tant concrets que la recuperació d'entitats senceres i el seu posterior tractament resultava ser més costós que realitzar una consulta específica sobre la BD. En aquests casos, s'han definit mètodes als accessors que realitzen aquest tipus de consultes i retornen el resultat directament al Controlador. D'altra banda, els DTO s'han emprat pel mapeig d'estructures de dades complexes que contenen dades de múltiples entitats.

Els Controladors de l'aplicació han estat tres. El primer ha gestionat aquelles funcions comunes per tots els usuaris de l'aplicació, com l'inici de sessió o la càrrega de qüestionaris de les assignatures. Els altres dos s'han encarregat de les funcions específiques per professors o alumnes, com la creació de qüestionaris o la resolució d'aquests, respectivament. A més, també s'ha comptat amb dos controladors addicionals, un pel control d'errors i l'altre per la gestió d'excepcions.

5.4.2 Implementació dels casos d'ús

Havent comentat com s'han construït el Model i els Controladors, en aquesta secció s'explica la implementació backend dels casos d'ús de l'aplicació. Abans però, s'ha implementat l'inici i tancament de sessió mitjançant Spring Security [12], amb el que s'ha pogut implementar un control d'accés basat en rols (*Role Based Control Access*, RBAC) i un xifratge de credencials d'usuari. Sobre l'RBAC, aquest s'ha format amb 3 possibles rols: alumne, professor, o administrador, emmagatzemats al camp *user.role* de la taula *User.Subject*. El darrer dels tres, però, no s'ha posat en pràctica ja que dins de l'abast d'aquest treball no s'ha contemplat implementar cap funcionalitat d'administrador, tot i que sí que s'ha valorat la possibilitat de desenvolupar aquest rol amb permisos especials en un futur. D'altra banda, la implementació dels altres dos s'ha dut a terme tenint en compte l'arquitectura de l'aplicació. Així, quan un alumne o professor inicia sessió se'l redirigeix a la banda de l'aplicació corresponent. En el cas especial en què un usuari exerceix el rol d'alumne en una assignatura però el de professor en una altra, quan aquest usuari inicia sessió és redirigit prèviament a una pàgina d'índex on se li permet triar amb quin dels dos rols vol iniciar sessió a la plataforma. Amb l'RBAC també s'han pogut controlar els accessos no autoritzats a pàgines de l'aplicació restringides a un cert rol, llençant una excepció d'accés denegat quan això succeeix. Pel que fa al xifratge, tot i que s'ha implementat l'ús d'una funció hash sobre les credencials dels usuaris, a la pràctica no s'ha aplicat perquè aquest treball s'ha limitat únicament a un entorn de desenvolupament. De totes maneres, s'espera que eventualment μ Learning es desplegui en l'entorn informàtic de la UAB i, per tant, faci ús del *Central Authentication Service* (CAS) del servei informàtic de la Universitat, el qual ja compta amb la seguretat pertinent i una BD amb tots els usuaris de la institució.

Amb un Login/Logout funcional, s'han passat a implementar les funcionalitats englobades dins dels casos d'ús de l'aplicació. Inicialment s'ha implementat la creació (i edició) de qüestionaris (RF-01), realitzada pels professors a

partir de l'enviament d'un formulari. Les dades del formulari són tractades en un mètode del controlador que guarda o actualitza, a través del Model, el qüestionari creat o editat sobre les taules *Quiz*, *Question* i *Answer*. Alguns camps a destacar a la taula *Quiz* són la data d'expiració (RF-02) i de darrera modificació, ambdues emmagatzemades en format UCT (*Universal Coordinated Time*), per mantenir la internacionalització de l'aplicació, però tractades amb la zona horària del servidor al Back End. També és destacable el camp *enabled*, el qual indica si un qüestionari es troba publicat o no. Un professor pot publicar un qüestionari al moment de la seva creació o bé posteriorment (RF-03). En cas que ho faci posteriorment, s'utilitza un mètode implementat al controlador dels professors que simplement modifica el booleà del camp *enabled* del qüestionari que el professor hagi volgut publicar. Pel que fa al qüestionari en sí, aquest té una dificultat assignada pel professor (RF-04) que es guarda al camp *difficulty*, i també pot incloure imatges que acompanyin opcionalment l'enunciat de les preguntes i respostes (RF-05). Les imatges s'emmagatzemen a la BD dins de les taules respectives, al camp *image*, de tipus *MEDIUMBLOB*. L'elecció d'aquest tipus pel seu emmagatzematge s'ha degut a que la mida màxima permesa per cada imatge és d'1 MB, una mida força major als 64 KB que pot contenir el tipus *BLOB*, aproximadament.

Un cop s'ha publicat un qüestionari, aquest passa a ser visible pels alumnes que cursen l'assignatura a la que pertany. En aquest punt s'ha implementat la consulta de qüestionaris. Donat que la lògica darrere d'aquesta funcionalitat és la mateixa tant per alumnes com per professors, aquesta s'ha implementat al controlador compartit per ambdós rols. Bàsicament, el controlador realitza un petició al Model demanant els qüestionaris de totes les assignatures amb les que l'usuari està relacionat, o bé únicament les d'una assignatura concreta si l'usuari aplica un filtre a la seva consulta (RF-06). Després, el controlador els carrega dinàmicament a la vista corresponent per a que l'usuari els pugui consultar i realitzar-hi accions. Els professors poden editar el qüestionari i també fer-lo visible o ocultar-lo als alumnes, canviant el seu estat de publicació tal i com s'ha mencionat anteriorment. En canvi, els alumnes únicament poden respondre el qüestionari o accedir a veure les seves respostes i la solució un cop l'han respost o ha expirat.

