

Bachelor's Thesis

Bachelor's degree in Industrial Technology Engineering

Development of a computer-based tool to manage via web the teaching assignment

REPORT

Author:
Director:
Call:

Maria de Luna Gascón
Manuel Moreno Eguílaz
January 2020



Escola Tècnica Superior
d'Enginyeria Industrial de Barcelona



Summary

Sometimes, the way a thing has always been done is not the best one, or the most efficient. For instance, in the Department of Electronic Engineering at ETSEIB, the staff has always managed the teaching assignment for the next year using a Microsoft Excel spreadsheet. Each professor has to fill this file and send it to the person in charge of this task. Then, each time this person receives it, he has to send it to the next professor in a predefined list and control that every professor is filling it correctly.

This project has the purpose of developing a computer-based tool to make this task easier and faster. The solution has to be able to upload and download files with the teaching information, allow the teaching staff to choose their classes taking into account established criteria and send automatic emails when needed.

With this goal, a web application has been designed, developed and tested. For the visual part, HTML5, some Javascript commands, CSS and a very popular framework called Bootstrap, that helps with the task of creating the interface, have been used. Moreover, the server and the functional part have been implemented in PHP language and SQL in order to manage the corresponding database.

CONTENTS

1. GLOSSARY	9
2. PREFACE	11
2.1. Background	11
2.2. Previous requirements	12
3. INTRODUCTION	13
3.1. Project objectives	13
3.2. Project scope.....	13
4. PROPOSED SOLUTION	14
4.1. Alternatives.....	14
4.2. The solution.....	14
4.3. Specifications	16
5. DEVELOPMENT	18
5.1. Tools.....	18
5.2. Database	20
5.3. Use cases.....	23
5.3.1. Log in	24
5.3.2. Create user	25
5.3.3. Edit user.....	26
5.3.4. Delete user	26
5.3.5. Consult user list	27
5.3.6. Create subject.....	27
5.3.7. Edit subject	28
5.3.8. Delete subject	28
5.3.9. Consult subject list	28
5.3.10. Send automatic email	29
5.3.11. Change the administrator password	29
5.3.12. Choose subjects	30
5.3.13. Validate subject choice	30
5.3.14. Consult personal data	31
5.3.15. Consult professor distribution	31
5.3.16. Download Excel	32
5.3.17. Sign out.....	32
5.4. Operation diagram	33

6. VALIDATION AND FUNCTIONALITY TEST	34
7. BUDGET	38
7.1. Human resources	38
7.2. Hardware	40
7.3. Software.....	40
7.4. Indirect costs.....	41
7.5. Total	42
8. ENVIRONMENTAL AND SOCIAL IMPACT	43
9. FUTURE IMPROVEMENTS	44
CONCLUSIONS	45
ACKNOWLEDGMENTS	46
BIBLIOGRAPHY	47
Bibliographic references.....	47
Complementary bibliography	48

FIGURE INDEX

Figure 1: Example of Excel File. Source: Department of Electronic Engineering at ETSEIB-UPC.	11
Figure 2: HTML, CSS and Javascript icons. Source: Shutterstock.....	15
Figure 3: Database named as tfg. Source: phpMyAdmin.	20
Figure 4: Table of Professors. Source: phpMyAdmin.	20
Figure 5: Table of Assignatures. Source: phpMyAdmin.....	21
Figure 6: Table of Globals. Source: phpMyAdmin.	22
Figure 7: Administrator options.	23
Figure 8: Login page.	23
Figure 9: Use cases.	24
Figure 10: Administrator Homepage.	25
Figure 11: Create teacher individually.....	25
Figure 12: Confirmation window when erasing users.	26
Figure 13: User list.	27
Figure 14: Subject list.....	29
Figure 15: Automatic mail generated.	29
Figure 16: Change administrator password page.	30
Figure 17: Page for choosing subjects.....	30
Figure 18: personal data page.	31
Figure 19: Professor distribution page.	32
Figure 20: Page with the button for downloading the Excel file.	32
Figure 21: Operation diagram of the developed web tool.	33

Figure 22: Survey questions. Source: Google Forms.	35
Figure 23: Survey results. Source: Google Forms.	36
Figure 24: Dedicated hours table.	39
Figure 25: Project members salary table.	40
Figure 26: Amortization cost table.....	40
Figure 27: Software cost table.	41
Figure 28: Electrical energy cost table	42
Figure 29: Total cost table.	42

Note: when there is not a specified source, it means it is own source.

1. Glossary

- CSS: Cascading Style Sheets. It is a language used in addition to HTML for defining styles such as colors, fonts, layouts, etc.
- CSV: Comma-Separated Values file. Each line of the file is a data record. Each record consists of one or more fields, separated by commas.
- ECTS Credits: European Credit Transfer and Accumulation System. Each ECTS credit is equal to 10 hours of teaching in the Bachelor's degree and 9 hours during the Master's degree.
- HTML: HyperText Markup Language. Used for creating and organizing the web content so it can be displayed by a browser.
- PADs: in Catalan *Punts d'Activitat Docent*. Each ECTS credit equals to 3 PADs.
- PHP: recursive acronym of Hypertext Preprocessor. Programming language.
- SMTP: Simple Mail Transfer Protocol.
- SQL: Structured Query Language. Standard language for storing, manipulating and retrieving data in databases.

2. Preface

2.1. Background

Currently, the management of the teaching assignment in the Department of Electronic Engineering at ETSEIB is done via Microsoft Excel. There is an Excel file with a table showing the information of subjects, groups per subject, ECTS credits, professors and other related data. Each professor has to choose which subjects wants to teach the next academic year, write it down and send it to the person in charge of this task (Figure 1 below is an example of this Excel file). Also, the professors are required to follow a specific order when choosing based on the seniority in the school, the more years they have been teaching the more priority they have (they will be able to choose firstly).

Therefore, the person in charge has to receive the Excel file from every professor, then each time he receives it he has to send it to the next professor in the order list and be aware that everyone fills it correctly; moreover; the professors cannot see before they receive the file the current situation, which groups are available. This method is not very efficient and can lead to some confusion or mistakes.

