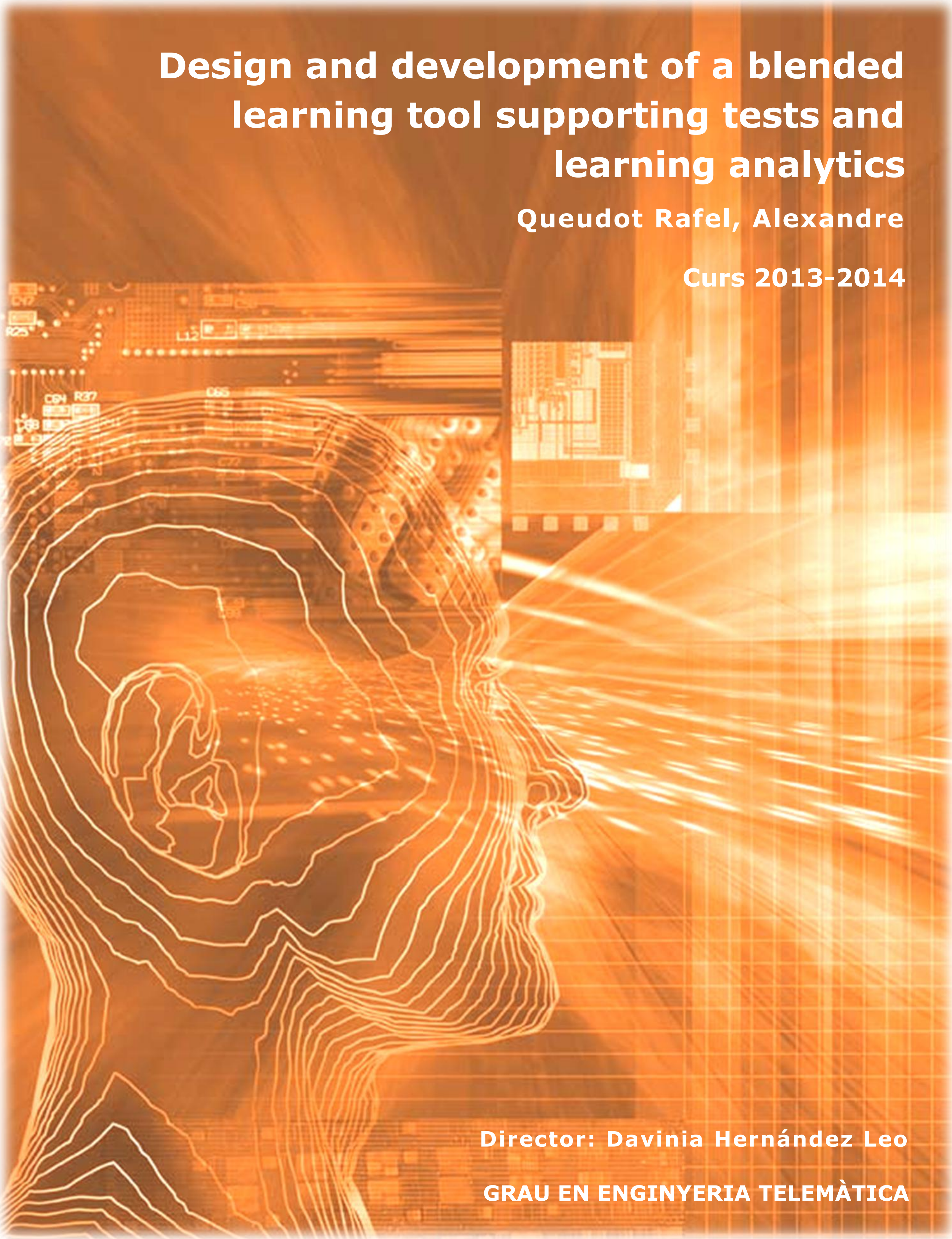


Design and development of a blended learning tool supporting tests and learning analytics

Queudot Rafel, Alexandre

Curs 2013-2014

Director: Davinia Hernández Leo

GRAU EN ENGINYERIA TELEMÀTICA

Universitat Pompeu Fabra Barcelona
Escola Superior Politècnica

Treball de Fi de Grau

Design and development of a blended learning tool supporting tests and learning analytics

Queudot Rafel, Alexandre

TREBALL FI DE GRAU

GRAU EN ENGINYERIA TELEMÀTICA

ESCOLA SUPERIOR POLITÈCNICA UPF

2014

DIRECTORA DEL TREBALL

Hernández Leo, Davinia

Dedicado a mis pequeños alumnos,
quienes me han inspirado a buscar los mejores modos de educar
y me han enseñado a hacerlo siempre con una sonrisa

Special Thanks

To Davinia, for believing in me and my chaotic way of working.

To my friends and coworkers, for cheering me up during this year.

And to my students, who inspire me every day.

Abstract

Technology has been growing exponentially for the last decades, with big tech companies fighting over to be leaders in creativity and innovation, pushing our society forward and trying to deliver the best products to the users. Currently, this race is beginning to take place in education as well.

With all the new discoveries in psychology, marketing and technology, there has not been though a groundbreaking application or pedagogic techniques applied in schools or universities to revolutionize the established educational system. However, governments and many private institutions are realizing that the school drop-out rates and low student engagement are a huge problem for the long term development of a country and society as a whole.

This project intends to analyze the current state of some technological appliances in education as well as developing a teaching web-based application from the ground.

The aim of the platform will be brining knowledge in a more natural and comfortable way as well as promoting teacher-student interaction. This will be accomplished by deploying a system where teachers can upload their fixed content that students need to learn and create basic quizzes to verify the user's comprehension on the topics. Furthermore, this will allow the display of detailed yet simple statistics for teachers to keep track of the course in a quick and visual look.

Resum

La tecnologia ha estat creixent de forma exponencial en les últimes dècades, amb grans empreses buscant el lideratge en creativitat i innovació, tirant endavant la nostra societat i buscant fer arribar els millors productes possibles als usuaris. Actualment estem començant a veure un desenvolupament similar en l'educació.

Tot i els nous descobriments en psicologia, màrqueting i tecnologia, no hi ha hagut cap aplicació o mètodes pedagògics que s'hagin aplicat en les escoles o universitats per revolucionar el sistema educatiu establert. Tot i així, governs i també empreses privades estan començant a veure que les taxes de finalització d'estudis o la falta d'interès per part dels estudiants són un gran problema de cara al desenvolupament global d’un país i d’una societat.

Aquest projecte busca ser un anàlisis de l’estat actual d’algunes exitoses aplicacions usades actualment en educació així com el desenvolupament des de zero d’una plataforma web destinada a l’aprenentatge.

L’objectiu d’aquest programa serà transmetre el coneixement de forma natura i còmode, així com promoure la interacció entre estudiants i professors. Això s’aconseguirà al crear un sistema on professors poden pujar el seu contingut establert que els estudiants han d’aprendre i crear tests per verificar-ne la seva comprensió. A més a més permetrà mostrar detallades estadístiques de cara als professors per poder seguir i veure com es desenvolupen els estudiants de forma ràpida i senzilla.

Resumen

La tecnología ha estado creciendo exponencialmente en las últimas décadas, con grandes empresas luchando por el liderazgo en creatividad e innovación, tirando adelante nuestra sociedad y creando los mejores productos para sus usuarios. Actualmente estamos empezando a ver este mismo desarrollo en educación.

A pesar de todos los nuevos descubrimientos en psicología, márketing y tecnología, no ha habido ninguna aplicación ni método pedagógico aplicado en las escuelas y universidades que revolucionara el sistema educativo establecido. Aun así, gobiernos e instituciones privadas están comenzando a darse cuenta que los ratios de finalización de estudios o la falta de interés de los estudiantes representan un gran problema para el desarrollo de un país y sociedad.

Este proyecto busca ser un análisis del estado actual de ciertas aplicaciones de renombre usadas en el campo de la educación, así como el desarrollo desde cero de una plataforma web destinada al aprendizaje.

El objetivo de esta plataforma será transmitir el conocimiento de una forma clara y natural, así como promover la interacción entre estudiantes y profesores. Para llevar a cabo este objetivo, se creará un sistema donde los profesores puedan subir su contenido fijo que

deben transmitir a los estudiantes y crear tests que les permita verificar la comprensión de ello. Además permitirá mostrar de forma detallada y sencilla estadísticas sobre el desarrollo y aprendizaje de los alumnos.

Index

1 Introduction

With the increasing use and development of ICT technologies, communication and information channels have grown and changed completely, and will continue to evolve and adapt, creating more useful and efficient channels.

Online chats, Blogs, Multimedia, Websites, Video calls, and many other technologies have been deployed and are available to a 78% of the world's population in developed countries [1].

The Internet then opened a new communication channel to easily reach out the population, so newspapers, films, books, dating, calls and many other exchanges are done also via internet. It was but a matter of time that education would be shared online.

1.1 Context

This topic will provide a brief insight about some types of technologies used in education.

a) Virtual Learning Environments

A Virtual Learning Environment (from now on 'VLE') is an e-learning set of teaching and learning tools based on the web, which provide an emulated virtualization of a traditional class and teacher-student interaction [2].

The principal components of a VLE package include curriculum mapping (breaking curriculum into sections that can be assigned and assessed), student tracking, online support for both teacher and student, online communication such as e-mails, forums, chats, etc. and external links to further resources related to the syllabus. The teacher has access to all students' content but with additional rights to create or modify content as well as tracking the student's performance.

A large list of companies and communities engaged into the creation of strong VLEs, being WebCT and **Blackboard** the top commercial ones [3] (both now owned by "Blackboard Inc.").

Google Inc. recently introduced "**Google Classroom**" [4], a VLE which is supposed to bring *"more teaching and less tech-ing"* into the classroom. It is currently in a closed beta phase but by September 2014 [5] it is expected to be available to all schools, turning it into the flagship application for the search giant's set of google apps for education [6].

b) Learning Management System

A Learning Management System (from now on 'LMS') is a software application designed for the administration of learning programs within an organization. It provides tracking, reporting and delivering of e-learning courses or training programs [7].

The most popular [8] LMS worldwide is **Moodle** [9] a community driven, free and open source software, followed by **Edmodo** [10], a free commercial application with a cleaner look that intends to bring the features of social networks into learning.

c) Massive Open Online Courses

A Massive Open Online Course (from now on 'MOOC') is *"a course of study made available over the Internet without charge to a very large number of people"* [11]. It provides a full virtual classroom with all the content and teacher interaction online (via web or mobile applications) and is expected to be scalable to any number of students who wish to join the course.

The top MOOC provider is the Californian Company **Coursera** [12], with more than 22 million of students [13] enrolled in their courses. For Europe, Spanish universities set the country as the top European nation in MOOC deployment [14], turning to **Miríada X** [15] as the most used platform.

d) Flipped Classroom

In a Flipped Classroom there is little to none live lecturing since the teacher provides these and other multimedia materials online. Class time is then transformed into hands-on work where the teacher can interact with the students (who will already have seen the online lecture and delivered the material), and engage into class-wide discussions and figure out new ways to solve problems.

With this hybrid class, students can learn the material at their own pace, when they can and as many times as they need. With this automation, the teacher has more time to focus on the students themselves and make sure that all of them are following up the course correctly.

1.2 Objectives

The goal of this project is to create a Blended Learning platform to support an offline course by adding its lectures and fixed content online while focusing live classes on students. This will allow teachers to select the best content in order to explain the subject (be it recorded by themselves or from external sources) while reusing and improving it easily. Furthermore, they will have real-time data, thus providing an opportunity to adapt the course, as well as easily detecting students having troubles following it.

The platform will be automatically structured according to the course's syllabus introduced by the teacher. For each topic and subtopic a page of information will be displayed, containing a video lecture, written content or any link and resource the teacher sees fit to add for the student's best understanding on the matter. Finally, after the student has gone through the content of the subject, a series of questions will be available for answering to assure its understanding.

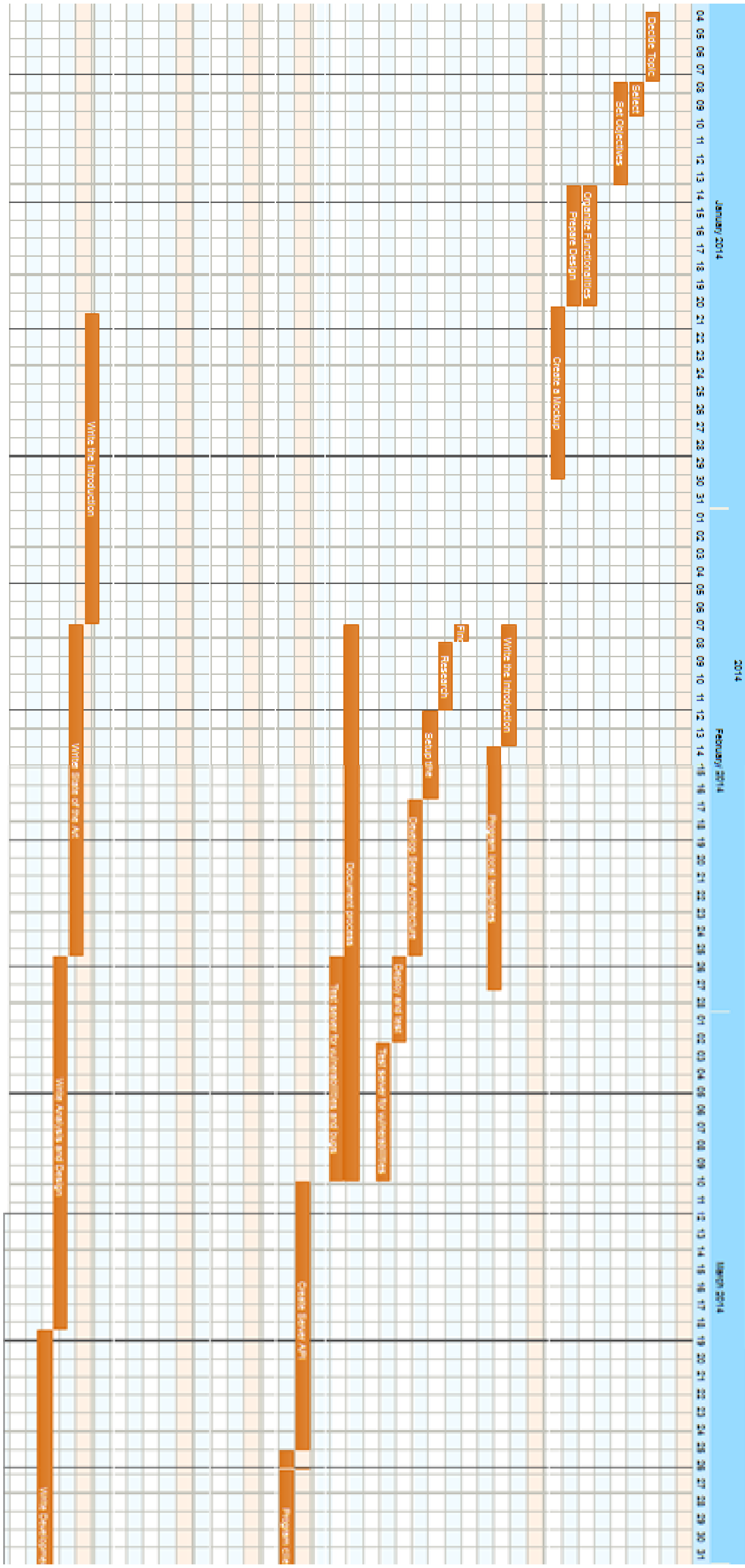
Besides being able to create and edit all the content, the teacher will also have access to statistics about the students' progress, which will be displayed in a simple and user-friendly way. The main goal of the project will be to optimize simplicity and user experience.