Quan un alumne accedeix per primer cop a un qüestionari actiu pot resoldre'l dins d'un temps limitat pel professor (RF-07). L'alumne simplement ha de seleccionar la resposta o respostes que cregui correctes per cada pregunta tipus test del qüestionari i enviar-lo (RF-08). Quan ho fa, s'envia una estructura de dades complexa a un mètode definit al controlador dels alumnes, el qual la rep i la mapeja directament a un DTO. La estructura és una llista d'objectes JSON, corresponents a cada pregunta resposta per l'usuari. Per tant, cadascun d'aquests JSON tenen com a clau l'ID únic de la pregunta i com a valor una llista amb els ID's de les respostes que l'usuari ha seleccionat com a correctes per aquella pregunta. Així doncs, posteriorment s'ha processat el DTO i s'ha emmagatzemat a BD la informació referent a la resolució d'aquell qüestionari, a la taula *User_Quiz* concretament. Els camps manipulats han estat *start_date* i *answer_date*, amb la data i hora d'inici i enviament del qüestionari, *device* i *origin*, amb el tipus de dispositiu (PC, Tauleta, o Mòbil) i IP de la xarxa des d'on s'ha enviat el

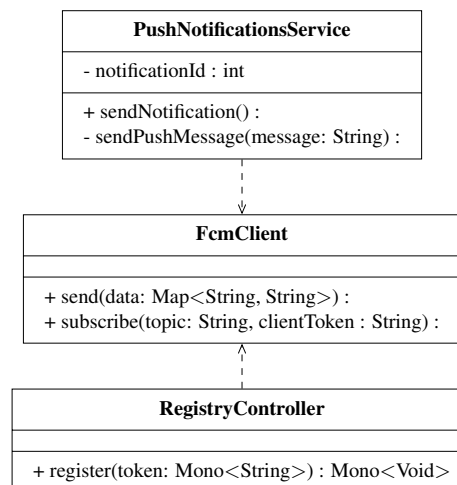


Fig. 5: Diagrama de classes del sistema de notificacions

qüestionari respectivament, i *mark*, on es guarda la nota obtinguda, sorgida de la ponderació sobre 10 punts del nombre total de preguntes encertades. Per realitzar aquest càlcul s'ha hagut de guardar el resultat de cada pregunta al camp *correctly_answered* de la taula *User_Question*, considerant-la correcta sempre i quan totes les respostes seleccionades siguin correctes. A més, per poder saber permanentment quines van ser les respostes d'un alumne a un qüestionari determinat, ha estat necessari emmagatzemar-les a la taula *User_Question_Answer*, relacionades amb la pregunta on l'alumne les va seleccionar.

El sistema de notificacions (RF-09) s'ha dut a terme mitjançant el servei Google Firebase Cloud Messaging (FCM) [13], escollit per ser gratuït i oferir la possibilitat d'enviar notificacions web PUSH il·limitades als clients des del servidor [14], ja siguin clients concrets, un grup, o tots aquells subscrits a un *topic*. Així doncs, la implementació proposada ha estat la que es descriu tot seguit. S'ha desenvolupat un servei de notificacions, encarregat de realitzar les comprovacions pertinents al Back End de l'aplicació per decidir a quins alumnes s'enviarà la notificació. Aquestes comprovacions es realitzen periòdicament, per així assegurar-se que l'alumne les rep eventualment. Així doncs, el servei transmet la notificació a enviar a un client de FCM, el qual realitza l'enviament del missatge a FCM mitjançant la anomenada *Firebase Admin SDK* i una autenticació prèvia amb una clau privada. Quan FCM rep el missatge, intenta entregar-lo a tots els clients que coneix, ja sigui individualment o per subscripció prèvia a les notificacions, contínuament fins passat un *timeout* determinat. D'aquesta manera, quan l'alumne es troba connectat al dispositiu des d'on s'ha subscrit al servei, rep la notificació indicant-li que hi ha qüestionaris pendents per respondre. La Figura 5 mostra el diagrama de classes UML del sistema descrit.

Per últim, cal dir que el cas d'ús de consulta d'estadístiques per part dels professors (RF-10) no s'ha pogut dur a terme per un canvi en la planificació, arrel de l'aplicació d'una mesura de contingència en un moment crític durant la fase de desenvolupament. A la secció 6 es detallen les causes que han portat a aquesta situació i l'impacte sobre els resultats del treball.