Gran Master (Laboratori i Altres)	Electrónica GRAU	Micros	Energías Renovables	Proyecto II	Amplificio Electrónica	Sistemas d'Instrum. Electrónica	Disseny sobre Silici	Microcomputadors	Sistemes Electr. Digitals	Sistemes d'Electr. de Potència	Vehicles híbrids	Energia Solar	RENE Project	Grups Prof.	PdDs Prof.	Professor
Balado	4,00					0,40								4,40	19,800	Balado
Carrasco								2,00						2,00	9,000	Carrasco
Bordonau										2,00			1,00	3,00	18,000	Bordonau
Rodriguez				1,00	4,00									5,00	28,800	Rodriguez
Manich	1,00				2,00	1,40								4,40	20,700	Manich
Moreno									2,00		1,0			3,00	12,000	Moreno
Busquets	4,00													4,00	18,000	Busquets
Vicenç Parici	0,00				5,00									5,00	24,750	Vicenç Parici
JM Torrents														0,000		JM Torrents
A. Gómez					2,00									2,00	9,900	A. Gómez
F. Guinjoan												1,00		1,00	9,000	F. Guinjoan
J.A. Santos														0,000		J.A. Santos
Daniel F.	2,00													2,00	9,000	Daniel F.
Roger Català	4,00													4,00	18,000	Roger Català
G. Masmitjà					4,00									4,00	19,800	G. Masmitjà
Paul Soto	1,00													1,00	4,500	Paul Soto
Grups	16,00	0,00	0,00	1,00	17,00	1,80	0,00	2,00	2,00	2,00	1,00	1,00	1,00	44,80		
Grups enc.	16,0	0,0	1,0	1,0	17,0	1,0	0,0	2,0	2,0	2,0	1,0	1,0	1,0	46,0		
PdDs	72,000	0,000	0,000	9,000	84,150	8,100	0,000	9,000	9,000	9,000	3,000	9,000	4,500		144,750	
PdDs enc.	72,000	0,000	3,375	9,000	84,150	4,500	0,000	9,000	9,000	9,000	3,000	9,000	4,500		144,625	
Tot PdDs	144,000	0,000	0,000	18,000	126,900	17,100	0,000	18,000	18,000	18,000	12,000	9,000	4,500		385,50	
Tot PdDs enc.	144,000	0,000	3,375	18,000	126,900	13,500	0,000	18,000	18,000	18,000	12,000	9,000	4,500		385,28	

Figure 1: Example of Excel File. Source: Department of Electronic Engineering at ETSEIB-UPC.

2.2. Previous requirements

On one hand, before starting working in the solution itself, it is very important to know and understand how the teaching assignment works and how it has been done until now. This information will help in the process and to think about the best way to improve it.

After this preliminary research, some useful and very relevant information was found. Each professor has a number of what it is known as PADs, in Catalan *Punts d'Activitat Docent*, hired. This number is directly related to the hours of classes the professor is going to impart. At the same time, each subject has a number of assigned ECTS credits.

The ECTS credits system (European Credit Transfer and Accumulation System) is the mechanism provided in the European Higher Education Area to measure the work carried out by university students by acquiring the knowledge and skills necessary to pass the subjects of their curriculum [1]. The study activity (between 25 and 30 hours per credit), includes the time spent in the classes, study hours, tutorials, seminars, work, practices or projects, as well as those required for the preparation and completion of exams and assessments.

For professors, each ECTS credit is equal to 10 hours of teaching in the Bachelor's degree and 9 hours during the Master's degree. Then, the number of PADs results from multiplying by 3 the number of credits ECTS.

On the other hand, there is some knowledge that is useful to have beforehand in order to develop the project. Since it is a computer science oriented solution, knowing some programming language will be very helpful even though is not the same that will be used in the whole project.

3. Introduction

3.1. Project objectives

The main objective of this project is to satisfy a need in the Department of Electronic Engineering at ETSEIB-UPC. As explained previously in the preface, until now to manage the teaching assignment in this department they have used an Excel file that every professor has to fill in with the preferences and it is not an efficient and fast way to do it.

The proposed solution in this project in order to solve this issue and satisfy this need is to create a computer-based tool to manage via web the teaching assignment, in other words, to develop a web application.

The ideal result of this project is to have a functional, intuitive and working web application that really helps the Department of Electronic Engineering at ETSEIB-UPC and makes the process of academic management easier and faster than before.

3.2. Project scope

The idea of the project is to upload the designed tool in one UPC server, so the professors can access it easily like a normal website. The users will only be able to access to the platform using their credentials that will have been sent by the administrator previously.

The web application has been developed for a computer rather than for a smartphone, although it can be accessed through any mobile device, the interface and the whole user experience may not be as intuitive and efficient. In addition, the whole interface will be written in Catalan.

4. Proposed solution

4.1. Alternatives

The first alternative is to keep doing it with the Microsoft Excel file. As it has been explained previously, it is not an efficient way to do it and for the person in charge of the task it is not practical at all.

Another possible solution in order to improve this way of managing the teaching assignment could be using a shared Excel, like Google Sheets, where the owner of the file can decide which people can see and modify it. The main problem with this method is that everyone can modify it completely, not only specific cells (ideally, each person should be able to only modify the cells about their subjects) and also, it would be more difficult to follow the specified order of choosing and the person in charge would have to be controlling very often that people were doing it in the right way.

Then, there is the option that has originated this project, to develop a web application in order to automatize the process and make it faster and easier.

4.2. The solution

As previously explained, the solution is a web tool, specifically a web application, in which every professor will be able to log in with his corresponding credentials.

Although a web application and a website are very similar and sometimes both terms are used indistinctly, they are not exactly the same. The difference between them is based on the content and functionalities. A website is usually static and its purpose is to provide information, while a web application is more dynamic and interactive, it allows the users to perform some actions, it is a website that the user can control [2]. For this reason, a web application can also be referred as a website, but it would not be correct to do it the other way around.

A web application has two parts: the front end and the back end [3], while a website only has the first one. The web functionality relies on each part communicating effectively with the other as a single unit. It is important to define them in order to understand better what web development is and how this project has been done.

The front end is the visible part of the web, also known as “client-side”. It includes the esthetics of the pages including the composition of the buttons, navigation menus, colors, text, labels, etc. The most common front end languages are:

- HTML language, HyperText Markup Language, for the web structure, it is used for creating and organizing the web content so it can be displayed by a browser.
- CSS (Cascading Style Sheets) is a language used in addition to HTML for defining styles such as colors, fonts, layouts, etc.
- Javascript is a programming language for adding some interactive features such as drop down menus, modal windows, and contact forms.