1.3 Scheduling

The goal of the project's scheduling is to visualize all the project's phases, along with each individual task and the time/resources assigned to each one. This will allow efficiently planning in advance and organizing the project optimally.

Below there is the Gantt diagram with the different stages the project will go through, with the estimated time assigned to each for its completion. It also displays the tasks' priority and which ones must be completed before starting another, as well as possible tasks that can be done in parallel.

Activity	Start	End	Days / Total	
Initial Tasks	08-01-14	03-02-14		36.0
Preliminars	08-01-14	17-01-14		12.0
Decide topic	08-01-14	11-01-14	4.0	
Select Manager	12-01-14	13-01-14	2.0	
Set Objectives	12-01-14	17-01-14	6.0	
Project Planning	18-01-14	03-02-14		24.0
Organize Functionalities	18-01-14	24-01-14	7.0	
Prepare Design	18-01-14	24-01-14	7.0	
Create a Mockup	25-01-14	03-02-14	10.0	
Development - Phase 1	12-02-14	15-03-14		98.0
Client Side	12-02-14	04-03-14		21.0
Find libraries and resources	12-02-14	18-02-14	7.0	
Program local templates	19-02-14	04-03-14	14.0	
Server Side	12-02-14	15-03-14		32.0
Find a computer	12-02-14	12-02-14	1.0	
Research Frameworks	13-02-14	16-02-14	4.0	
Setup the Environment	17-02-14	21-02-14	5.0	
Develop Server Architecture	22-02-14	02-03-14	9.0	
Deploy and test static tables	03-03-14	07-03-14	5.0	
Test server for vulnerabilities and bugs	08-03-14	15-03-14	8.0	
Documentation	12-02-14	15-03-14		45.0
Document process	12-02-14	15-03-14	32.0	
Draw diagrams and relations	03-03-14	15-03-14	13.0	
Development - Phase 2	16-03-14	13-04-14		29.0
Create Server API	16-03-14	30-03-14	15.0	
Program client for dynamic data input	31-03-14	13-04-14	14.0	
Testing	14-04-14	27-04-14		28.0
Find client UX problems	14-04-14	20-04-14	7.0	
Test for vulnerabilities on the Server	21-04-14	27-04-14	7.0	
Fix minor bugs	14-04-14	27-04-14	14.0	
Deployment	28-04-14	25-05-14		42.0
Try on Beta users	28-04-14	25-05-14	28.0	
Feedback report	05-05-14	05-05-14	1.0	
Fix problems	06-05-14	11-05-14	6.0	
Feedback report	12-05-14	12-05-14	1.0	
Fix problems	13-05-14	18-05-14	6.0	
Documentation	25-01-14	08-06-14		100.0
Write the Introduction	25-01-14	11-02-14	18.0	
Write State of the Art	12-02-14	02-03-14	19.0	
Write Analysis and Design	03-03-14	23-03-14	21.0	
Write Development	24-03-14	13-04-14	21.0	
Write Testing	19-05-14	25-05-14	7.0	
Write Conclusions and Future Work	26-05-14	08-06-14	14.0	


January 2014
2014
February 2014
March 2014
Decide Topic
Select
Set Objectives
Organize Functionalities
Prepare Design
Create a Mockup
Write the Introduction
Find
Research
Write the Introduction
Program local templates
Develop Server Architecture
Document process
Write State of the Art
Deploy and test
Test server for vulnerabilities and bugs
Test server for vulnerabilities
Write Analysis and Design
Create Server API


2014
April 2014
May 2014
nt for dynamic data input
Find client UX problems
Fix minor bugs
Test for vulnerabilities on
Try on Beta users
Fix problems
Fix problems
Write Testing
Write Conclusions and Future Work

2. State of the Art

An optimal use of technology in education can boost the quality of teaching and the overall student experience. It can help teachers explain complex subjects using simple visual stimulus, interact with the students in a more efficient way and focus their attention in the students as a result of simplifying the teaching methods. In this chapter there will be some interesting and actual appliances of ICT in education.

2.1 Learning Platforms

This topic will provide an analysis of three successful uses of ICT regarding education.

a) Moodle

Moodle is currently the reference when talking about e-learning software. It is used worldwide, with more than 71 million users in 235 countries [16].

The educator and researcher Colin Matheson was asked the reason why to use Moodle when there are many other tools that provide similar services with a better usability. His answer [17] was a reflect of Moodle's biggest feature, flexibility: *"Is Moodle the best blog tool, the best forum tool, the best wiki tool, the best quiz tool, the best survey tool, the best social networking tool? No. But it is the only web app I have found that has all of these in one spot."* Even though Moodle offers all those services in a single platform, it is far from being the perfect e-learning software. As Martin Dougiamas, CEO and Founder of Moodle wrote in Quora [18]: *"I would love it if we had loads of UX specialists at our disposal full-time but we just don't. Remember Moodle is open source and most people pay us nothing for it"*. That is one of the key concepts in understanding why Moodle is so successful and why it still has so much space to grow. Moodle is a free and open source software platform; everyone can have access to it and "hack it" as they please. However, the typical use of Moodle is to download the standardized stable builds [19] that the company provides and deploy them, as they are, in a server. Still, Moodle is fully customizable and there are many plugins and additional content that is constantly being created by third-party developers [20] which can greatly increase the student's experience while using the platform.

When Moodle is deployed in the school's server, students can log in using their own school's account that has already been registered for them. That is a big advantage for the software compared to other external platforms that force users to sign up on their own private server, resulting in a set of different accounts for each different platform the students decided to use. However, there is a large room for improvement regarding the user profile, because with a registration and user management system being already built, Moodle is not using it to its full potential to expand the platform with additional social features or setting a more detailed statistics display about the students (more about this topic will be explained below).

One of Moodle's biggest problems is design. The company reported [18] that due to the platform's complexity (being it considered above other software's like Facebook, Drupal or Blogger) it was crucial that they focused on *"areas such as stability, performance, backend functionality, education features, accessibility, standards"* and leave User Experience (from now on UX) as a minor priority. This decision resulted in the creation of a strong and stable platform that even though it will mostly work with high efficiency, it doesn't provide the best UX that the software has to offer. In Figure 1 there is a sample of Moodle's purely functional and structural interface while in Figure 2 the VLE has adopted a more "modern" style with a flat design approach, which shares many similarities to Facebook's GUI, making it more appealing to students.

Figure 1 - Moodle's interface sample from a course instructed by "Universitat Pompeu Fabra"

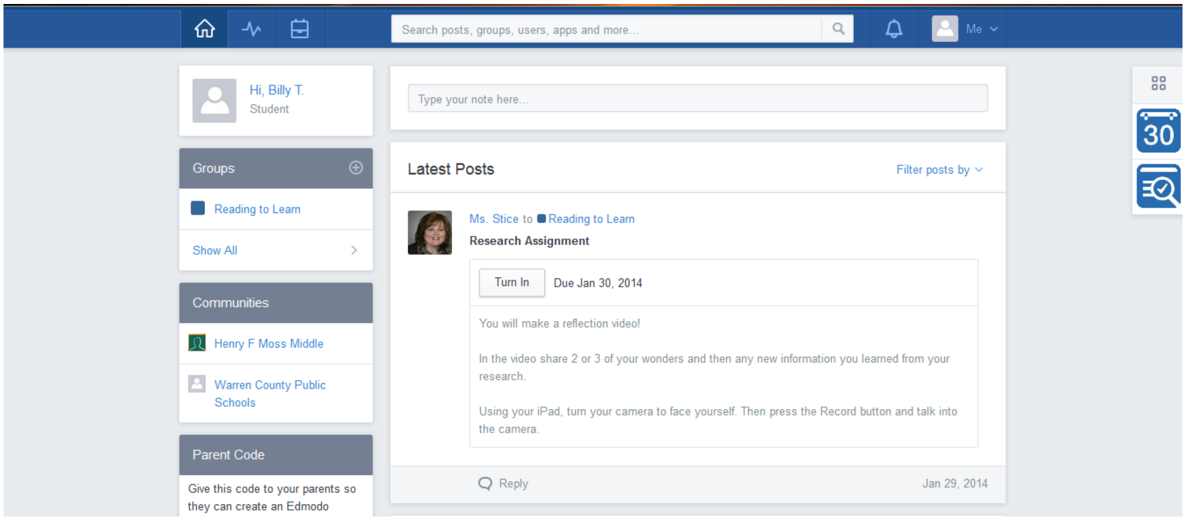


Figure 2 - Edmodo's homepage sample from a student's perspective

Finally, a last point to review about Moodle is the statistics it displays to teachers. When using the platform in universities or other educational organizations, it is crucial for the educators to have real time information about their students' progress. Even though Moodle provides access to the content and stores user's results and uploads, it does not parse this information in a friendly and quick access way to allow teachers to get a look into what is going on in their course and how are their students performing.

Analytics

With all the data Moodle collects in the server's database about student's grades, assignments, dates, uploads, etc. there is not a strong statistics system where teachers can see this information parsed in a simple and practical way thus allowing them to quickly see what goes on in the course and how are the students performing. The current system does have analytics, as it is displayed below in Figures 3 and 4, however the relevance and quality of the information displayed in it is relatively poor compared with all the possibilities it could offer. Teachers could appreciate a simple and easy to find display of important information that would allow them to have a general idea of the user's performance, with the possibility to dig deeper and have more detailed and specific analytics about each student and or assignments

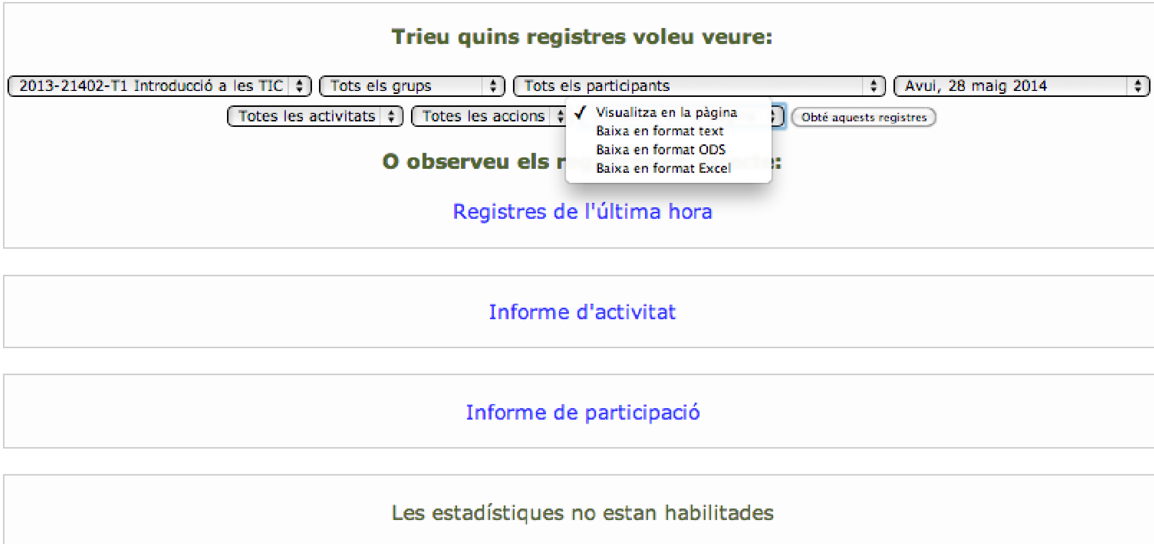


Figure 3 - Accessing Moodle's analytics system through a registers filter

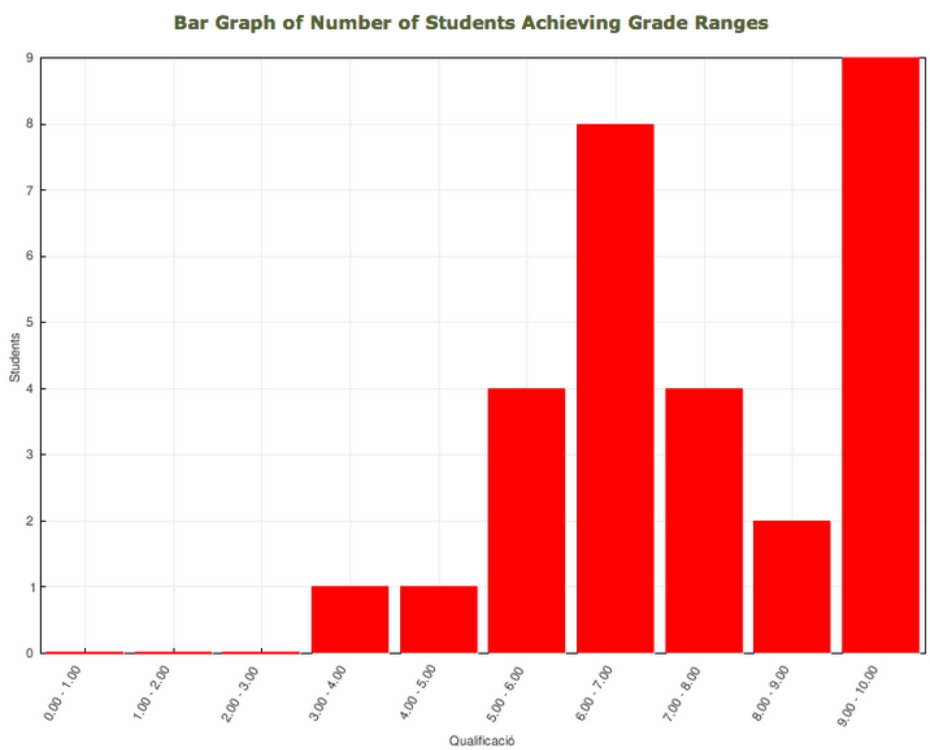


Figure 4- Moodle's sample graph featuring the students' grades in an assignment

b) MOOCs

The term MOOC was coined in 2008 by Dave Cornier and Bryan Alexander after the course “Connectivism and Connective Knowledge” (also known as CCK08) led by George Siemens and Stephen Downes which attracted 2.200 participants from all around the globe [21].