5.5 Front End

5.5.1 Implementació de les vistes

Paral·lelament al Back End, s'ha desenvolupat el Front End de l'aplicació. Per la implementació de les vistes s'han aprofitat al màxim les capacitats del motor *Thymeleaf*. La primera característica emprada ha estat el *layout*, el que permet crear vistes seguint un estructura fixada definida en una plantilla base. Per aquesta aplicació s'ha seguit la estructura bàsica d'una aplicació web: *header*, *body*, i *footer*. El *header* i el *footer* s'han mantingut estàtics per totes les vistes de l'aplicació. El *header* simplement té el logo de μ Learning, que redirigeix a l'usuari a la pàgina principal al clicar-lo, i el botó per tancar sessió. El *footer* té els links que porten a les pàgines web de la UAB, la Escola d'Enginyeria, i el dEIC, i un *disclaimer* junt amb un text que indica amb quin rol s'ha iniciat sessió a la plataforma. En el cas dels *body's*, el desacoblament comentat anteriorment pel que fa a les bandes de l'aplicació per professors i alumnes, s'ha vist reduït en certs casos en què alguns components de les vistes s'han reutilitzat a les dues bandes, amb l'objectiu de minimitzar la repetició de codi. Cal destacar també que tot el text que es mostra al Front End de l'aplicació està únicament en català, però és fàcilment extensible a qualsevol altre idioma gràcies a la capacitat d'internacionalització (i18n) de Spring Boot (RNF-07), que ofereix la possibilitat de carregar *strings* en diferents idiomes directament des d'arxius de tipus *properties*, basant-se en el valor de l'objecte *Locale*.

Abans de comentar les vistes particulars de professors i alumnes, es mencionen les vistes compartides. Unes han estat les pàgines dels errors més comuns, com el 403 (Accés denegat), 404 (Recurs no trobat), 500 (Error intern del servidor), totes elles definides com el número i títol de l'error a la part superior i un breu subtítol a sota explicant-ne la causa juntament amb un enllaç per redirigir l'usuari a la pàgina principal de l'aplicació. Una altra vista comuna implementada ha estat la pàgina de *login/logout* (veure Figura A.2), on s'ha desenvolupat un formulari amb dos camps, NIU i contrasenya. A més, en aquesta pàgina hi pot aparèixer una alerta amb el missatge "Credencials incorrectes" quan l'usuari intenta iniciar sessió amb credencials errònies, o bé "Sessió tancada correctament" quan l'usuari ha tancat la seva sessió a l'aplicació. Com s'ha comentat anteriorment, aquesta vista queda subjecta a canvis en cas que μ Learning utilitzi la pàgina d'autenticació del CAS de la UAB en un futur. La darrera vista comuna per tots els usuaris ha estat l'índex (veure Figura A.3), des d'on els usuaris amb rol múltiple correctament autenticats poden escollir entre dos links que els duguin a la banda de l'aplicació corresponent segons el rol amb el que volen iniciar sessió. També s'ha tractat com a compartida la vista *main*, la qual correspon a la pàgina principal de l'aplicació on es poden consultar tots els qüestionaris en una llista, segons assignatura. D'aquesta vista s'ha reutilitzat la estructura, el filtre de qüestionaris per assignatura i la generació iterativa del llistat de qüestionaris. En aquest llistat, en canvi, s'han realitzat implementacions diferents per professors (veure Figura A.4) i alumnes (veure Figura A.5), gràcies a la segona característica de *Thymeleaf* emprada, els *fragments*. Els fragments són blocs de codi HTML tractats com una unitat reutilitzable entre algunes pàgines de l'aplicació o repetida diverses vegades en una mateixa vista. Així doncs, s'han desenvolupat

dos fragments, un per cada rol d'usuari, els quals representen cada qüestionari element de la llista, i s'anomenen *studentQuizBox* o *teacherQuizBox* segons a la vista *main* on es mostrin. Un altre component en aquesta vista ha estat un botó tipus *switch* per canviar el mode de la vista, edició o estadístiques, exclusiu per professors. En mode edició poden consultar la informació principal sobre els qüestionaris i realitzar-hi accions com ocultar-los o publicar-los, o anar a la pàgina d'edició directament prement un botó dins de cada *teacherQuizBox*; mentre que en mode estadístiques poden consultar un resum de les estadístiques de cada qüestionari llistat o bé accedir a la pàgina d'estadístiques on es mostrarien les estadístiques globals per assignatura. Tot i això, aquest últim mode no s'ha implementat finalment, tal com s'ha comentat a la secció anterior.

Pel que fa a les vistes particulars, es comenten tot seguit. Pels professors només s'ha desenvolupat la vista de creació d'un qüestionari (veure Figura A.6), que consta simplement d'un formulari a omplir amb la informació d'un qüestionari i un fragment de *Thymeleaf* amb el formulari d'una pregunta. Aquest permet omplir l'enunciat, el *feedback* i la imatge opcionals, i les respostes, també amb imatge opcional, on cadascuna té un *checkbox* per a que el professor determini quina o quines són les respostes correctes a la pregunta. La implementació d'aquest darrer formulari com un fragment s'ha degut a que el nombre de preguntes d'un qüestionari és variable, i per això s'ofereix al professor la possibilitat d'afegir preguntes dinàmicament. Aquesta mateixa vista també permet editar els qüestionaris modificant el valor dels camps directament sobre un formulari carregat. Per la banda dels alumnes, s'ha desenvolupat la vista de resolució d'un qüestionari (veure Figura A.7), que consta d'una part superior amb la informació del qüestionari, on es reutilitza el fragment *studentQuizBox*, i un llistat de fragments corresponents a les preguntes del qüestionari (*quizQuestion*).