Figure 2: HTML, CSS and Javascript icons. Source: Shutterstock.

These are the basic front end languages, but they can be accompanied by frameworks like Bootstrap and Angular [4], as well as JavaScript libraries like jQuery, and CSS extensions like Sass and LESS. Their purpose is simply to make code easier and more organized by providing several tools and templates compatible with common coding languages.

Then, there is the back end, the invisible yet a primordial part of a web application. It is responsible for storing and organizing data (it manages for example databases) and it also ensures that everything on the client-side actually works.

A database stores website content in a structure that makes it easy to retrieve, organize, edit, and save data. It runs on a remote computer called a server. There are many different databases that are widely used, such as MySQL (now MariaDB), SQL Server, PostgreSQL, and Oracle [5]. They all work with SQL (Structured Query Language), that is, a standard language for storing, manipulating and retrieving data in databases.

The back end communicates with the front end, sending and receiving information to be displayed as a web page. Whenever there is any user interaction on the client-side such as filling out a contact form, logging in or making a purchase, the browser sends a request to the server-side, which returns information in the form of front end code that the browser can interpret and display.

Some common back end languages are Ruby, PHP, Java, .Net, and Python [6]. These programming languages often run on frameworks that simplify the web development process.

The web application referred to in this document has been developed in PHP (recursive acronym of Hypertext Preprocessor) [7] and SQL languages for the back end. Then, the front end has been done with HTML, CSS and Javascript. In addition to these languages, some tools and frameworks have been used. It will be explained later in section “4.2 Tools”.

4.3. Specifications

Concretely, the web application has to be able to perform the following tasks:

- Allow the professors to log in with their corresponding credentials. The idea is that all the professors of the department can log in with their email and DNI number, which will be their password.
- Have an administrator user that can manage the web tool. This person has to be able to create, edit and delete users and subjects.
- Upload the files with the needed information of the professors and subjects in order to create a database. The file has to be in CSV format (comma-separated values file, each line of the file is a data record. Each record consists of one or more fields,

separated by commas) with a determined number of columns specified in the web page.

- Allow the teaching staff to choose their teaching groups for the next academic year.
- Show the current situation of group occupation. The users have to be able to check if there are available groups for all the subjects in every moment and which groups they have selected.
- Show the group distribution. The professors have to be able to check what the other professors have chosen.
- Take into account the established criteria for choosing according to the seniority in the school. The administrator has to be able to change the order if needed.
- Send automatic emails to let the professors know when they can start choosing their academic preferences.
- Download the information in Microsoft Excel format in order to see more clearly the information and distribute it if needed.

5. Development

5.1. Tools

Following a detailed explanation of the main tools used to create the website developed in this project. All the used tools are free and open-source software.

- **Netbeans [8]** is a development environment for creating applications. It recognizes a lot of different programming languages and warns the user when there is a mistake, makes suggestions when coding and allows browsing through the different files and folders of the project. It is generally for Java applications but it supports other programming languages like PHP, C or HTML.
- **XAMPP [9]** is a web server solution developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database (former MySQL), and interpreters for scripts written in the PHP and Perl programming languages. Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local testing server to a live server possible. Therefore, it is a multidisciplinary tool, used mainly for web designers and programmers for testing their projects in their computers before uploading them in a server and without needing access to the internet. .It also provides access to phpMyAdmin, a very useful tool for managing the MariaDB database. Once the tool is installed in the computer, the project folder has to be located inside a specific folder, called htdocs, of the package.
- **Bootstrap [10]** is an open source framework (one of the most popular front-end frameworks) that provides basic templates for the HTML files of a web app and various design tools, such as forms, buttons or drop-down menus, based on HTML and CSS, as well as additional JavaScript extensions. Bootstrap allows you to create an responsive web design, that is, it automatically adjusts to the characteristics of the used device, be it smartphones, computers or tablets, and can be adapted to its size and resolution. It also gives rise to simple and intuitive layouts and is compatible with all modern browsers (Google Chrome, Mozilla Firefox, Internet Explorer, Safari and Opera). All this, added to its extensive documentation [4], makes it the ideal

option for developing the graphic part of the web app.

- **Java SDK [11]** is a software development kit, which provides programming tools. It is required for running Netbeans. For downloading it, a registration in Oracle (the owner of it) is needed but it is free.
- **PHPMailer [12]** is the most used framework for managing the automatic sending of emails (Simple Mail Transfer Protocol, SMTP) from inside a website programmed with PHP language. It can be downloaded in Github.
- **PHPSpreadsheets [13]** is a library written in pure PHP and providing a set of classes that allow to read from and to write to different spreadsheet file formats, like Microsoft Excel and LibreOffice Calc. In this project, it has been used to write in an existing Excel file, which works as a template, the data of groups chosen by every professor. Once the data is imported in the file, some functions calculate the number of PADs and groups covered for every professor and in total. This information can be checked either in general (complete academic year) or for each semester separately. The template used and an example of the filled file can be found in the annex.
- **Github Desktop [14]** is the computer program of Github.com. First of all, Git is an open-source version control system. When developers create a website for instance, they make constant changes to the code, releasing new versions up to and after the first official (non-beta) release. Version control systems keep these revisions straight, storing the modifications in a central repository. This allows developers to easily collaborate, as they can download a new version of the software, make changes, and upload the newest revision. Then, Github.com is where developers store their projects. A repository is a location where all the files for a particular project are stored and can be accessed with a unique URL.

All the files and code of this project are public and can be found in the following link:

<https://github.com/mdlunag/tfg>.

5.2. Database

A database called *tfg* has been created in order to store and display all the needed information in the web application. The XAMPP environment includes a database manager called phpMyAdmin. There, you can manage all the data used in the web application, create new tables, check the information that is being uploaded and modify it.

Three tables (*Professors*, *Assignatures* and *Globals*) have been created inside the database. The phpMyAdmin interface with the database can be seen in Fig. 3 .