Since then a lot of companies started offering all kind of MOOCs, being three the top pioneers in the MOOC field [22]:

I) **Coursera:** The for-profit company born in Stanford defines themselves as “*an education platform that partners with top universities and organizations worldwide, to offer courses online for anyone to take, for free”* [23]. On the 23rd of October, 2013 Coursera announced a total of 107 university partnerships [24].

II) **edX:** Founded by the M.I.T. and Harvard, is an open source platform that “*goes beyond offering courses and content”* and aims to also understand *“how students learn, how technology can transform learning, and the ways teachers teach on campus and beyond*” [25].

III) **Udacity:** Born in Stanford (though not affiliated with), Udacitiy’s founders, both teachers at the university, launched their courser online for free and found out that over 160.000 students in more than 190 countries enrolled on it. After that they decided to launch the MOOC Company. Their mission is to “*bring accessible, affordable, engaging, and highly effective higher education to the world”* [26].

MOOCs Structure

All MOOCs share a similar structure which comes from as typical university’s course organization but adapted to an online and massive scale.

I) Syllabus

The Syllabus is structured in weeks as can be seen in Figure 5. For each week, a different topic is discussed. Larger topics can be divided in two weeks. Each topic has a set of lectures where the teacher explains the different sections of the content.

The course also contains an initial short introductory phase where the teachers explain what the course is about and how it will be done, as well as referencing to the resources available to the students. It often includes an initial survey to see the current student's level on the subject in order to focus and adapt the course efficiently to the "classroom" needs.

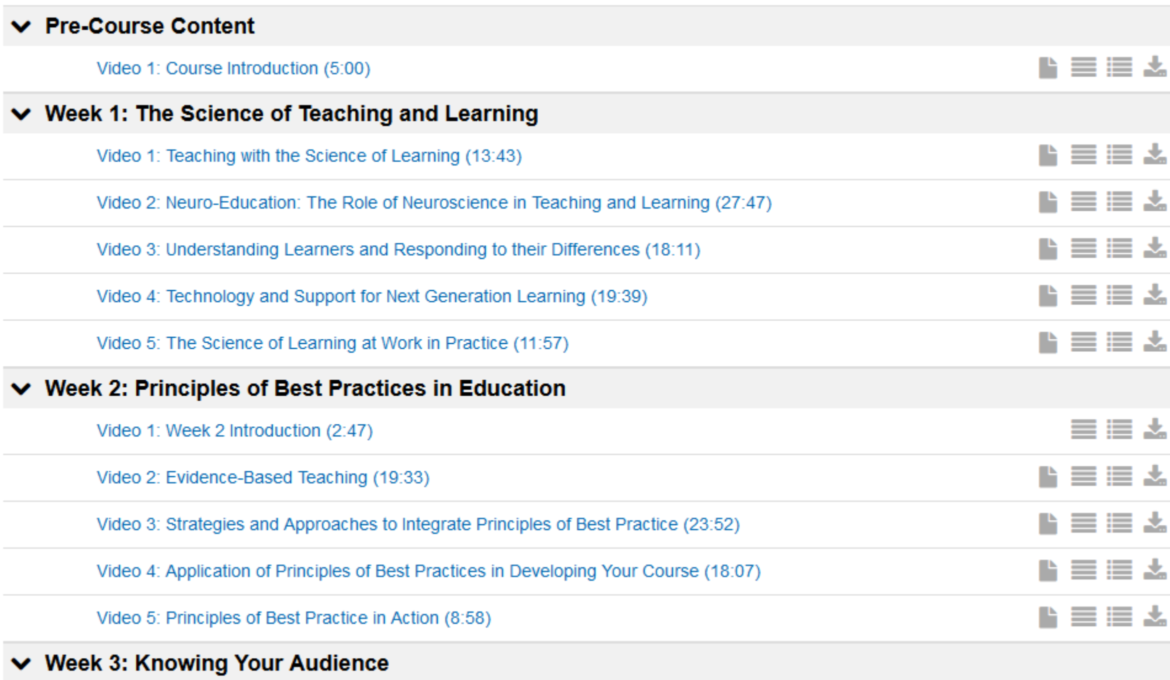


Figure 5 - Course Syllabus from the MOOC "University Teaching 101" offered through coursera.org

II) Lectures

Each lecture has an average length ranging between 8 and 12 minutes, being 30 the maximum amount of time recommended for a lecture. That way teachers force themselves to optimize the content of the courses, including only key information and avoiding redundant and unnecessary material, since the students can review the lectures as many times as they wish. Lectures typically include visual content such as graphs, videos, comics and notes as well as the teacher explaining all the material (examples of that can be seen on

Figure 6). It is also common to see end-of-week special lectures where teachers interview renowned professionals on the subject to tell their experiences and points of view to students

The lectures also contain subtitles in different languages so students from different countries can turn on the captions on their own language. Voluntary student groups can be formed at the beginning of the course to help translate its content into different languages.

The lectures and material (presentations, articles, etc.) are available for students to download.

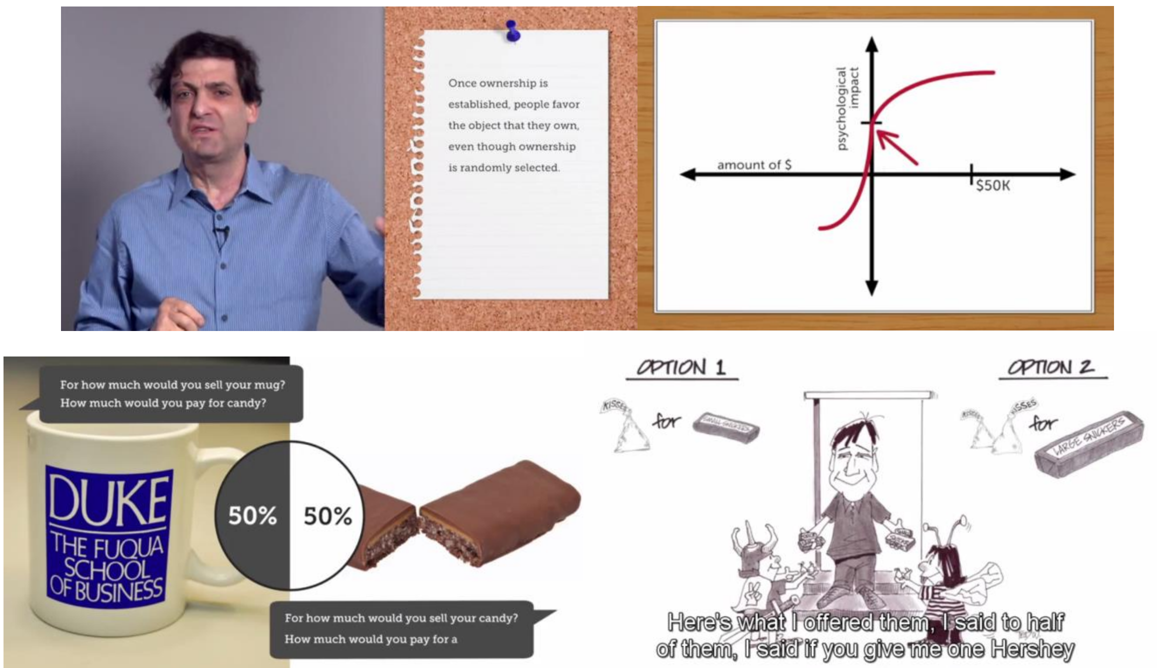


Figure 6 - Captures from Dan Ariely's course "A Beginner's Guide to Irrational Behavior" from Duke's University

III) Resources

Discussion forum: In addition the lectures, students can also learn and express their opinions on the course's forum. Any question or disagreement about the lectures or the course can be pointed out there and start conversations with all the students enrolled on the course as well as the teachers.

Readings: Every course has a list of recommended and/or required material, containing books, articles, talks, etc., to further increase the knowledge of the subject.

Meetups: Students who wish to meet face to face with other fellow students can organize a Meetup [27] with their local peers. This is encouraged by the MOOC and course providers but not organized nor officially supported by them.

Hangouts: Teachers usually organize hangouts [28] periodically to solve doubts, explain topics further and connect with the students.

IV) Grading

Quizzes: There are a set of quizzes along the course so the student can see if he understands the material correctly. With it teachers also have real-time data about how the course is going in order to adapt it, either by lowering the difficulty or adding extra materials and content.

Peer Assignments: Students wishing to have a grade must deliver a set of exercises told by the teacher wish will be reviewed by their fellow students. For smaller MOOCs the teachers are usually in charge of doing that.

Exams: The greater percentage of the grade comes with an old-fashioned exam to be delivered before a specified date where the student has to apply all the content he has learned during the course. Since the student has a long time to deliver it and all the resources available on his computer, the exams are designed to truly test the understanding and appliance of the subject rather than if the student succeeded memorizing the content

MOOCs by the Numbers [29]

Scope: The major MOOC providers (Coursera, edX and Udacity) add up more than 8 million students from 225+ countries, 1200+ courses and 150+ university partnerships.

Courses: The average course takes 8 to 10 weeks to complete, has a 7% completion rate (with that the amount of students who were graded and got a completion statement, which is optional) and questions during the course are answered either by teachers or fellow peers in around 11 minutes.

More detailed information about MOOC's reach is displayed on Figure 7.

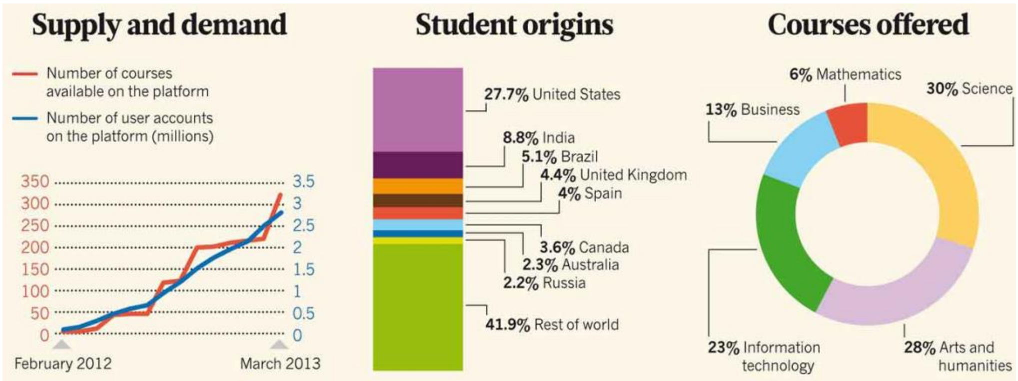


Figure 7 - MOOC's scope [30]

An Alternative to University

As seen previously, MOOCS have been growing at incredibly fast rates since their large-scale debut in 2012, which has even been called the "Year of the MOOC" by the New York Times [31] and started getting an incredibly amount of media coverage since then. Now in 2014 this explosion of attention has ceased [32] but the number of MOOCs continues to grow exponentially [33].

The 15th of January, 2013 Udacity announced a partnership with "San Jose State University" [34] to provide three courses at an affordable price and for college credit, turning that moment into the first time a college course could be done purely online through a MOOC [35]. Coursera also has available a "Verified Certificate" [36] which consists of a certificate signed by the course's professors ratifying that you completed and achieved the course. Its use though it's just to *"List it on your résumé/CV and include it on social media/career profiles"*. The American Council of Education [37] is starting to review MOOCs to officially allow awarding college credit upon completion [38].

With all this high quality material from elite universities freely available on the web for everyone, with the comfort of doing it at your own pace and receiving university credits for it, The Economist [39] poses one obvious question "Will MOOCs kill university degrees" [40]?

c) Khan Academy

Khan Academy [41] is an organization that defines itself as a *"not-for-profit with the goal of changing education for the better by providing a free world-class education for anyone anywhere"* [42]. More concretely, Khan Academy is an e-learning US-based company that provides a virtual classroom for students around the world. Currently it has more than 10 million users around the world with more than 5.000 courses [43].