5.5.2 Implementació JS i CSS

A totes les vistes de l'aplicació s'han implementat també funcionalitats JS i estils CSS. A la pàgina de creació/edició de qüestionaris, l'addició dinàmica de preguntes al qüestionari es realitza a través d'un botó enllaçat amb JS a una crida asíncrona AJAX al servidor. En aquesta crida, el servidor afegeix una pregunta al llistat de preguntes del qüestionari que està lligat al formulari, i després retorna el fragment d'una nova pregunta per emplenar (*questionForm*). Després, quan el professor guarda el qüestionari (o el guarda i publica) es realitza una validació JS que comprova si totes les preguntes tenen com a mínim una resposta correcta seleccionada i que les imatges, si se n'han pujat, compleixen els requisits de mida màxima (1 MB) i format (PNG, JPG, JPEG). Si s'incompleix alguna d'aquestes restriccions el professor rep una alerta en pantalla i s'impedeix l'enviament del formulari (veure Figura A.8). A la banda dels alumnes, quan s'accedeix per primer cop a resoldre un qüestionari s'activa el comptador de temps per respondre'l, implementat en JS, el qual es manté actiu independentment de si l'alumne abandona la pàgina del qüestionari, ja sigui clicant a "Sortir" o tancant l'aplicació. Això és possible havent guardat a la BD la data i hora d'inici del qüestionari, evitant així que es pugui reiniciar el comptador forçosament o per un error al client. Quan el temps s'ha exhaurit o quan

el qüestionari ha expirat apareix una alerta en pantalla (veure Figures A.9 i A.10, respectivament), implementada amb Bootstrap Modal, indicant això a l'alumne i oferint la solució al qüestionari amb el *feedback* pertinent a cada pregunta. Si, en canvi, l'alumne respon el qüestionari, pot consultar la solució i el *feedback* immediatament després de respondre'l, o més endavant si ho prefereix (RF-11). Per fer la comprovació de la solució més intuïtiva s'ha implementat un estil CSS que mostra les respostes correctes en verd i les incorrectes en vermell.

Amb l'objectiu de poder reutilitzar tots els fragments de vistes esmentats per a diferents funcions s'ha definit un *body* amb una estructura responsiva i adaptable a la inserció d'aquests fragments. Això s'ha aconseguit fent ús del *framework* Bootstrap Grid, basat en la definició d'elements contenidors amb files i columnes a dins mitjançant únicament classes CSS. A part de la estructura, també s'han hagut de definir múltiples classes SCSS, flexibles a l'estat de certes variables, seguint la notació *Block Element Modifier* (BEM) per tal d'estilar components específics [15]. Amb aquesta notació, en conjunt amb les expressions lògiques i operadors de *Thymeleaf*, s'han pogut aplicar diferents estils a mateixos fragments de codi HTML de manera dinàmica. Un exemple d'això són les *quizBoxes*, les quals canvien el seu color depenent de l'estat en què es trobi el qüestionari i el rol de l'usuari. Els professors veuen un estat binari, si el qüestionari està publicat (verd) o ocult (gris), mentre que els alumnes els poden diferenciar entre els estats disponible (blau), respost i aprovat (verd), o respost i suspès, expirat o temps de resposta exhaurit (vermell) (RF-12).

6 RESULTATS

Després de la fase de desenvolupament s'ha realitzat un anàlisi dels resultats obtinguts, fins aquest punt del projecte. Això ha estat així a causa de que la magnitud del projecte no ha permès completar-lo en la seva totalitat, el que inclouria una segona fase de desenvolupament, un procés de testeig i un desplegament en un entorn real de producció. És per això que els resultats comentats en aquesta secció s'han analitzat des del punt de vista del desenvolupament realitzat, en termes qualitius. Per tant, s'analitza el grau d'acompliment dels requisits definits, la qualitat del producte obtingut fins al moment i la extensibilitat per a continuació futura.

Primerament, en valoració global es considera que els resultats obtinguts són força satisfactoris. De les tasques de desenvolupament s'han assolit totes a excepció de dues, que són la finalització del sistema de notificacions (RF-09) i la consulta d'estadístiques a la banda dels professors (RF-10). Degut als retards produïts durant tota la fase de desenvolupament, es va arribar al moment d'implementació de les dues tasques mencionades amb molt pocs recursos temporals. Aleshores, es va optar per intentar realitzar una de les dues tasques. Encara que el risc mitjà del sistema de notificacions era més alt que el risc baix de la consulta d'estadístiques, es va considerar que el valor que podien proporcionar les notificacions, en termes d'*engagement* dels alumnes amb l'aplicació, era major que el valor que aportarien les estadístiques. Per això, com a contingència davant d'aquest punt crític en el desenvolupament, es va decidir prioritzar la realització d'aquesta tasca, inicialment planificada com la darrera a ser implementada, per davant de la consul-

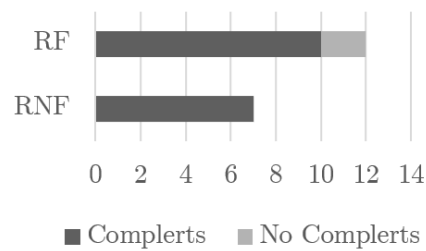


Fig. 6: Acompliment de requisits

ta d'estadístiques. Tot i això, una caiguda del servei FCM durant la implementació de la tasca ha causat que finalment no s'hagi pogut completar. Ara bé, malgrat els imprevistos finals, es valora que la càrrega de treball assignada ha resultat ser adequada per un enginyer informàtic dins de la finestra temporal disponible per realitzar el treball. Això s'ha reflectit en l'elevat grau d'acompliment dels requisits definits, tal i com es pot comprovar a la Figura 6.