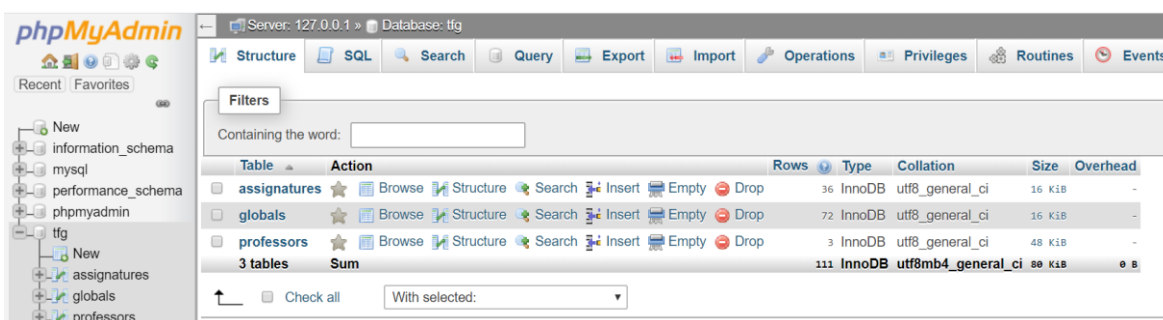


Figure 3: Database named as *tfg*. Source: *phpMyAdmin*.

Next, a detailed explanation of the tables and its fields can be found.

Professors

Each entry of this table gathers the necessary information of the professors except the first entry with *id* = 0, that is the administrator user.

id	nom	cognoms	email	punts	cobert_Q1	cobert_Q2	contrasenya	data_creacio	estat	admin
0	Administrador		admin@admin.com	0	0	0	1234	2019-11-24 00:00:00	0	1
1	Luz	Balado	maria.delunag@gmail.com	123	0	0	0000000x	2020-01-02 12:31:49	1	0
2	Juan Manuel	Moreno Eguilaz	lunapolo2222@gmail.com	321	0	0	0000030x	2020-01-02 12:31:49	0	0

Figure 4: Table of Professors. Source: *phpMyAdmin*.

Its fields are the following:

- *Id*: represents the priority of the professor. The bigger the *id*, the less priority the professor has.

- Nom: the first name of the professor.
- Cognoms: last name of the professor, there can be more than one last name.
- Punts: number of hired PADs of the professor.
- Cobert_Q1: number of PADs that the professor has already covered for the first semester.
- Cobert_Q2: number of PADs that the professor has already covered for the second semester.
- Contrasenya: DNI number of the professor and password for accessing the tool.
- Data_creacio: date when the user was created.
- Estat: state of the user. This field can have 3 different values: 0, 1 or 2.
 - o State 0: the professor does not have permission to choose subjects.
 - o State 1: the professor can choose subjects but has not validated the subject choice.
 - o State 2: the professor has already validated the data but can keep choosing and changing the election made.
- Admin: used to identify the administrator in order to adapt the content of the web application. All the entries have a 0 in this field except the administrator entry, which has value 1.

Assignatures

This table has the information related to the subjects.

id	nom	tipus	credits	grups	quadri
1	electrònica	pràctica	3.0	1	Q1
2	taller elèctric	teoria	4.0	2	Q2
3	tfg	tfg	6.0	1	tots

Figure 5: Table of Assignatures. Source: phpMyAdmin.

Its fields are explained below:

- Id: value used internally for identifying each entry in order to modify or erase it.
- Nom: name of the subject.
- Tipus: type of the subject such as *pràctica* or *teoria*.
- Credits: number of ECTS credits of the subject.
- Grups: number of groups of the subject.
- Quadri: semester in which the subject is going to be imparted.

Globals

This table contains the relation between subjects and professors. Each professor has as many entries as subjects in the table called Assignatures.

id	professor	assignatura	tipus	quadri	grups
1	Luz Balado	electrònica	pràctica	Q1	0.0
2	Luz Balado	taller elèctric	teoria	Q2	0.0
3	Luz Balado	tfg	tfg	tots	0.0
4	Juan Manuel Moreno Eguilaz	electrònica	pràctica	Q1	0.0
5	Juan Manuel Moreno Eguilaz	taller elèctric	teoria	Q2	0.0
6	Juan Manuel Moreno Eguilaz	tfg	tfg	tots	0.0

Figure 6: Table of Globals. Source: phpMyAdmin.

Its fields are:

- Id: value used internally for identifying each entry in order to modify or erase it.
- Professor: full name of the professor.
- Assignatura: name of the subject.
- Tipus: type of the subject.
- Quadri: semester.
- Grups: number of groups of the subject that the professor has chosen to impart.

5.3. Use cases

There are 2 defined roles that can access the web application, the web application will show different content depending on which one has accessed it:

- Administrator: logs in the platform and can perform various administration functions such as create or eliminate users (professors). The administrator can access the administrator options page below (Fig. 7) that is hidden for the other roles.

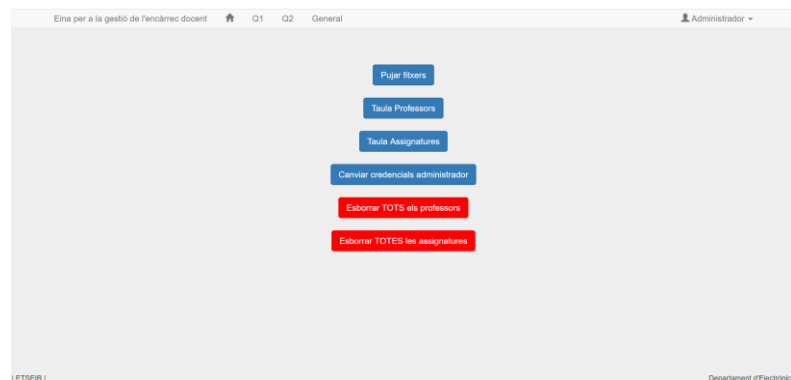


Figure 7: Administrator options.

- User: logs in the platform as a professor in the account that has been previously created by the administrator.

In Fig. 7, there are all the actions that both roles can perform once they access the link of the web application. The first displayed page is the login web page shown in Fig. 8:



Figure 8: Login page.