In 2004, Salman Khan, founder of Khan Academy, started teaching his 12-year-old cousin via the Internet to help her with math [43]. To do that, he created simple visual explanatory videos that he uploaded to YouTube and later sent to her. People started seeing those videos and used them to help their own children having the same troubles. Noticing the success of his lectures, he decided to found "Khan Academy" in 2006. It first started as a simple library of a few YouTube videos explaining some topics such as arithmetic, geometry and algebra but it also started adding other topics like history or biology. Later on, the platform was turned into a "*fully featured, interactive educational system"* [44]. With the support of Bill Gates in 2010 [45], Khan Academy started to be introduced in K-12 schools [46] to provide students with a strong educational environment where they could learn the topics at their own pace and do the exercises as many times as they needed, something that Salman Khan saw as a key factor in learning [47] which formal education wasn't paying attention to.

Figures 8, 9 and 10 below show how Khan Academy structures its information in a simple and visually appealing design to attract the students. The platform also uses Gamification (represented for example in Figure 8 with the badges) to reward students for their successes and encourage them to keep learning.

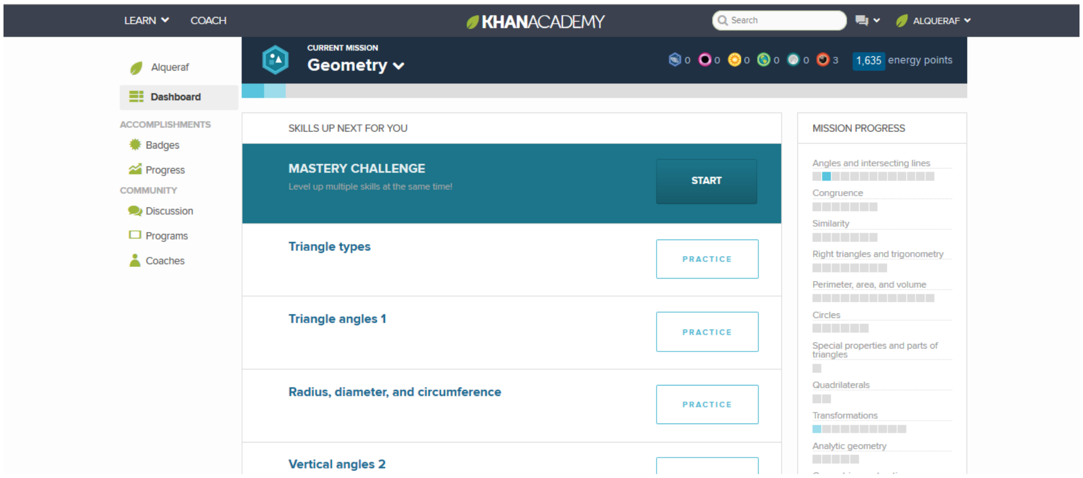


Figure 8 - Khan Academy student's homepage sample

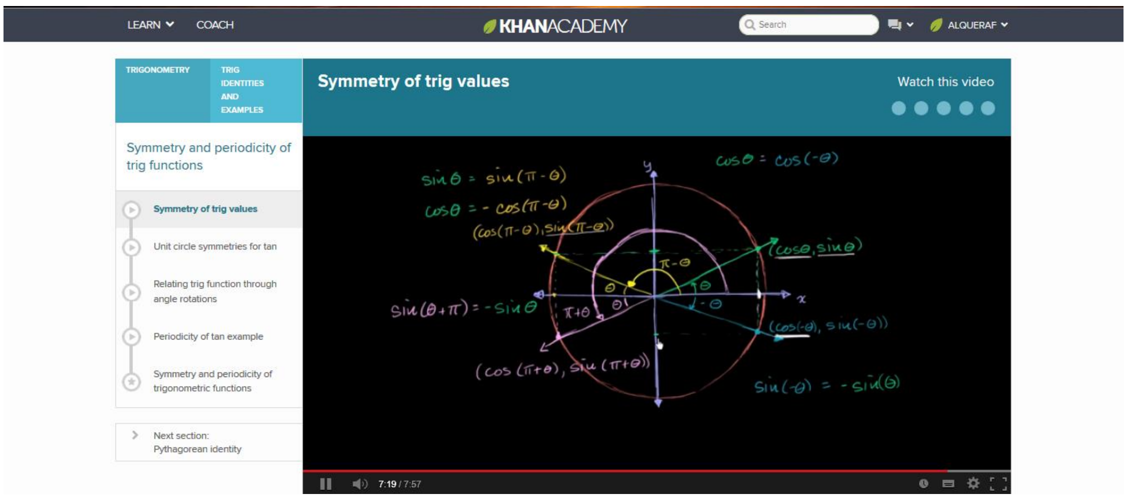


Figure 9 - Sample video lesson from Trigonometry

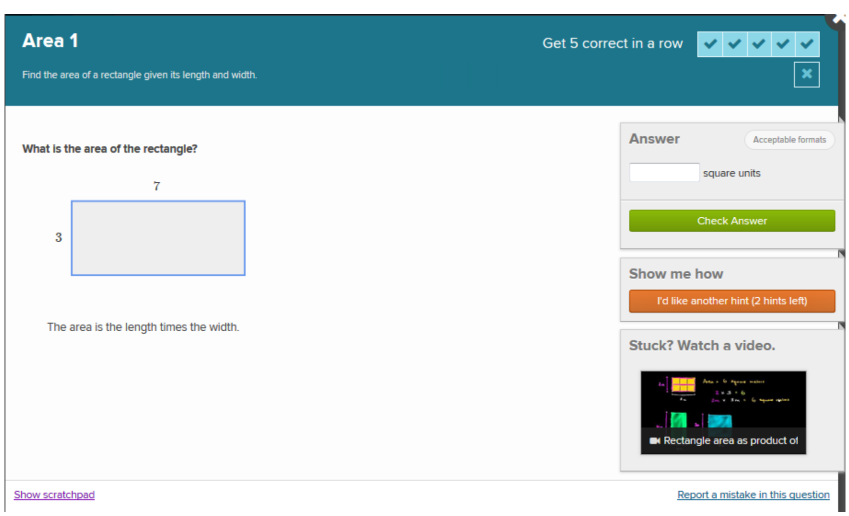


Figure 10 - Quiz screen from the topic "Area 1" in Khan Academy

Starting November 2010, Khan Academy was used for the first time in a school in Los Altos [48], CA, United States. The objective of this implementation, was to let students learn at their own pace, with better communication channels like videos, games and other simple visual stimulus and as a result have more free time for the teacher, *"allowing classrooms to be more interactive and experiential"* as Salman Khan states in the Washington Post blog [49]. Teachers then could register themselves as "coach" and create a virtual classroom where they could follow the student's progress. However, the content of the site cannot be changed, so teachers have to rely on the material provided by the platform, which has sometimes put to doubt by educators [50].

Figure 11 shows how Khan Academy implements its analytics system for teachers, displaying the most relevant information (students' grades and completion) in a simple grid view, using colors related to assignments and grades, to allow the teacher to have a general idea how the students are performing with just a quick look.

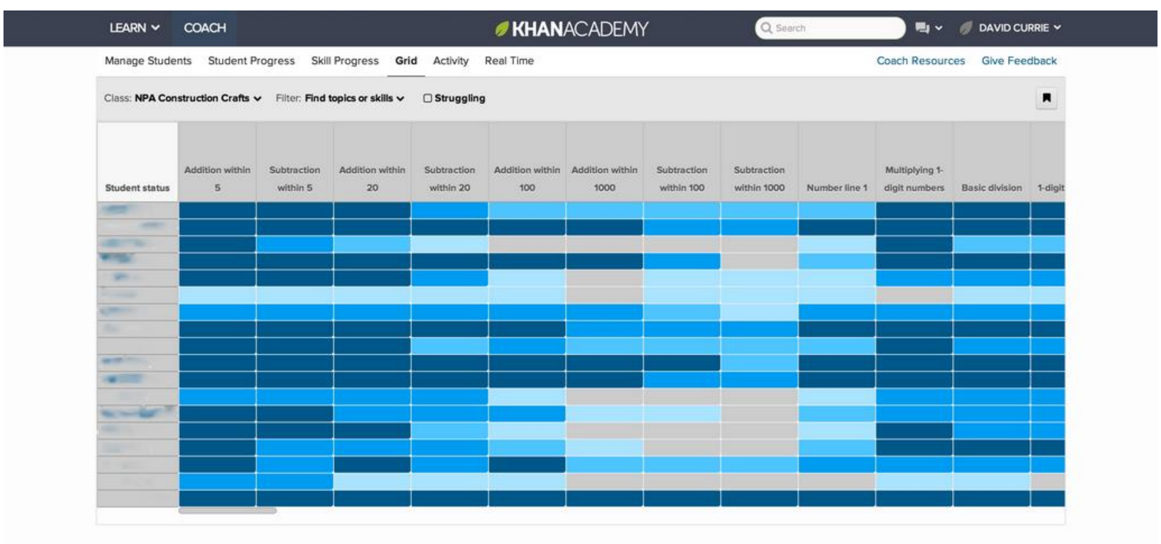


Figure 11 - Example of a progress grid of the students in a "Khan Academy class"

If Khan Academy continues to be implemented in more schools, it will create an interesting change of mindset, splitting education into the acquisition of knowledge, through a highly efficient and optimized global repository of material, and the real-life appliance of this content through dynamic and student-engaging activities.

2.2 The Learning Revolution

Wired published an article about the "Learning Revolution" [51] claiming that *"there is indeed a revolution going on. But it is not about education systems. It is about learning. It is people taking learning into their own hands"*.

The number of startups providing education through technology is rising, as is the amount of funding and corporate interest linked to it. These companies provide a quality service without dealing with governments, focusing solely on users, to provide the best service for their customers. The graph in Figure 12 accounts for the amount of venture capital being invested into Ed-Tech startups in the United States of America which has been steadily increasing for the past 5 years [52].

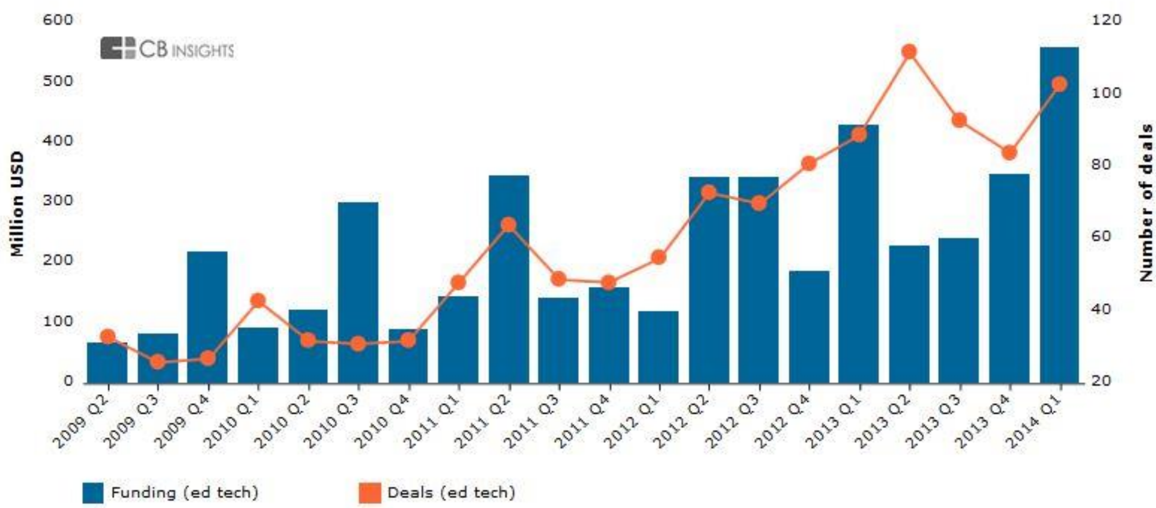


Figure 12 - Ed-Tech investments growth

The article poses an example about traditional education which states that *"Formal education is more about passing a French exam than about being able to actually talk to a French person"*. People are looking for knowledge [53], and MOOCs are providing it in an innovative and more efficient way. As Henry C. Lucas, professor at the University of Maryland stated [54] *"If higher education institutions embrace the status quo, they will no longer be in control of their own fate"*. Institutions still using obsolete traditional models may face a hard blow in the near future by having to compete with companies offering higher quality content, available at the student request, and for free.

2.3 Conclusion and Personal Motivations

In this chapter there have been some analysis and statistics showing how education is starting to attract the attention of the private sector and independent students, creating a parallel line of education that is not being controlled by public administrations and does not have pre-established fixed rules, compared to the formal one. This opens a wide path for innovation, where schools and universities can use these platforms for teaching, and research on its strengths and failures to improve the current system. It will also allow any person anywhere in the world with a device connected to the internet to have access to an education that otherwise may not have been possible due to location or personal circumstances.

It is important to start using all the technology available to optimize the quality of teaching, since the recipients of this education, the students, will be the ones who will continue to create and improve the current systems. It is then, in my opinion, our duty to offer the best education that our knowledge of psychology, pedagogy, neurology and technology has to offer and by refusing to innovate or sticking up to old and traditional methods we are not accomplishing it. I have therefore developed a platform which intends to bring technology and human interaction into classrooms by uploading all the non-interactive content online, to be learnt and reviewed by students as they need, and leave time in class to focus on the most important bases of learning: experiment and teacher-student interaction.

3 Analysis and Design

This chapter will go through the requirements and specifications of the platform as well as analyzing the structure and define user cases.