Sobre la qualitat de l'aplicació, s'ha prioritzat la simplicitat d'ús que ofereixen les aplicacions mòbils davant de la sobrecàrrega i la multitud de pestanyes, desplegable, menús laterals i altres elements que presenten els llocs webs tradicionals. A aquest fet cal afegir-li l'enfoc basat en aplicacions web tipus *single-page*, on gràcies a la càrrega asíncrona i dinàmica d'elements *in page* dona la sensació a l'usuari de canviar de pantalles d'una aplicació enlloc de navegar per pàgines completament diferents entre elles. Pel que fa al disseny, gràcies a la *responsiveness* dels elements en pantalla, aquest s'adapta a les pantalles de múltiples dispositius (veure Figura A.11). Sobre la interacció de l'usuari, el desplaçament per l'aplicació (*scroll*) és únicament vertical i els elements com botons, alertes o enllaços no requereixen clics llargs, desplaçament per submenús o d'altres interaccions que resultin incòmodes de realitzar sense un ratolí. Aquest tipus d'interacció, doncs, apropa μ Learning encara més a una app mòbil, on els usuaris es desplacen lliscant el dit verticalment i hi interactuen mitjançant pulsacions curtes sobre una pantalla tàctil. Per aquest motiu, es considera que aquest disseny té en compte la usabilitat de l'aplicació i com a conseqüència millora la experiència d'usuari.

Encara que, com s'ha discutit fins ara, globalment s'ha obtingut una aplicació funcional, s'ha deixat oberta a desenvolupament futur. La seva implementació s'ha orientat a un codi modular, programat amb l'objectiu de ser auto-documentat, tot i que també inclou alguns comentaris, tant a Back End com a Front End, adequats per a que sigui possible la seva completa comprensió per part d'altres desenvolupadors. També, gràcies a l'ús del patró MVC en conjunt amb el mòdul Spring MVC, la extensibilitat del projecte resulta molt més senzilla. Per exemple, si fos necessària la implementació d'una banda d'administració a l'aplicació, simplement caldria afegir un usuari a la BD amb rol administrador i desenvolupar un nou controlador amb els *end-points* necessaris per carregar la vista d'administració i tota la seva lògica. A més, donada la maduresa i alta adopció del patró MVC en el desenvolupament web, s'espera que sigui molt més senzilla la familiarització amb el projecte per la gran majoria de desenvolupadors. Cal dir, però, que tot i així és completament necessari el coneixement de les tecnologies emprades en el projecte, especialment *Spring* pel desenvolupament back-end i *Thymeleaf* pel front-end.

Per últim, degut a que l'aplicació es troba encara en desenvolupament, no s'ha redactat la documentació associada, com seria el manual d'usuari i el manual de referència. Tot i haver inclòs aquestes tasques a la planificació del projecte, al seguir una metodologia àgil, s'ha prioritzat el desenvolupament per davant de la redacció de documentació al llarg de les iteracions del projecte.

7 CONCLUSIONS

Les conclusions a les que s'arriba en la finalització d'aquest TFG són que s'han cobert la majoria dels requisits tècnics definits al seu inici, complint així els objectius més prioritaris dels proposats pel projecte.

En la realització d'aquest treball s'ha desenvolupat una aplicació web funcional que suposa una innovació docent i en el moment de la seva implantació final oferirà als professors una eina complementària a les classes presencials (O-1). La seva utilització suposarà un suport en l'assoliment de coneixements dels alumnes (O-2) i s'ha vist com el seu disseny intuïtiu i portable a qualsevol dispositiu motiva el seguiment de les assignatures fora de les aules (O-3). Ara bé, degut a que el projecte es troba encara en desenvolupament, no es pot valorar l'acompliment d'aquells objectius subjectes al desplegament i ús de l'aplicació per part de l'alumnat, com són la millora dels resultats acadèmics (O-4) i les valoracions positives a les enquestes de professorat i assignatures (O-5).

Resten com a futur desenvolupament les tasques no completades en la planificació del projecte. Un cop finalitzat el desenvolupament planificat es requereix la realització de proves del sistema per assegurar la seva qualitat i correcte funcionament. Addicionalment, es proposen com a extensions futures del projecte l'ús de HTTPS com a protocol segur a nivell d'aplicació, la millora de la seguretat web contra atacs com XSS o CSRF, i l'ampliació de la internacionalització als idiomes castellà i anglès. També, es podria afegir la possibilitat de que els professors puguin enviar notificacions personalitzades per assignatura o qüestionari, així com millorar la interfície d'usuari per a que sigui més vistosa sense perdre la usabilitat actual. Per últim, serà necessari un desplegament en un entorn real de producció, pel que es recomana valorar la implantació d'un servidor d'integració i desplegament continu que realitzi tests unitaris i d'integració a cada *commit* del projecte, i posteriorment el desplegui al servidor web corresponent.