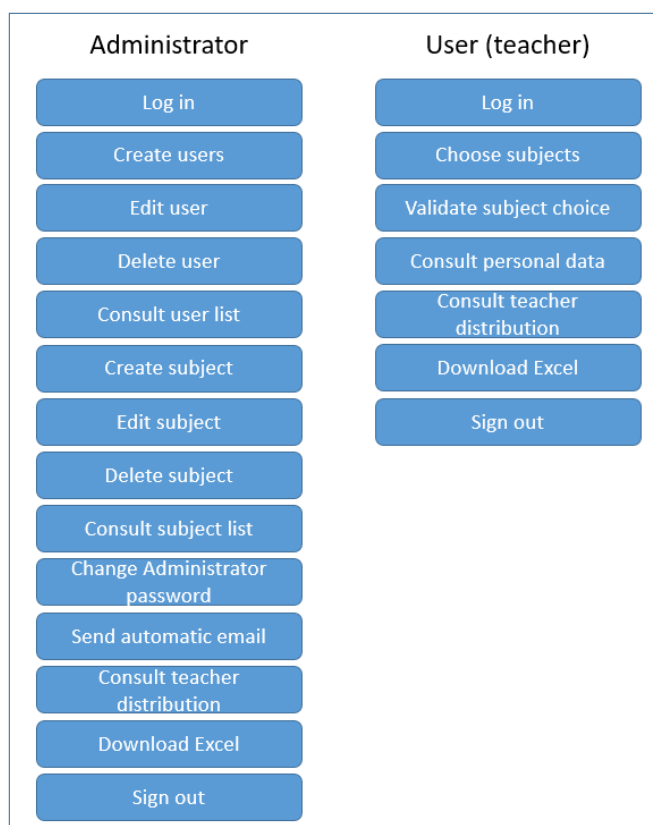


Figure 9: Use cases.

In the following sections, there can be found some screenshots of the interface. In the annex provided there are screenshots of all the web pages.

5.3.1. Log in

Description: authentication in the web in order to access the administrator or user functionalities.

Subjects: administrator and users.

Entry conditions: none.

It checks that the user is registered in the database with administrator or user role and that the entered password is correct. Once the administrator or user logs in, the Homepage is displayed.

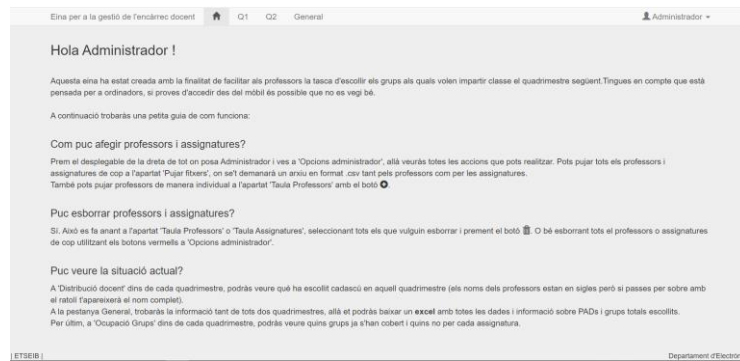


Figure 10: Administrator Homepage.

5.3.2. Create user

Description: registration of users (professors) with all their information such as name, last name, email and DNI (their password) so they can log in to the platform.

Subjects: administrator.

Entry Conditions: to be logged in as an administrator.

There are two ways of creating users inside the web application. The first way is uploading all of them at once through a CSV file and the other one is individually in the web page showing the list of professors (see Fig. 8).

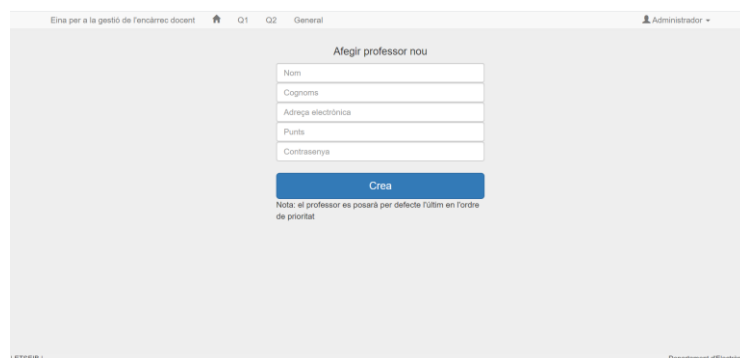


Figure 11: Create teacher individually.

5.3.3. Edit user

Description: change a user information, priority order or state.

Subjects: administrator.

Entry Conditions: to be logged in as an administrator.

Multiple users can be modified at the same time. The administrator has to select the ones desired from the professor list.

The web page shown is the same as the one used for creating a user but with the fields filled with the data of the user that is going to be modified.

5.3.4. Delete user

Description: Remove a user from the database.

Subjects: administrator.

Entry Conditions: to be logged in as an administrator.

When a user is deleted, his related entries in the tables *Professors* and *Globals* are deleted. It can be done globally (erase at once all the users) or individually. Either way, a confirmation window appears before erasing them.

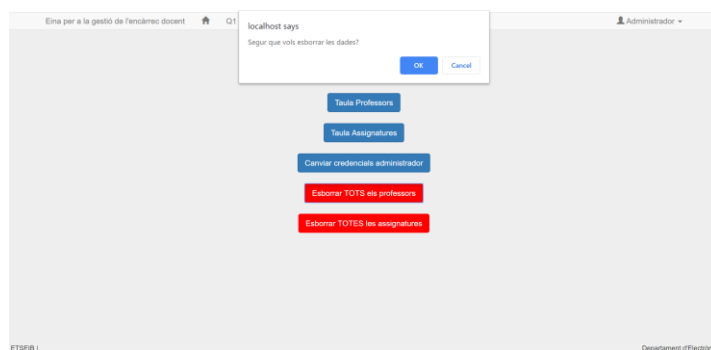


Figure 12: Confirmation window when erasing users.

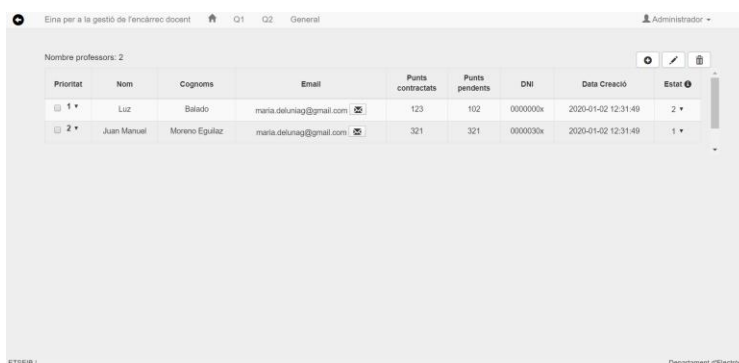
5.3.5. Consult user list

Description: view table with the information of all registered users, their order of priority and state.