3.1 Requirements

The requirements will be divided between functional and non-functional, where functional requirements stand for the tasks the system will be able to perform while non-functional requirements are related to the overall qualities the platform needs to achieve its functions.

a) Non Functional Requirements (RNF)

ID	Name	Description	Relevance
RNF001	Roles	Users will be initially divided into two roles, TEACHER or STUDENT. Each type will have different rights of access to data.	★★☆

ID	Name	Description	Relevance
RNF002	Syllabus	Course topics must be divided into a hierarchical structure for better accessibility and organization.	★★☆

ID	Name	Description	Relevance

RNF003	Flat Design	Interface design must meet the current trends for users to feel more comfortable and have a higher appealing to content.	★★☆

ID	Name	Description	Relevance
RNF004	UX	The platform must be built with User Experience as a top priority to allow better access to content as well as attracting users.	★★★

ID	Name	Description	Relevance
RNF005	Error handling	A strong system of internal analytics and error report must be built to assure a high quality backend.	★★★

ID	Name	Description	Relevance
RNF006	Server Framework	The server will be built in python using the rest “Django”	★☆☆

ID	Name	Description	Relevance
RNF007	Client Application	The client application will be built in “HTML5”, using “Ajax” to communicate with the server.	★☆☆

ID	Name	Description	Relevance
RNF008	API	The Server will interact with the application as an API, making requests to the server. That way new applications can be created using the data stored in the server but allowing to displayed and parsed differently.	★☆☆

b) Functional Requirements

ID	Name	Description	Relevance
RF001	Authentication	The Server must manage users in order to display analytics and track progress.	★★★

ID	Name	Description	Relevance
RF002	Content Creation	Teacher roles must be able to create, edit and delete the content.	★★★

ID	Name	Description	Relevance
RF003	Analytics	The platform must contain an Analytics section for Teachers to evaluate the student’s progress.	★★★

3.2 Use Cases

In this chapter there will be an analysis of the use cases, described as a sequence of events that users perform in order to achieve a goal. It will reveal how users use the system and the flow they follow.

Below are some user cases also displayed graphically in Figures 13, 14, 15 and 16:

<table>
<tr><td>Name/ID</td><td colspan="2">UC001 - Login.</td></tr>
<tr><td>Description</td><td colspan="2">The user performs a login with its account.</td></tr>
<tr><td>Requirements</td><td colspan="2">RF001.</td></tr>
<tr><td>Previous Conditions</td><td colspan="2">The account must have been created by the administrator.</td></tr>
<tr><td>Flow</td><td colspan="2"></td></tr>
<tr><td colspan="2">Actor</td><td>System</td></tr>
<tr><td colspan="2">1 - The user opens the platform in the browser.
3 - The user authenticates with its account.</td><td>2 - The system detects the user is not logged in so it displays the login webpage.</td></tr>
</table>

Success Post conditions	The system redirects the user to its main page.
Failure Post conditions	The system displays an “incorrect login” message.
Extension points	The user can view the courses and its content.

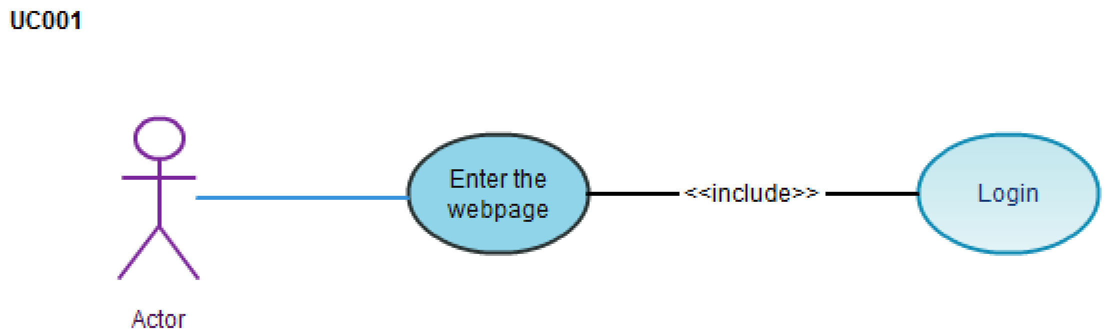


Figure 13 - UC001

Name/ID	UC002 - View the material
Description	The student selects what topic wants to study and reviews the content
Requirements	RNF002.
Previous Conditions	The user must have performed a login in the website.
Flow	
Actor	**System**
2 - The student selects what topic wants to learn from the list available. 4 - The student can view the videos and content displayed.	1 - The system retrieves the courses assigned to that user. 3 - The system loads the content for that specific topic.
Post conditions	The user can view the content template
Extension points	The user can access the quizzes

Name/ID	UC003 - Take the tests
Description	The student solves the questionnaires assigned to a topic
Requirements	RF001.

Previous Conditions	The user must have performed a login in the website and selected a topic.
Flow	
Actor	**System**
2 - The student solves a question. 4 - The student can check if the answer selected was correct or wrong.	1 - The system retrieves the questions assigned to that topic. 3 - The system obtains the answer selected, checks if it is correct and stores it in the system to use it for the analytics.
Post conditions	A right or wrong message is displayed.
Extension points	The user can solve the next question.

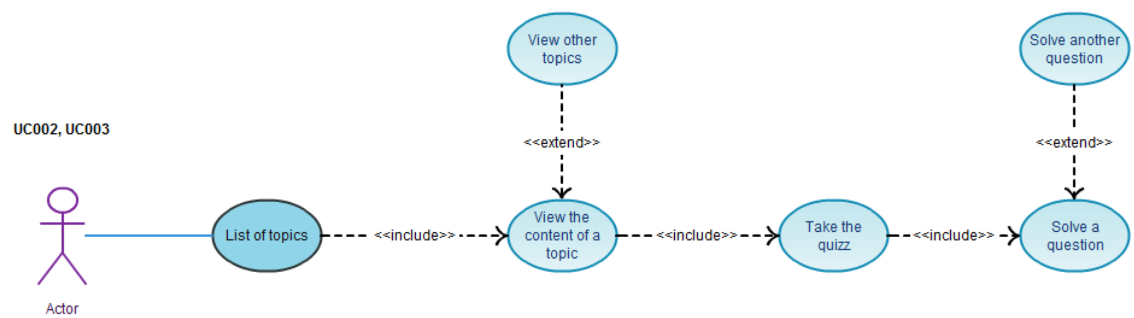


Figure 14 - UC002 and UC003

Name/ID	UC004 - Analytics
Description	The teacher role can view the analytics of the course
Requirements	RF001, RNF001
Previous Conditions	The user must have the teacher role and the students must have solved some quizzes.
Flow	
Actor	**System**
3 - The teacher can view the analytics available for the course.	1 - The system retrieves the all the questions solved by the students. 2 - The client application parses the information and displays graphs to show the data.

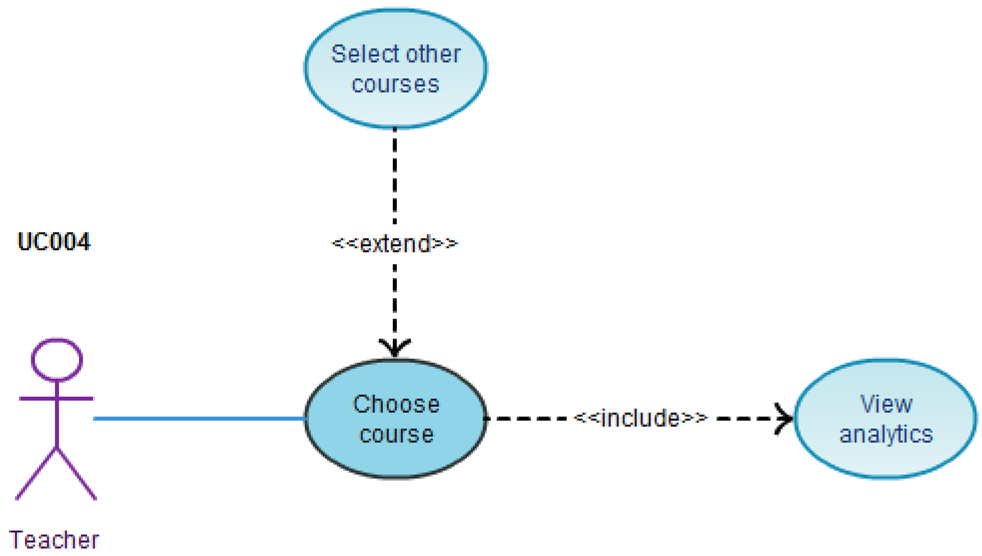


Figure 15 - UC004

Name/ID	UC005 - Add content
Description	The teacher role can add or edit content
Requirements	RF001, RF002, RNF001
Previous Conditions	The user must have the teacher role.
Flow	
Actor	**System**
2 - The teacher inputs or modifies the content for a topic or question.	1 - The system retrieves the current content and structure of the course. 3 - The system checks for errors and stores the content in the database.
Post conditions	A success message is displayed and the new content appears in the admin panel.
Extension points	Students can now access the new content from the web platform.

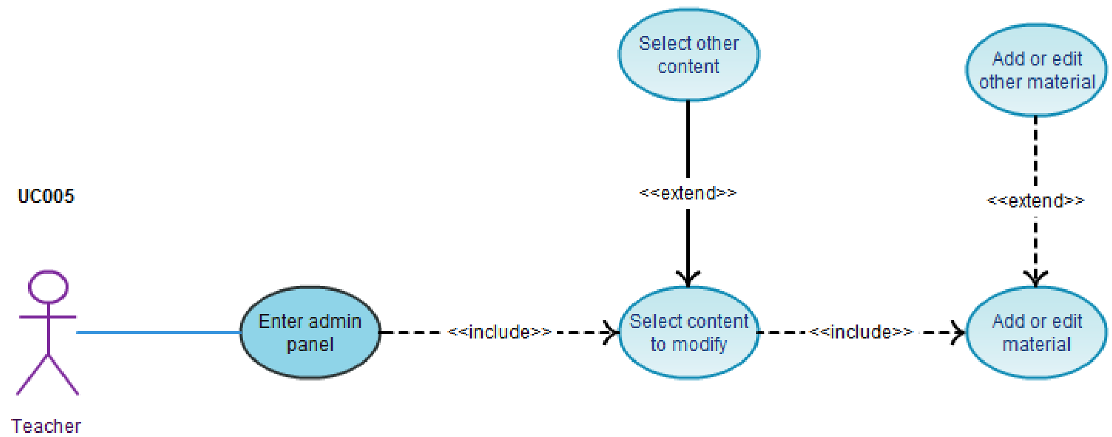


Figure 16 - UC005

3.3 Design

In this section a detailed analysis with diagrams will be done to explain the relations between all the elements the platform holds.

Sequence Diagram

In Figure 17 below there is the sequence diagram for the basic functionalities a user can use. The sequence flow includes a login/authentication. Next the client app must retrieve all the courses and for each topic/subtopic selected by the user a request for its content will be made to the server to prevent initial congestions due to the huge possible amount of information. Furthermore, for each question answered, the client app will ask for verification to the server to prevent any leaks of information because of important data (the answer) being stored locally. Finally, for analytics the teacher role will request the data for the course and it will be parsed by the client app into visually appealing content.

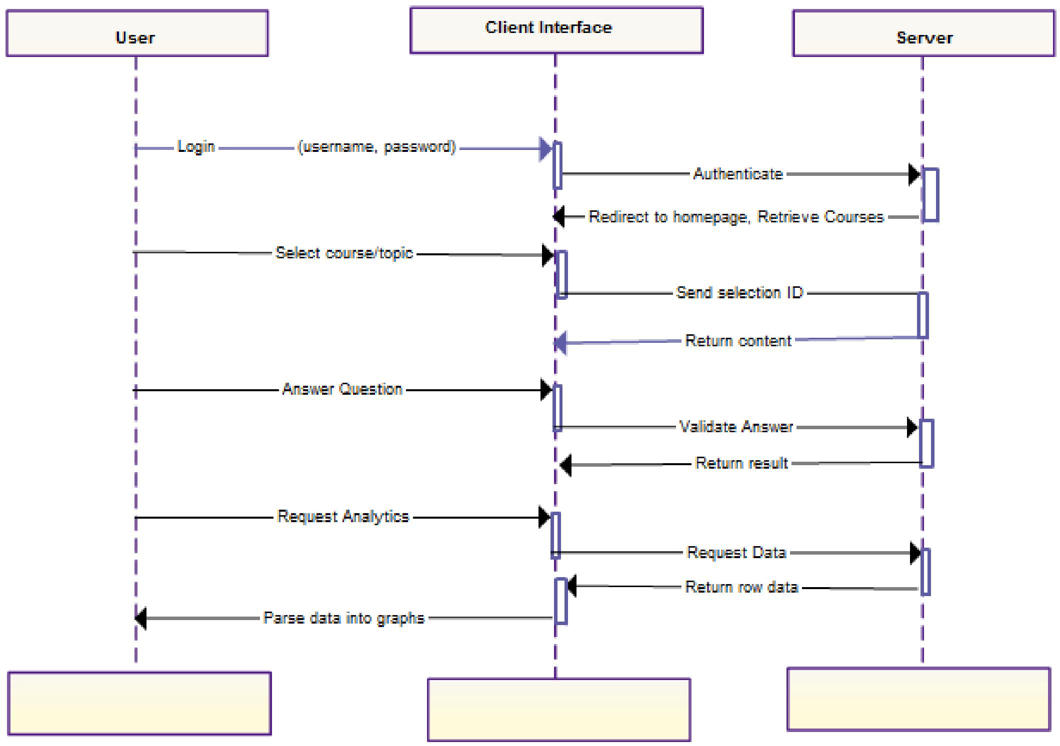


Figure 17 - Sequence Diagram

Entity Relationship Diagram

Figure 18 is an Entity Relationship Diagram which displays how is the server's database structured, with what elements and the relations between them. As displayed below, each element of the courses' data has an identifier, a title/name and the content. For the users group, each one will have a role (currently student or teacher, not taking into account the admin/superuser) a name to identify them and an optional description with ready/write access only from the teacher's role that is not currently implemented in the client application but may be helpful for the teachers to keep notes about their students.