AGRAÏMENTS

Aquest treball no hauria estat possible sense l'ajut i suport d'algunes persones. Considero que mereixen especial menció el professor Joan Bartrina Rapesta, per la seva bona tutorització i disponibilitat durant el treball, i el meu company i amic Óscar Sánchez Bocero, pel seu ajut en moments difícils quan no sabia com seguir endavant.

Per últim, m'agradaria agrair la paciència dels meus pares, que han viscut els meus maldecaps de ben a prop però tot i així han continuat donant-me suport, i a la meva parella Xènia, per tot l'afecte i confiança que m'ha transmès no només durant aquest treball sinó també al llarg del grau.

REFERÈNCIES

- [1] I. Fernández, “Juego serio: gamificación y aprendizaje.” *Comunicación y pedagogía: Nuevas tecnologías y recursos didácticos*, no. 281-282, pp. 43-48, 2015.
- [2] R. Pàmies, A. Fabregat, J. Puig, L. Jordi, i A. Hernández, “Ús de Kahoot! com a eina de ludificació per a la retroalimentació a temps”, 2018.
- [3] A. Sánchez, S. Robles, i J. Pons, “A gamification experience to improve engineering students' performance through motivation”, *JOTSE*, vol. 7, no. 2, 2017.
- [4] K. Beck, et al., “Principles behind the Agile Manifesto”, 2001.
- [5] “Scrum: qué es, cómo funciona y por qué es excelente”. [En línia]. Disponible a: <https://es.atlassian.com/agile/scrum>. Consulta: 28 de febrer de 2019.
- [6] “Using Trello for Agile Software Development: The Complete Guide”. [En línia]. Disponible a: <http://buildbettersoftware.com/trello-for-software-development>. Consulta: 28 de febrer de 2019.
- [7] “Spring Boot”. [En línia]. Disponible a: <https://spring.io/projects/spring-boot>. Consulta: 1 de març de 2019.
- [8] Cem Eygi, “Advantages of Using a Preprocessor (Sass) in CSS”, *Medium*, Nov. 2018. [En línia]. Disponible a: <https://medium.com/swlh/advantages-of-using-a-preprocessor-sass-in-css-eb7310179944>. Consulta: 10 de març de 2019.
- [9] “Thymeleaf”. [En línia]. Disponible a: <https://www.thymeleaf.org/>. Consulta: 11 de març de 2019.
- [10] “Database Migrations with Flyway”, *Baeldung*, Nov. 2018. [En línia]. Disponible a: <https://www.baeldung.com/database-migrations-with-flyway>. Consulta: 6 de març de 2019.
- [11] “Hibernate ORM”. [En línia]. Disponible a: <https://hibernate.org/orm/>. Consulta: 7 de març de 2019.
- [12] C. Scaroni, “Web Security”, dins *Pro Spring Security*. Apress, 2013, pp. 57-110.
- [13] “Firebase Cloud Messaging”. [En línia]. Disponible a: <https://firebase.google.com/products/cloud-messaging>. Consulta: 23 de maig de 2019.
- [14] “Sending Web push messages from Spring Boot to Browsers”, Des. 2018. [En línia]. Disponible a: <https://golb.hplar.ch/2018/01/Sending-Web-push-messages-from-Spring-Boot-to-Browsers.html>. Consulta: 23 de maig de 2019.
- [15] “CSS Naming Conventions that Will Save You Hours of Debugging”. [En línia]. Disponible a: <https://www.freecodecamp.org/news/css-naming-conventions-that-will-save-you-hours-of-debugging-35cea737d849/#the-bem-naming-convention>. Consulta: 25 d'abril de 2019.

APÈNDIX

Tot seguit es presenten el model relacional de la base de dades i algunes figures que mostren les vistes mencionades al llarg d'aquest document.