Subjects: administrator.

Entry Conditions: to be logged in as an administrator.

The page that shows the user list also allows the administrator to create, edit or delete users.



Nombre professors: 2

Prioritat	Nom	Cognoms	Email	Punts contractats	Punts pendents	DNI	Data Creació	Estat
1	Luz	Balado	maria.delunag@gmail.com	123	102	0000000x	2020-01-02 12:31:49	2
2	Juan Manuel	Moreno Egulaz	maria.delunag@gmail.com	321	321	0000030x	2020-01-02 12:31:49	1

ETSEIB | Departament d'Electrònica

Figure 13: User list.

5.3.6. Create subject

Description: creation of the subjects of the following year providing all the necessary information.

Subjects: administrator.

Entry Conditions: to be logged in as an administrator.

There is only one way of creating subjects inside the web application. It is done by uploading all of them at once through a CSV file.

5.3.7. Edit subject

Description: change a subject information such as the name or type.

Subjects: administrator.

Entry Conditions: to be logged in as an administrator.

5.3.8. Delete subject

Description: Remove a subject from the database.

Subjects: administrator.

Entry Conditions: to be logged in as an administrator.

When a subject is deleted, its related entries in the tables *Assignatures* and *Globals* are deleted. It can be done globally (erase all the subjects at once) or individually. When erasing subjects the same confirmation window as when users are deleted appears.

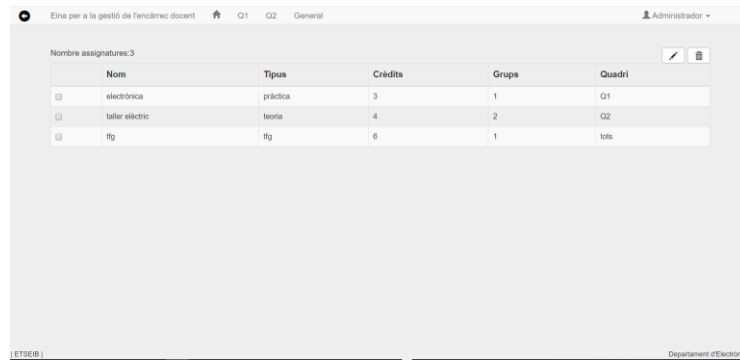
5.3.9. Consult subject list

Description: view table with the information of the subjects.

Subjects: administrator.

Entry Conditions: to be logged in as an administrator.

The page that shows the subject list also allows the administrator to edit or delete subjects.



Elina per a la gestió de l'encàrrec docent

Q1 Q2 General

Administrador

Nombre assignatures: 3

	Nom	Tipus	Crèdits	Grups	Quadri
<input type="checkbox"/>	electrónica	práctica	3	1	Q1
<input type="checkbox"/>	taller eléctrico	teoría	4	2	Q2
<input type="checkbox"/>	itg	itg	6	1	tots

ETSEIB

Departament d'Electrònica

Figure 14: Subject list.

5.3.10. Send automatic email

Description: send an email to let a teacher know that he can already choose subjects.

Subjects: administrator.

Entry Conditions: to be logged in as an administrator.

The administrator can send automatic emails at any time notifying the professor that he can already choose subjects. This action is carried out by clicking the button with an envelope on the web page that shows the professor list.



Figure 15: Automatic mail generated.

5.3.11. Change the administrator password

Description: to change the password.

Subjects: administrator.

Entry Conditions: to be logged in as an administrator.

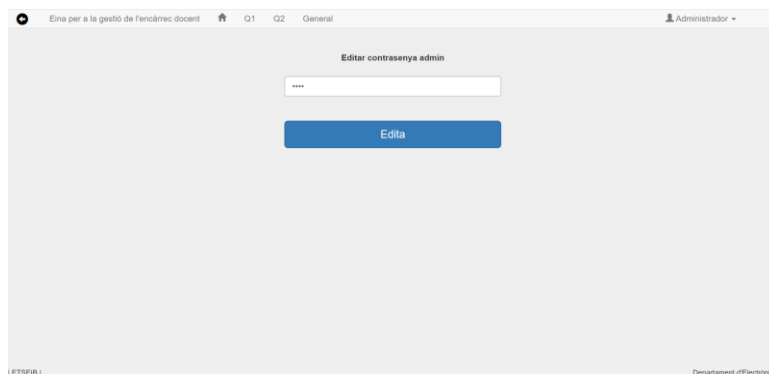


Figure 16: Change administrator password page.

5.3.12. Choose subjects

Description: The users (teachers) have to choose the subjects they want to impart classes to and how many groups they want to teach. There is a different web page for each semester in order to avoid confusions.

Subjects: user.

Entry Conditions: to be logged in as a user with state 1 or 2.



Figure 17: Page for choosing subjects.

5.3.13. Validate subject choice

Description: Once the professors have chosen the subjects and the corresponding number

of groups of each one, they can validate the choice by clicking the *Validar* button.

Subjects: user.

Entry Conditions: to be logged in as a user with state 1.

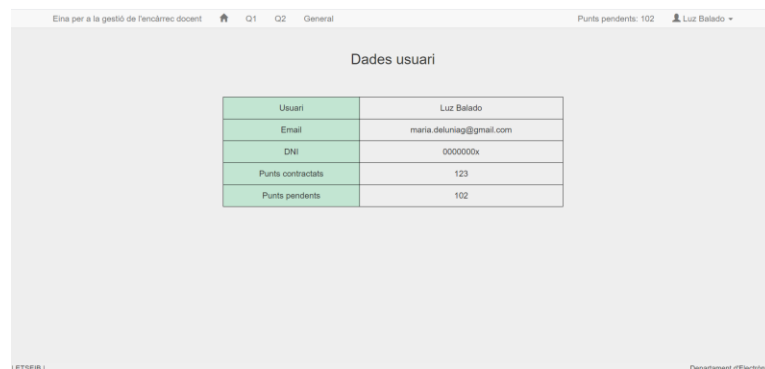
When a professor clicks, the *Validar* button that will appear in the navigation bar once he is in state 1, his state will change from 1 to 2. Then, the next professor in the list will receive an email notifying him that he has permission to choose and his state will change from 0 to 1.