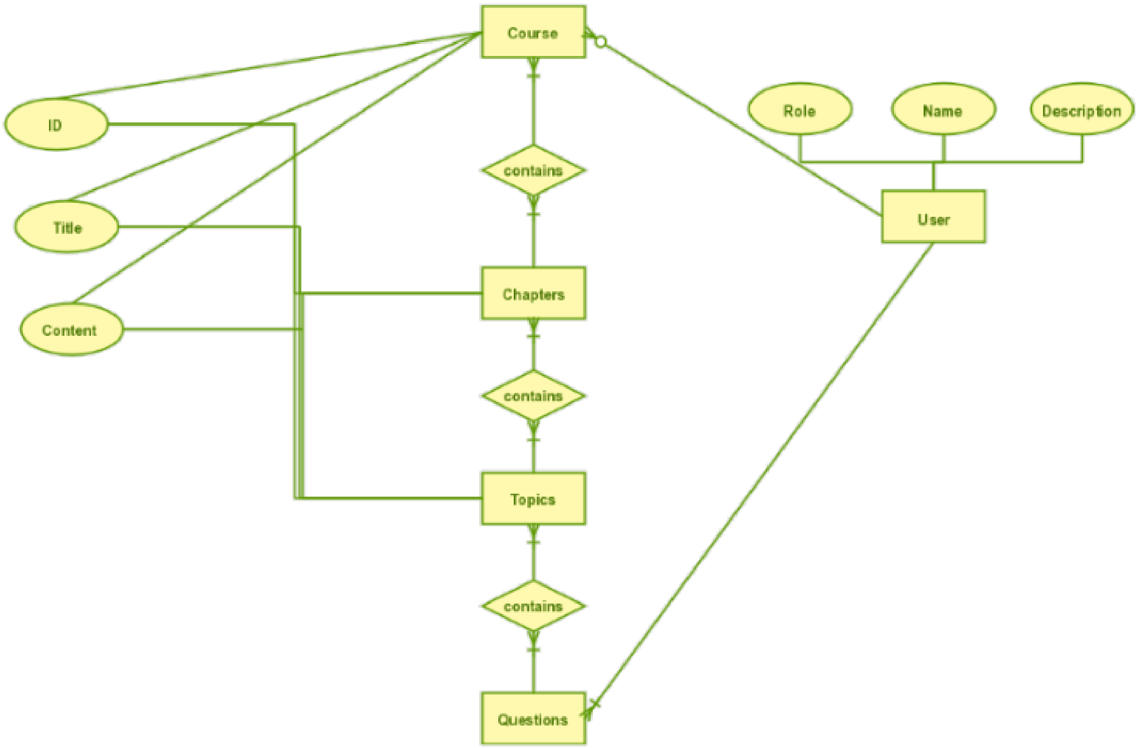


Figure 18 - Database Entity Relationship Diagram

Class Diagram

Figure 19 displays a detailed Diagram of all the classes in the Server that holds the information and structures the platform. There is a clear hierarchical structure regarding a course, where each element may have 0 or many sub elements (e.g. a topic may have 0 or many subtopics and those may have 0 or many questions) as decided by the content's creator.

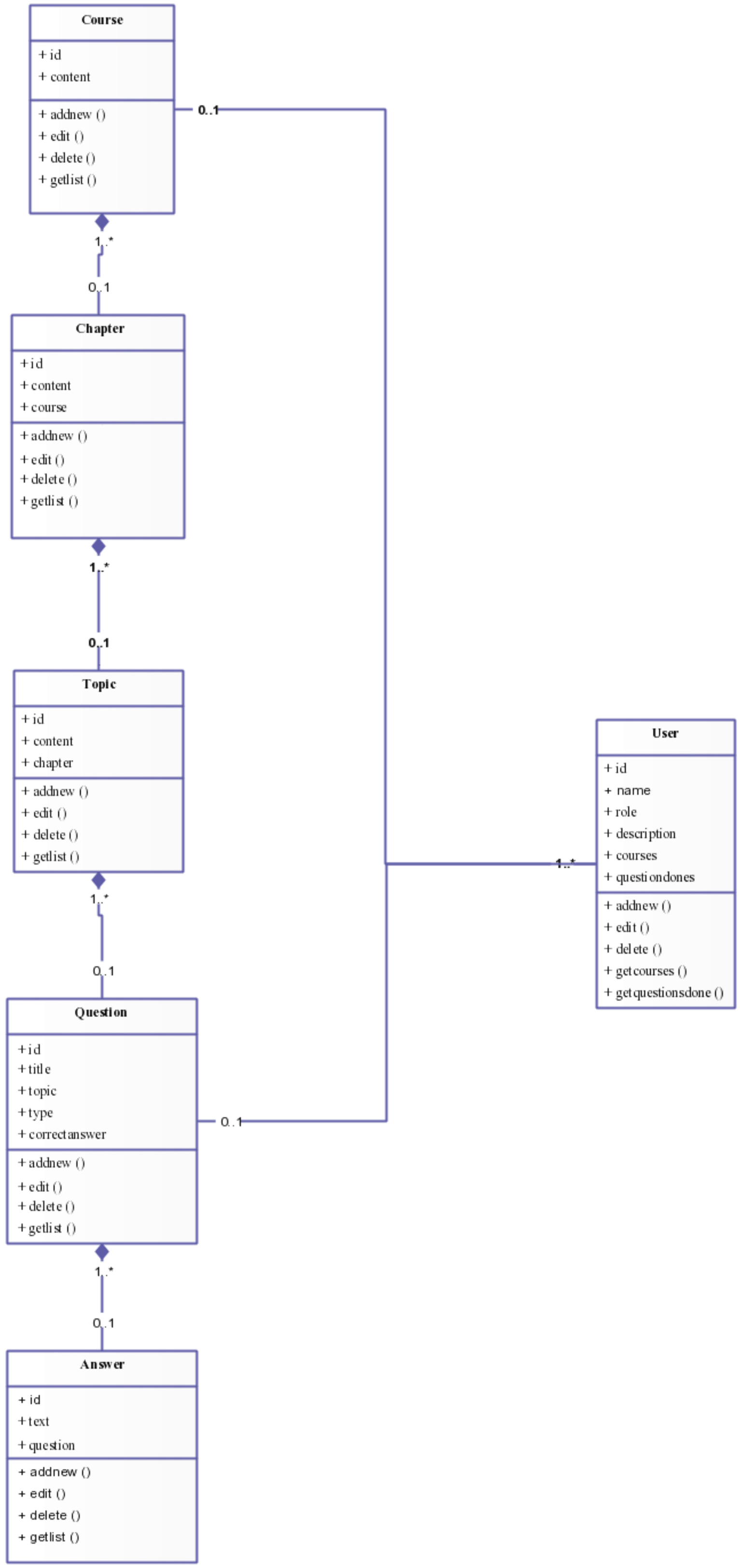


Figure 19 - Server's Class Diagram

3.4 Graphical User Interface

Once the software structure and architecture has been established, the next step is to create a Graphical User Interface (from now on GUI) where users can interact with the platform in a comfortable and easy way.

Main Screen

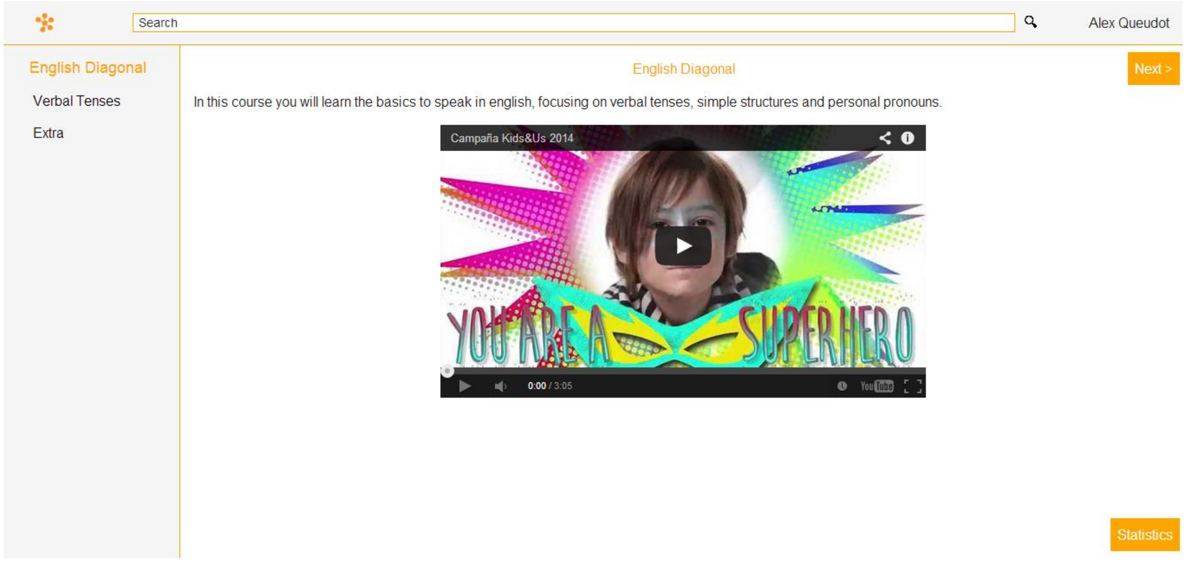


Figure 20 - "Home" screen

Figure 20 shows the platform's main screen, from now on "home". The application's main color is orange, using also white black and grey for visual contrasts and text. The design tries to be as minimalistic as possible, in terms of UX as well as visual stimulus. Users should be able to navigate comfortable and naturally around the app with visual updates indicating the next step to take. Only the necessary information shall be displayed on screen. Colors and shapes are completely flat to prevent distracting the student and overloading the interface, as it follows the current trend of flat and square designs that most users are used to and feel comfortable with.

Regarding in-app navigation and structure, there are three main parts to take into account.

I) Top bar: This section serves as a banner for extra elements such as the logo, user settings and possible logout and searches.

II) Left navigation: Every user has assigned a number of courses that will automatically appear on the left once authenticated. Students can navigate through the course by clicking on each section. Topics/subtopics will expand with an indent to differentiate them and the current selection will be highlighted in orange.

III) Main section: This element holds all the content the platform offers. For every selection the user takes in the navigation bar, its data will be displayed there as well as the quizzes (Figure 21) and statistics (Figure 22) if the user has the teacher role. If the user has student role the statistics button will not be displayed.

Regarding the quizzes, the platform currently supports only multiple choice quiz. However both the server and client app exchange a parameter "type" which defines how the question can be answered. Therefore, the platform is ready for a different variety of questions which may include word completion, single word answer, etc. More information regarding this matter will be given in the Future work chapter.

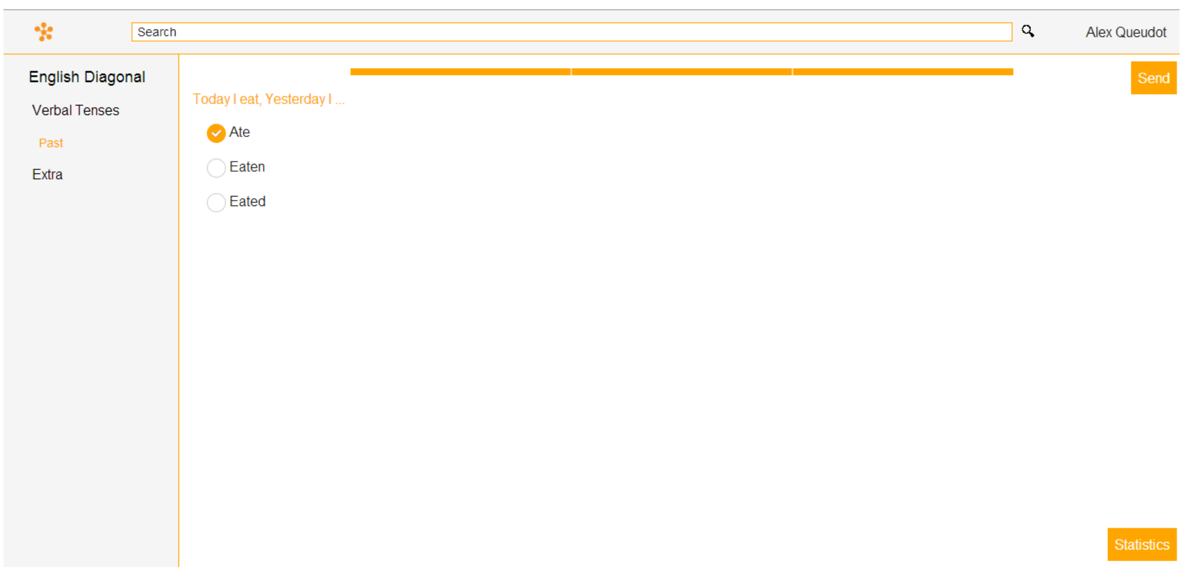


Figure 21 - Question template

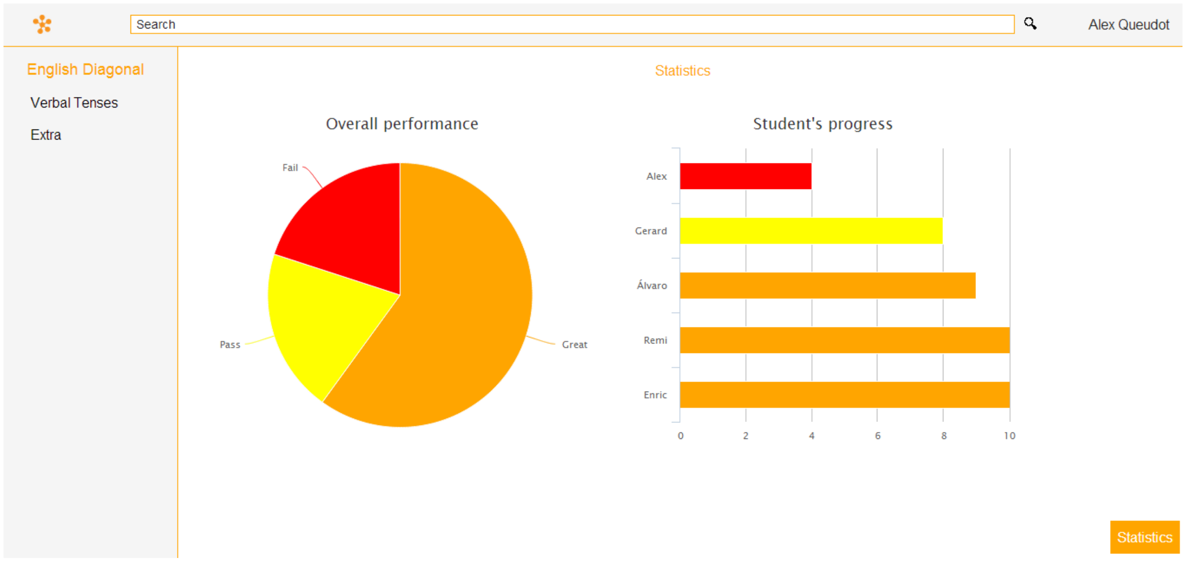


Figure 22 - Statistics

4 Development and Testing

In this chapter there will be a detailed analysis of all the technologies used in the deployment of the platform as well as some of the reasons behind those choices.