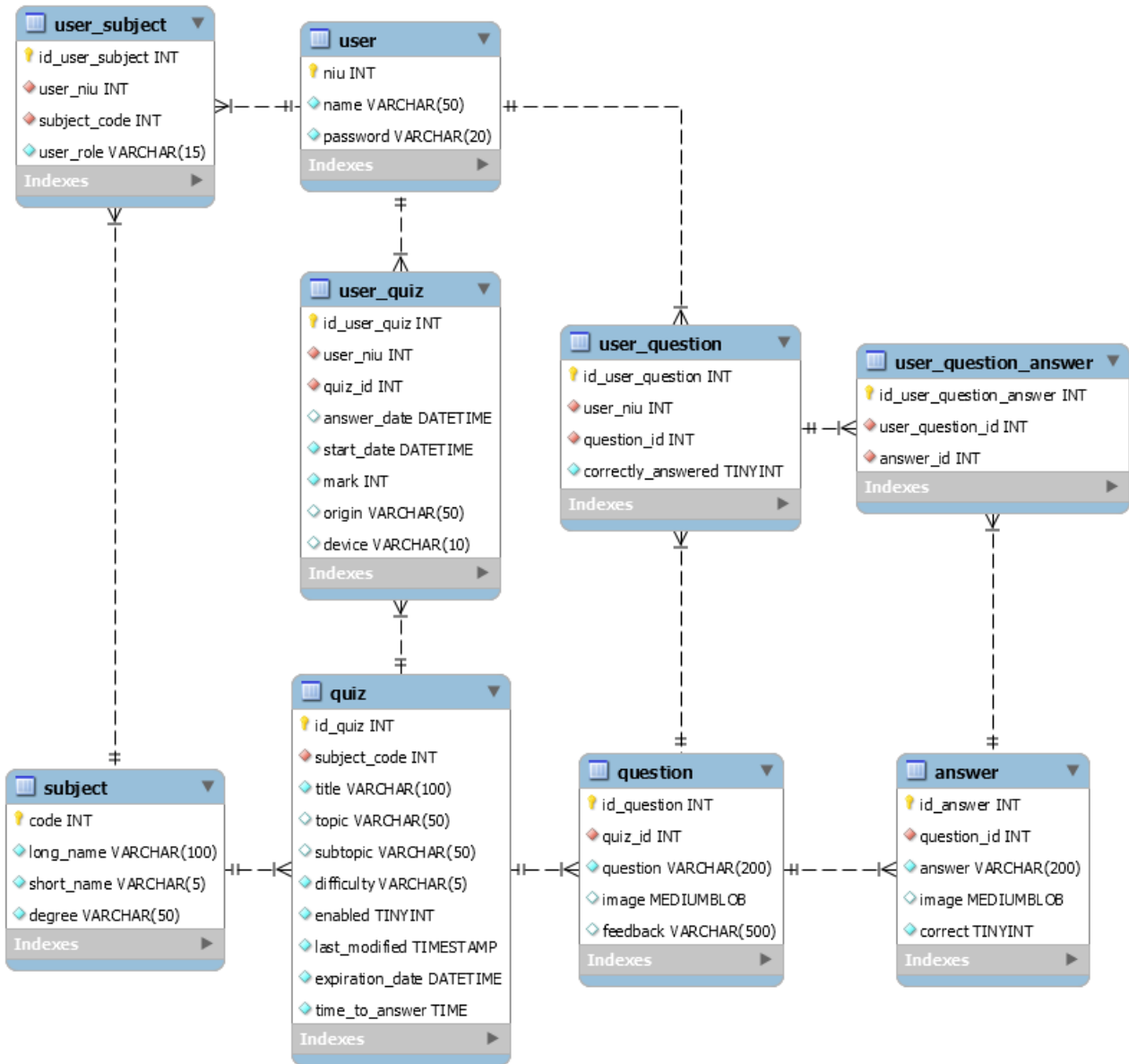


Fig. A.1: Model relacional de la base de dades

Iniciar sessió



	NIU
	Contrasenya
Iniciar sessió	

Fig. A.2: Pàgina Login/Logout


On vols anar?

Ets professor i alumne. Escolleix a quina secció vols accedir:

[Estudiant](#) [Professor](#)

Fig. A.3: Pàgina Índex - Elecció de redirecció

Qüestionaris

 Assignatura*














QUIZ GDS	Tema: A Subtema: B Dificultat: Alta  29/06/2019 18:33:38  01:00:30	 
QUIZ ATS - Hadoop	Tema: Big Data Subtema: Plataformes Big Data Dificultat: Mitja  30/06/2019 20:19:03  01:00:30	 
QUIZ ATS - Docker	Tema: DevOps Subtema: Deployment Dificultat: Mitja  14/05/2019 14:10:43  01:00:30	 



Fig. A.4: Pàgina principal per professors - Consulta de qüestionaris

Qüestionaris

 Assignatura*










QUIZ ATS - Docker	Tema: DevOps Subtema: Deployment Dificultat: Mitja  14/05/2019 14:10:43  01:00:30 Nota: 0	
QUIZ STW - EJB	Tema: Java Avançat Subtema: Desenvolupament Web Dificultat: Mitja  30/06/2019 20:19:03  01:00:30 Nota: 10	
QUIZ ATS - Hadoop	Tema: Big Data Subtema: Plataformes Big Data Dificultat: Mitja  30/06/2019 20:19:03  01:00:30 Nota: --	