5.3.14. Consult personal data

Description: Users can check their personal data such as email, DNI number or hired points.

Subjects: user.

Entry Conditions: to be logged in as a user.



Dades usuari	
Usuari	Luz Balado
Email	maria.delunag@gmail.com
DNI	0000000x
Puntis contractats	123
Puntis pendents	102

Figure 18: personal data page.

5.3.15. Consult professor distribution

Description: See what all the professors have chosen.

Subjects: Administrator and users.

Entry Conditions: to be logged in as an administrator or user.

	LB	JMME
electrónica práctica	1	0
tfp tfg	1	0

Figure 19: Professor distribution page.

5.3.16. Download Excel

Description: Download Microsoft Excel file with all the teaching assignment information.

Subjects: Administrator and users.

Entry Conditions: to be logged in as an administrator or user.

The Export to Excel button can be found in the page that shows the professor distribution for all semesters.

	LB	JMME
electrónica práctica Q1	1	0
taller eléctrico teoría Q2	0	0
tfp tfg tots	1	0

Figure 20: Page with the button for downloading the Excel file.

5.3.17. Sign out

Description: Leave the user login void.

Subjects: Administrator and users.

Entry Conditions: to be logged in as an administrator or user.

When the administrator or users sing out they are redirected to the login page.

5.4. Operation diagram

The diagram displayed in Fig. 21 shows the operation diagram of the web tool. The purpose of this diagram is to clarify the actions that each person that access the web application can take depending on the role. The first distinction is whether the person is the administrator or not. If not, it is important to determine the user state and adapt the permissions depending on it.

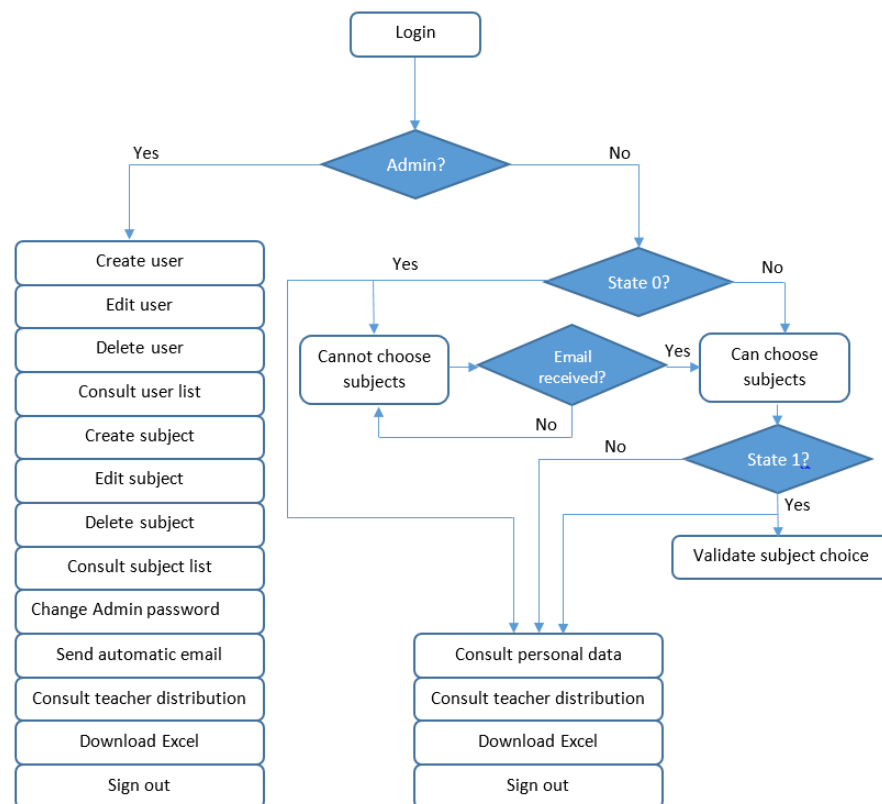


Figure 21: Operation diagram of the developed web tool.

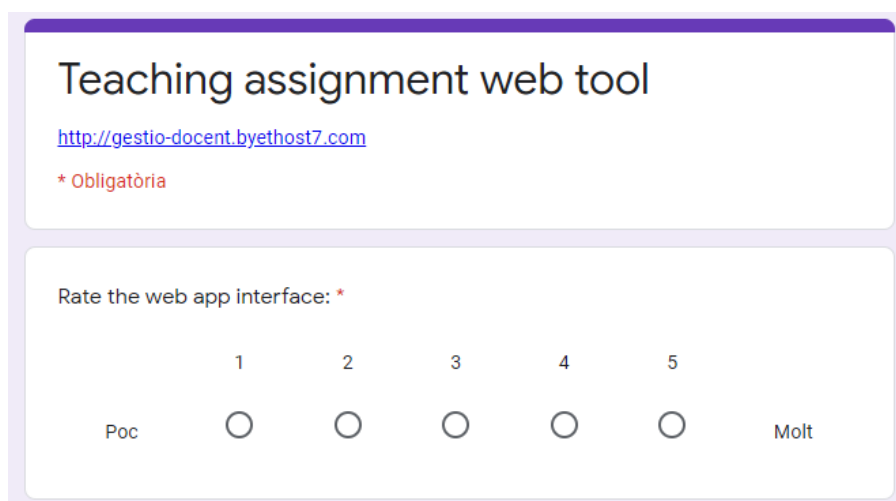
6. Validation and functionality test

A preliminary validation process has been carried out to evaluate its usability and acceptance. The term preliminary validation is used because it is a small recruitment sample (of 10 people) and the recruitment process has been carried out through close people. A larger, more random sample would be needed in order to consider (completely) significant results. The results are therefore indicative and aimed at finding out the reaction and evaluation of the application, as well as identifying possible errors and improvements.

In order to test the web application it has been uploaded in a free hosting server called byethost7. The link of the website is the following: <http://gestio-docent.byethost7.com>.

Then, the link has been sent to ten different people with some instructions (like username, and administrator password, purpose of the web application, etc) so they could test it, find bugs and make suggestions.