The application has been split into server and client side, where both can be run in different machines while exchanging information using REST HTTP calls. Even though that was not necessary, it allowed a bigger insight into how to deploy and use an effective API as well as creating a more dynamic platform with centralized data that can be used and manipulated by different front-end applications.

4.1 Server Side Analysis

The server will be placed in a machine running Linux's Debian 7.0 "Wheezy" distribution. Debian is a widely used Linux environment which a strong community behind and is supported by many servers and applications that run on a UNIX like operative system [55]. An old computer has been put to use for such purpose. It will be located inside a LAN (Local Area Network) and will have access to the internet through a router, which will in turn act as a firewall, forwarding only the http/https requests to the machine's specific port where the server is running. Debian also provides its own firewall, iptables, to prevent an attacker already inside the LAN from reaching other ports without access.

The server itself will use the django REST framework [56] based in python that will act as layer between client applications and the root database where all the content and data will be stored. Django REST is a well-documented, secure and stable framework that will handle the basic tasks of server-client communications. The framework works as an API, meaning that other client applications can be created and access the database as well as the one deployed for this platform. The use of this framework will also allow to experience a python based environment to try and learn a new programming language.

The database will use the MySQL model which will be created and managed through the django framework. MySQL is a stable and widely spread relational database management system that has been studied in the university, so there is previous knowledge about how to structure the architecture and relationships as well as creating and optimizing queries.

In order to make the server available through the internet, the domain "*nouage.org*" has been acquired via "one.com" [57] and redirected to the local IP where the server is running inside a NAT.

a) Deploying Django REST

The Django REST framework contains 4 key files, used to deploy this platform, that shape the core of the back-end application. All those files have been programmed via terminal using the command line "vim" editor.

I) settings.py

This is the main file for configuring the server. It includes the location of html templates, all the installed apps, modules, global variables, database info, etc.

II) models.py

This file defines all the elements that can be added and manipulated inside the database. It is structured in "classes" which is the equivalent as a table from a database point of view. Each class contains a set of parameters, such as the ID as a default primary key, text fields, foreign keys, Booleans, specific functions, etc.

The code below is a snippet of the models file and is used in order to declare a topic for a course in the database.

```
class Tema(models.Model):
        course = models.ForeignKey(Course)
        name = models.CharField(max_length=512)
        description = models.TextField()
        def __unicode__(self):
           return self.name
```

As explained in previous chapters it is linked with a course and it contains the name and a description (the content) as parameters. This class also has a defined function to display the name as the identifier when accessing the admin GUI console.

The models file also allows extending the default User Profile to add more attributes and have a customized control over the users. The property below for example, creates a new

"StudentProfile" for each user added in the platform that is linked to the original User instance from the framework. The StudentProfile can then be modified, which allows to declare and link any information relevant for the platform.

User.profile = property(lambda u: StudentProfile.objects.get_or_create(user=u)[0])

III) views.py

This is the main file for querying the database through the django REST framework. It uses python language instead of SQL for the queries since it actually refers to the models (see above) and then the framework will create the SQL query from that. All the functions declared there will define how users can interact with the elements in the database (which are the models created in the models.py file). Non-admin users do not have access to the database outside the range defined by using those methods.

The code below is a basic sample of a function called to retrieve all the topics inside the database in order to display them in the client application.

```
def gettemas(request):
	tema_key = 'temas'
	course_id = request.GET['id']
	courseInstance = Course.objects.get(id=course_id)
	temas = list(courseInstance.tema_set.all().values('pk', 'name', 'description',
'course'))
	response=HttpResponse(json.dumps({tema_key:temas}),
content_type='application/json; charset=utf8')
	response['Access-Control-Allow-Origin'] = "*"
	return response
```

This method can be called from a URL (more information in the urls.py analysis below) and through the http request, interact with the database following the rules defined in the function.

Django REST also allows for notations on its methods such as "@login_required", allowing the execution of the code only if the user is logged in the system, or "@csrf_exempt", to remove the security "cookie" that django uses in its communication to verify that the user has previously connected with the database.

IV) urls.py

This file can be considered the “DNS” of a django REST API. What it does, basically, is mapping methods defined in the views.py file with external URLs accessible to users. It contains the list of all the possible addresses that clients can use in order to interact with the database.

The line of code used to map the method shown above to a URL can be seen below.

url (r'^api/gettemas/$', views.gettemas, name="Get Temas")

To run the *“gettemas”* function then, a user would have to send an HTTP request to the url specified above. An example for that call would be:

http://www.nouage.org/courses/api/gettemas/?id=1

4.2 Client Side Analysis

The client application is web based and has been developed using the common web languages HTML5, CSS3, JavaScript, AJAX and jQuery. For that purpose, the NetBeans IDE 8.0 has been used as coding environment.

The base html file is the “index.html” which is blank by default. With the user logged in, the web-app automatically calls the server in order to retrieve all the information about the courses and fill the navigation bar. Later on, it will send additional calls to get the content for each item selected by the user. Therefore, the application is entirely dynamic, depending completely on the server’s content, and is being created with the HTTP requests and JavaScript functions as the user navigates through the app.

For each model in the database, a JavaScript object is created, inside an array as long as the retrieved elements from the server, to store the data locally after the first request and display it to the user. Handling objects locally also allows for more specific and optimal requests since all the ids of the elements are being stored, as well as optimizing the internet transmissions. The code below is an example of the simplest element, the course and then the function called to retrieve the course models from the database.

```
function Course(courseid, coursename, description) {
   this.id = courseid;
   this.name = coursename;
   this.description = description;
   this.temas = new Array();
}

function getCourses() {
   $.getJSON(SERVER_IP + SERVER_GET_COURSES, function(data) {
      if (data.courses.length === 0)
         return;
      for (var i in data.courses) {
         COURSES.push(new Course(data.courses[i].pk, data.courses[i].name,
data.courses[i].description));
      }
      setCourses();
   });
}
```

Each element in the left navigation drawer has click listeners attached which will call similar functions to retrieve each element's content from the server and display it on screen.

For the analytics section, the JavaScript library "Highcharts" [58] has been used, under the Creative Commons Attribution-NonCommercial 3.0 License, to simplify the process of displaying the statistics. This library handles all the elements' drawing while allowing for a high degree of customization with very few and simple lines of code.

4.3 Testing

a) Security

As explained above, the server has a public domain name (*nouage.org*) that can be accessed by anyone. This domain is translated into a public IP given by ONO, the ISP (internet service provider) of the line where the server is running and finally redirected to a LAN router. This router acts as a first firewall for all the requests made to nouage.org, forwarding only requests coming from the port 80 (http) and 8000 (the port used by the server). Those requests will be redirected only to the server machine, therefore preventing attackers to access other machines inside the LAN. This configuration can be seen in Figure 23 below.

PORT FORWARDING

External IP Address: 62.57.176.113

Port Forwarding							
Local IP Addr	External		Internal		Protocol	Enabled	Delete
	Start Port	End Port	Start Port	End Port			
192.168.1.5	80	80	8000	8000	Both	☑	☐
192.168.1.5	8000	8000	8000	8000	Both	☑	☐

Figure 23 - Port Forwarding

The Debian server adds another layer of security, the iptables firewall, which is configured to allow only INPUT and OUTPUT packets going through the server and with the destination port 8000. IP forwarding is deactivated so that a user who has accessed the server won't be able to connect with other machines inside the same network.

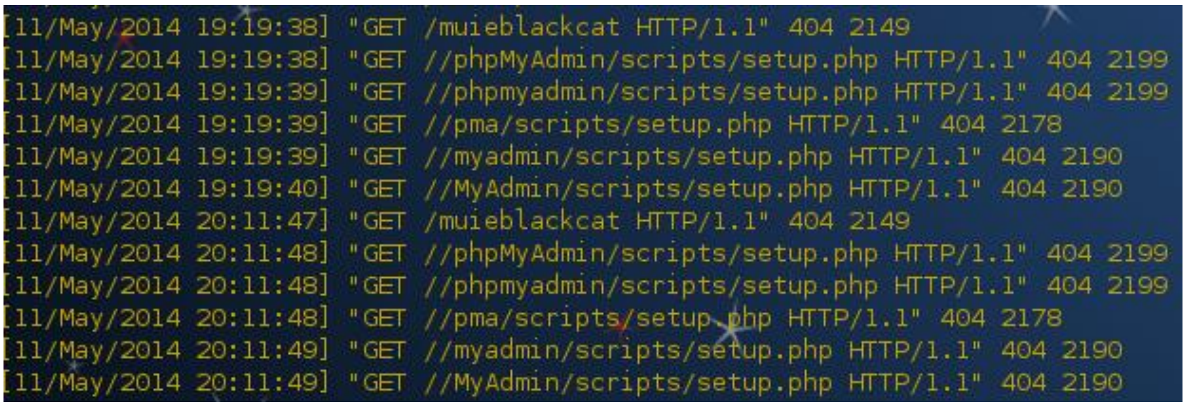
To test its security in a real scenario, the server remained open during a whole week with real-time logs to see all the incoming connections. After a few hours, there were already attempts to access the server. Those can be classified into two categories.

I) Indexing: Harmless requests, mostly done by search engines or domain listing webpages, trying to index the site and get part of its content stored in their databases for public viewing. They can easily be identified because they try to access the "index.html" and/or "robots.txt" file and do not request specific root or admin locations as seen in the log below with Figure 24.

```
/2014 18:31:43] "GET / HTTP/1.0" 404 2110
/2014 18:31:44] "GET /index.php HTTP/1.0" 404 2137
/2014 18:34:37] "GET /robots.txt HTTP/1.1" 404 2144
/2014 18:34:37] "GET / HTTP/1.1" 404 2114
```

Figure 24 - Indexing attempt

II) Hacking attempts: Those are standardized attacks, usually scripts trying to access typical locations such as "~root" or admin folders used by common frameworks. Examples of this can be seen in Figures 25 and 26 below, with all the unauthorized requests returning 404 Access Forbidden to the attacker. In Figure 26 particularly, the attacker found the admin location to configure the database via web GUI and tried a brute force attack to find the login/password combination using the "Fast HTTP Auth Scanner"[1] and then log as root. However, the password is a combination of upper and lower case letters, numbers and special characters, so a brute force attack to find it is not a very good approach. Either way, the attack was unsuccessful,



```
[11/May/2014 19:19:38] "GET /muieblackcat HTTP/1.1" 404 2149
[11/May/2014 19:19:38] "GET //phpMyAdmin/scripts/setup.php HTTP/1.1" 404 2199
[11/May/2014 19:19:39] "GET //phpmyadmin/scripts/setup.php HTTP/1.1" 404 2199
[11/May/2014 19:19:39] "GET //pma/scripts/setup.php HTTP/1.1" 404 2178
[11/May/2014 19:19:39] "GET //myadmin/scripts/setup.php HTTP/1.1" 404 2190
[11/May/2014 19:19:40] "GET //MyAdmin/scripts/setup.php HTTP/1.1" 404 2190
[11/May/2014 20:11:47] "GET /muieblackcat HTTP/1.1" 404 2149
[11/May/2014 20:11:48] "GET //phpMyAdmin/scripts/setup.php HTTP/1.1" 404 2199
[11/May/2014 20:11:48] "GET //phpmyadmin/scripts/setup.php HTTP/1.1" 404 2199
[11/May/2014 20:11:48] "GET //pma/scripts/setup.php HTTP/1.1" 404 2178
[11/May/2014 20:11:49] "GET //myadmin/scripts/setup.php HTTP/1.1" 404 2190
[11/May/2014 20:11:49] "GET //MyAdmin/scripts/setup.php HTTP/1.1" 404 2190
```

Figure 25 - Unauthorized server access attempt

```
[25/Apr/2014 18:07:47] "GET / HTTP/1.1" 404 2113
[25/Apr/2014 18:07:48] "GET http://www.fbi.gov/ HTTP/1.1" 404 1644
[25/Apr/2014 18:07:49] "CONNECT www.fbi.gov:80 HTTP/1.0" 404 1640
[25/Apr/2014 18:07:50] "GET /admin/ HTTP/1.1" 200 1870
[25/Apr/2014 18:07:55] "GET /FastHTTPAuthScanner200test/ HTTP/1.1" 404 2194
[25/Apr/2014 18:07:56] "GET /~root/ HTTP/1.1" 404 2131
```

Figure 26 - Brute force login attempt

b) Usability

Once the platform is deployed and working, it is important to test its components from a user's point of view. This basically involves the client application (considering the server queries are already secure, fast and optimal) and UX. There are some areas that are sometimes not regarded during the programming phase since their functionality is purely esthetical or just to provide a more natural and user-friendly control of the application. This is usually done with user feedback since they provide a completely new and external point of view of someone who is not used to deal with the application.

It is important then to check that all steps are intuitive and easy to guess, or otherwise provide clear and simple instructions as what to do next. Also try that all the buttons or links are responsive and that the user knows what is going on in the app all the time.

[1] http://seclists.org/fulldisclosure/2007/Jul/518

Another important test is to try worse case scenarios, such as very long names for topics, huge amount of content, large questions, uppercase letters, etc.

One change made in this platform after usability testing was to highlight the current item the user selected in the navigation bar so it has a better visual idea of where he is in every moment. Another change considered that was not finally deployed was to display some sort of “waiting bar” to show every time the user queried for new content, so he would knew that the request was sent and would not try to click again or think that the link was broken. However since the server has a very high response time, the delay was not important enough to add an extra element in the screen which may annoy the user due to its needless.