Fig. A.5: Pàgina principal per alumnes - Consulta de qüestionaris

Fig. A.6: Pàgina de creació/edició d'un qüestionari

Fig. A.7: Pàgina de resolució d'un qüestionari

Fig. A.8: Pàgina de creació/edició d'un qüestionari - Validació amb errors



Fig. A.9: Pàgina de resolució d'un qüestionari - Temps exhaurit

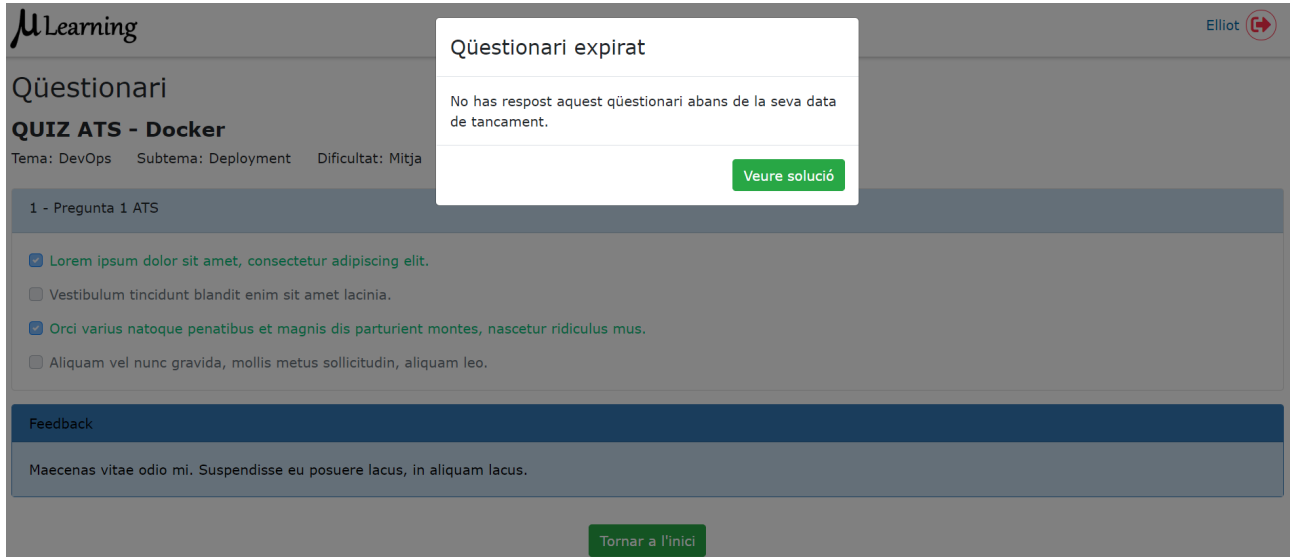
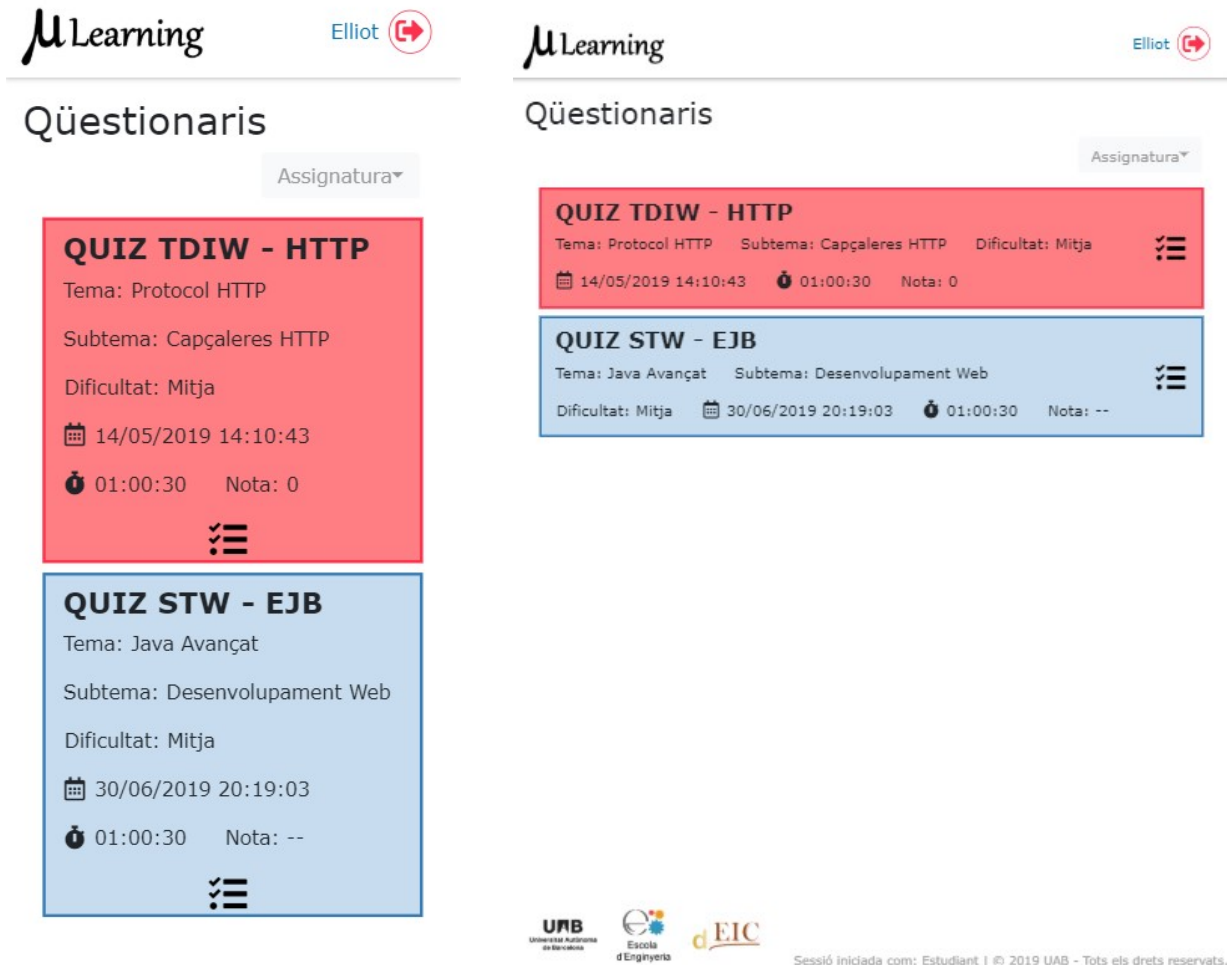


Fig. A.10: Pàgina de resolució d'un qüestionari - Qüestionari expirat



(a) Samsung Galaxy S10

(b) iPad Pro

Fig. A.11: Vista responsiva en dispositius mòbils i tauletes