4.4 Deployment and User Feedback

When the platform is ready, it is important to try it with real users who can provide feedback to improve it, as well as test it, in a real life scenario, to demonstrate its strengths and flaws. That will lead to future improvements as well as providing a reality check for the original idea.

For this purpose, I selected a group of five 10 year old students I teach English to, and asked them to use this platform to do their homework instead of handing it to them in paper. The testing was done in two sessions, and was performed by the students at their homes. After each of the two consecutive classes, the homework was uploaded to the server and the students would log in from their computer/iPads and solve the questions. The next class there would be a short period of time dedicated to feedback and the students’ experience on using the platform. The results and conclusions after the few testing iterations were the following:

I) Minor bug fixes: As expected, at the beginning some small errors were found in the application yet quickly solved.

II) Dynamic content: Since the homework was fixed and prepared at the beginning of the course, the platform allowed to change some of those questions after a class to reflex better the needs of the student; diminishing the areas they already excel as well as increasing those they had more trouble with during the class.

III) Analytics: One of the core features of the platform, the statistics system together with all that data collected in a database, allowed for a deeper view of the student's understanding on concrete topics and the appliance of that in next classes.

IV) Feedback: Having users test the application for its purpose, allowed for new ideas and possibilities to implement. More about this will be discussed in the next chapter.

V) Engagement: Doing the homework from their computers was more comfortable to them than having to write the answer in a piece of paper. However this part can be greatly improved and will be explained with further details in the next chapter.

VI) Real-time learning: Instead of answering the questions in the homework sheet and having to wait until the class to know if it was right or wrong, they could see it right away and learn the mistake in order to not commit it again in the next question. It also solves a problem from a learning point of view, where the students are answering something they think it is correct yet it may not be, but their brains retain that information as true until it is proven otherwise.

VII) Optimization: In the next class, instead of giving the same weight to each question for each student, it was easier to focus on the most problematic ones and identify students having troubles with specific topics and help them.

5 Conclusions and Future Work

This has been a great project to make; it has allowed me to learn how technology can be effectively mixed with education, from a psychological point of view as well as a deeper technological approach, more related to my studies. It was very interesting to have a deeper understanding on the current state of ICT appliances in education, both failures and successes, as well as developing a teaching platform from scratch. I had the opportunity to learn all those different and useful programming languages (python, html5, JavaScript, css3 and jQuery) and discovering great frameworks, libraries and resources to use in further projects. It was also great to see how to deploy a server and create a web-based client application pretty much from the ground and join them to work as one while creating a dynamic environment that can be expanded with future features, content and/or external apps. Testing and putting to work a self-made application to solve a real-life problem and have a purpose, was definitely an incredibly enriching and rewarding experience.

Regarding the project itself, it is definitely a useful tool that brings some fresh clean air into the current platforms used in education and that, at least in the environment where it was deployed and tested, provided a new perspective and set of optimizations that before couldn’t be possible to do with so much ease. It improved the way my students learn, as well as allowing me to have more detailed information about their learning progress in order to optimize my classes and adapt them to their needs.

With all that in mind, there is still a huge room for improvements and further work to do. Below are some interesting and key features to add in order to give a personality to the platform and make it more appealing to students:

I) Edit interface: Simple interface for teachers to add its content directly from the client app instead of going through the server.

II) Socialization: A comments section should be added into topics and/or questions, where students can leave their opinion or expose their doubts and create discussions inside the content itself.

III) Analytics: The statistics system can be improved a lot. Teachers should be able to customize what they personally believe they need to know about their course and students and see this information with just a glance at the screen. The data is already there, it just needs the adequate parsing and display.

IV) Gamification: Right now the application is purely simple and functional. A small degree of Gamification should be added to motivate students into solving the exercises and create a healthy competition to promote learning. This could be done by creating a leaderboard of best students, awarding “badges” to great performers and, overall, engage them into the learning process turning it into a game.

V) Machine Learning: That is a really difficult yet incredibly interesting feature to deploy in a learning platform. The typical quiz or single word answering methods are great and easy to implement, in order to quickly see if the student understood the topic, yet it definitely has its downsides. These testing methods completely limits the student’s scope and establishes the idea that for every question there are a set of predefined answers that you have to choose from, which in most cases it does not match reality and it definitely does not empower the student’s creativity. It would be great to create an algorithm that scans through a custom and lengthy answer from a student, identifies patterns and suggests corrections, as well as displaying the ideas the student had right. This might be done through engaging students to correct each other answers under the supervision of the teacher and store all this information in the database during a certain period of time. When enough data and answers about a certain topic are collected, that algorithm would parse it and compare it with the analysis of a new answer. That should lead to an accurate correction as if it was a real teacher, with all the advantages of scalability and personalization.

Bibliograhy

[1] ITU (February 2014). ICT Facts and Figures. Available: http://www.itu.int/en/ITU-D/Statistics/Documents/facts/ICTFactsFigures2014-e.pdf

[2] Oxford University Press (February 2013). Online Resources Centres. Available: http://global.oup.com/uk/orc/learnvle/

[3] UCISA (December 2011). Management and implementations of Virtual Learning Environments. Available: http://www.ucisa.ac.uk/~/media/groups/tlig/vle_surveys/VLEsurvey%20pdf.ashx

[4] Google. Google Classroom: Home page. *2014 (06/03/2014)*. Available: http://www.google.com/edu/classroom/

[5] Google. Official Blog - Previewing a new classroom. (May 2014). Available: http://googleblog.blogspot.com.es/2014/05/previewing-new-classroom.html

[6] Google. Google Apps for Education: Home page. *2014 (06/03/2014).* Available: http://www.google.com/edu/

[7] Ryann K. Ellis. (2009). Guide to Learning Management Systems. Available: http://www.astd.org/~/media/Files/Publications/LMS_fieldguide_20091

[8] TheLMSapp. (January 2014). The Top 5 Most Popular LMS. Available: http://www.thelmsapp.com/the-top-5-most-popular-lms/

[9] Moodle. Moodle: Home page. *2014 (06/03/2014)*. Available: https://moodle.org/

[10] Edmodo. Edmodo: Home page. *2014 (06/03/2014).* Available: https://www.edmodo.com

[11] Oxford Dictionary. MOOC's definition. *2014 (06/03/2014)*. Available: http://www.oxforddictionaries.com/definition/english/MOOC

[12] Coursera. Coursera: Home page. *2014 (06/03/2014).* Available: https://www.coursera.org/

[13] Coursera. About Coursera. *2014 (06/03/2014)*. Available: https://www.coursera.org/about/community

[14] Miquel Oliver, Davinia Hernádez-Leo, Vanesa Daza, Carles Martín and Laia Albó. (January 2014). Panorama actual de los Cursos Masivos Abiertos en Línea en las universidades españolas. Available: http://www.catedratelefonica.upf.edu/wp-content/uploads/2014/02/MOOCs-en-Espa%C3%B1a1.pdf

[15] Miríada X. Nuestra Filosofía. *2014 (06/03/2014)*. Available:
https://www.miriadax.net/nuestra-filosofia
[16] Moodle. Moodle's Statistics. *2014 (06/03/2014)*. Available: https://moodle.org/stats/
[17] Colin Matheson. (January 2012). The joys of ed tech. Available:
http://cytochromec.net/blog/2012/01/why-use-moodle-when-you-can-use-something-shiny/
[18] Quora. Why after all these years is Moodle still so ugly. *2014 (06/03/2014)*. Available:
http://www.quora.com/Moodle/Why-after-all-these-years-is-Moodle-still-so-ugly#
[19] Moodle. Resources for Download. *2014 (06/03/2014)*. Available:
http://download.moodle.org/
[20] Moodle. Plugins list. *2014 (06/03/2014)*. Available:
https://moodle.org/plugins/browse.php?list=category&id=2
[21] The MOOC Guide. CCK08 - The Distributed Course. *2014 (06/03/2014)*. Available:
https://sites.google.com/site/themoocguide/3-cck08---the-distributed-course
[22] The New York Times (November 2012). The Big Three MOOC Providers. Available:
http://www.nytimes.com/2012/11/04/education/edlife/the-big-three-mooc-providers.html
[23] Coursera. About Coursera. *2014 (06/03/2014)*. Available:
https://www.coursera.org/about/
[24] Coursera. Coursera Official Blog. *2014 (06/03/2014)*. Available:
http://blog.coursera.org/post/64905875290/with-13-new-organizations-coursera-reaches-100
[25] edX. About edX. *2014 (06/03/2014)*. Available: https://www.edx.org/about-us
[26] Udacity. About Udacity. *2014 (06/03/2014)*. Available: https://www.udacity.com/us
[27] Meetup. Meetup: Home page. *2014 (06/03/2014)*. Available: http://www.meetup.com/
[28] Google. Google Hangouts: Home page. *2014 (06/03/2014)*. Available:
https://plus.google.com/hangouts
[29] MOOCS. MOOCS: Homepage. *2014 (18/03/2014)*. Available:
http://www.moocs.com/
[30] Scientific American. (March 2013). Massive Open Online Courses, aka MOOCs, Transform Higher Education and Science. Available:
http://www.scientificamerican.com/article/massive-open-online-courses-transform-higher-education-and-science/

[31] New York Times (November 2014). Massive Open Online Courses Are Multiplying at a Rapid Pace. Available: http://www.nytimes.com/2012/11/04/education/edlife/massive-open-online-courses-are-multiplying-at-a-rapid-pace.html?pagewanted=all&_r=0

[32] Standford University - Education's Digital Future. (April 2014). Where has all the media's MOOC coverage gone? Available: http://edf.stanford.edu/readings/2014-where-has-all-medias-mooc-coverage-gone

[33] Dhawal Shah. (December 2013). MOOC's in 2013: Breaking Down the Numbers. Available: https://www.edsurge.com/n/2013-12-22-moocs-in-2013-breaking-down-the-numbers

[34] San José University. San José University: Home page. *2014 (18/03/2014)*. Available: http://www.sjsu.edu/

[35] Udacity. Udacity's Official Blog. *2014 (18/03/2014)*. Available: http://blog.udacity.com/2013/01/sebastian-thrun-udacity-announces-for.html

[36] Coursera. Coursera Signature Track. *2014 (18/03/2014)*. Available: https://www.coursera.org/signature/guidebook

[37] ACE. American Council on Education. *2014 (18/03/2014)*. Available: http://www.acenet.edu/Pages/default.aspx

[38] The EvoLLLution. American Council on Education Looking to Recommend MOOCs for Credit. *2014 (18/03/2014)*. Available: http://www.evolllution.com/friday-links/american-council-on-education-looking-to-recommend-moocs-for-credit/

[39] The Economist. The Economist: Home page. *2014 (18/03/2014)*. Available: http://www.economist.com/

[40] The Economist. (October 2013). Will MOOCs kill university degrees? Available: http://www.economist.com/blogs/economist-explains/2013/10/economist-explains

[41] Khan Academy. Khan Academy: Home page. *2014 (25/03/2014)*. Available: https://www.khanacademy.org/

[42] Khan Academy. About Khan Academy. *2014 (25/03/2014)*. Available: https://www.khanacademy.org/about

[43] New York Times. (January 2014). Salman Khan Turned Family Tutoring into Khan Academy. Available: http://www.nytimes.com/2014/01/28/science/salman-khan-turned-family-tutoring-into-khan-academy.html

[44] Jeff Dunn. (April 2013). How 12 Schools Are Using Khan Academy Right Now. Available: http://www.edudemic.com/how-12-schools-are-using-khan-academy-right-now/

[45] Gates Foundation. Gran - Khan Academy. *2014 (25/03/2014)*. Available: http://www.gatesfoundation.org/How-We-Work/Quick-Links/Grants-Database/Grants/2010/10/OPP1025663

[46] California K-12 Books. What are K-12 Schools? *2014 (25/03/2014)*. Available: http://www.californiak12books.org/what-are-k-12-schools.php

[47] Salman Khan. (March 2011). Let's use video to reinvent education. Available: http://www.ted.com/talks/salman_khan_let_s_use_video_to_reinvent_education

[48] Nicole Baldocchi. (August 2011). School District Expands Khan Academy to All Schools. Available: http://losaltos.patch.com/groups/schools/p/school-district-expands-khan-academy-to-all-schools

[49] The Washington Post. (February 2012). Sal Khan responds to a critic -- and the critic answers back. Available: http://www.washingtonpost.com/blogs/answer-sheet/post/sal-khan-responds-to-critic/2012/07/25/gJQA83rW9W_blog.html

[50] The Washington Post. (July 2012). Khan Academy: The hype and the reality. Available: http://www.washingtonpost.com/blogs/answer-sheet/post/khan-academy-the-hype-and-the-reality/2012/07/22/gJQAuw4J3W_blog.html

[51] Wired. (January 2014). The Learning Revolution: It's Not About Education. Available: http://www.wired.com/2014/01/learning-revolution-education/

[52] CB Insights. (April 2014). Global Ed Tech Financing Hits Record in Q1 2014. Available: http://www.cbinsights.com/blog/ed-tech-venture-capital-record

[53] The Guardian. (January 2014). Are Moocs the best chance we have to satisfy a global thirst for education? Available: http://www.theguardian.com/education/2014/jan/20/moocs-global-thirst-education

[54] Henry C. Lucas. (October 2013). Can the Current Model of Higher Education Survive MOOCs and Online Learning? Available: http://www.educause.edu/ero/article/can-current-model-higher-education-survive-moocs-and-online-learning

[55] Debian. Debian: Homepage. *2014 (12/04/2014)*. Available: https://www.debian.org/

[56] Django REST Framework. Django REST Framework: Homepage. *2014 (25/03/2014)*. Available: http://www.django-rest-framework.org/

[57] One.com. One.com: Homepage. *2014 (25/03/2014)*. Available:
http://www.one.com/en/

[58] HighCharts. HighCharts: Homepage. *2014 (25/03/2014)*. Available:
http://www.highcharts.com/