In addition, a very short survey has been sent to these people. The questions of the survey are the following:



The screenshot shows a web-based survey form. At the top, it has a title 'Teaching assignment web tool' and a URL <http://gestio-docent.byethost7.com>. Below the URL, there is a red asterisk followed by the word 'Obligatòria'. The main question is 'Rate the web app interface: *'. Below this question, there are five radio buttons labeled 1, 2, 3, 4, and 5. To the left of the first radio button (1) is the word 'Poc' and to the right of the fifth radio button (5) is the word 'Molt'. The radio buttons are arranged in a horizontal line.

Rate the web app usability: *

1 2 3 4 5

Poc ☐ ☐ ☐ ☐ ☐ Molt

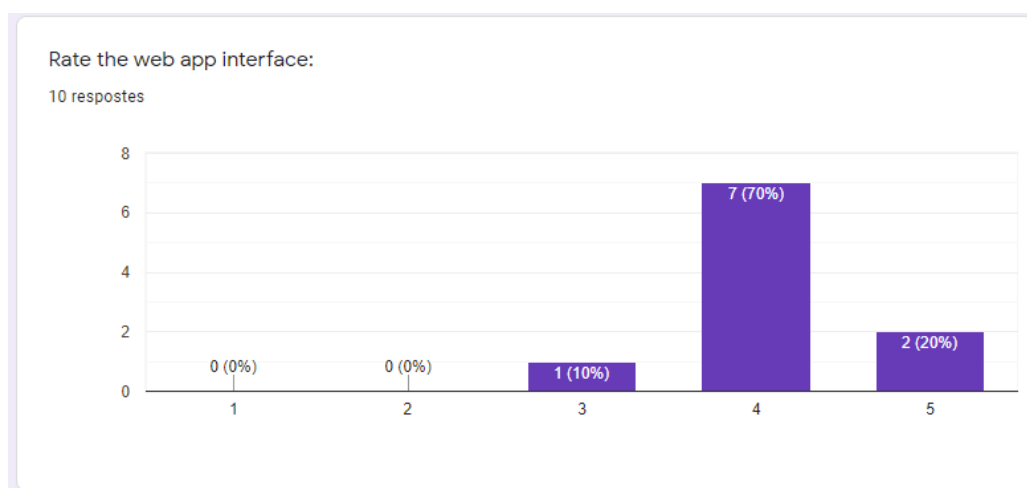
Do you think the web application is an improvement over the previous situation?
(before, this task was performed using an Excel file that each teacher had to fill and send to the person in charge of this task. Then each time this person received it he had to send it to the next teacher in the order list and be aware that everyone filled it correctly) *

☐ Yes

☐ No

Figure 22: Survey questions. Source: Google Forms.

The results of the survey can be found in Fig. 23:



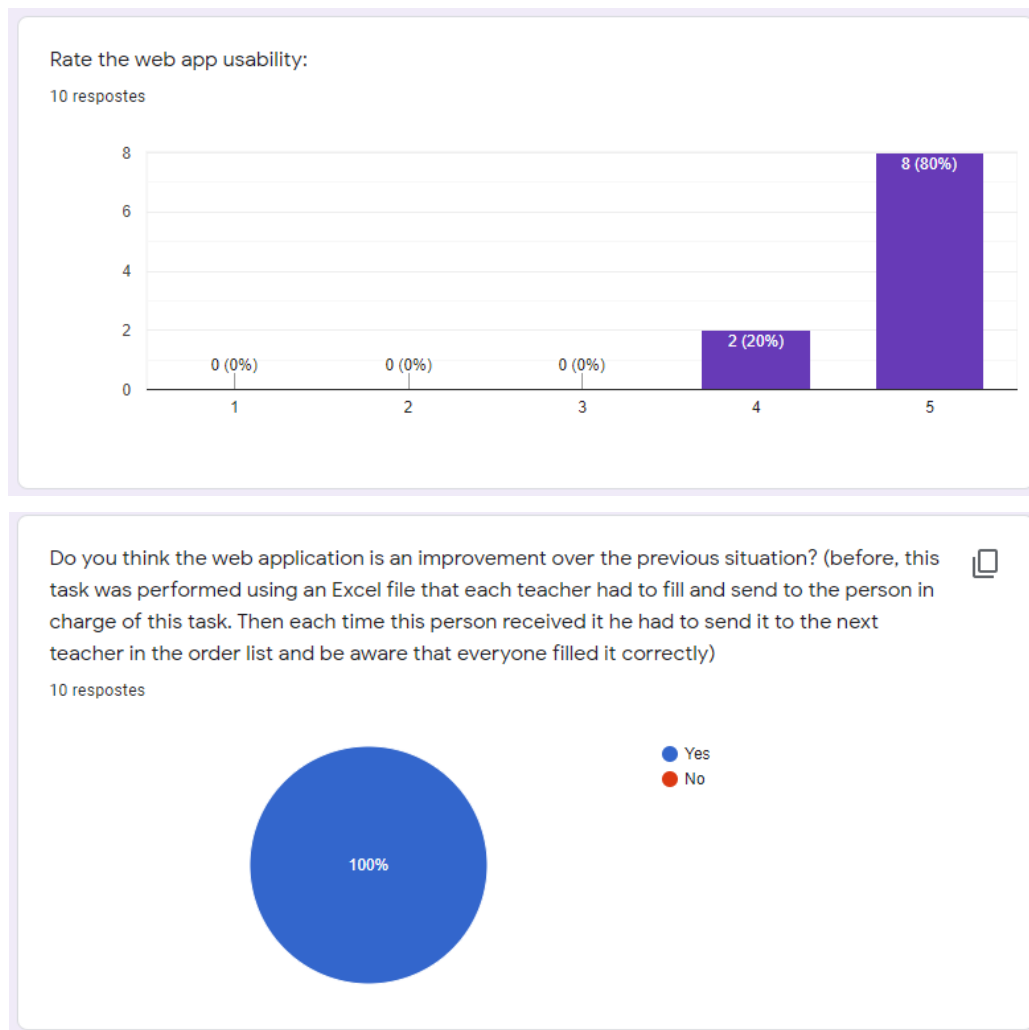


Figure 23: Survey results. Source: Google Forms.

As it can be seen, the web interface got an average rate of 4.1 out of 5 and the web usability 4.8 out of 5. All the answers to the question “Do you think the web application is an improvement over the previous situation?” have resulted to be Yes.

The conclusions that can be extracted from these results are that the web application interface could be a little better but people like it, it has a high rate but not the maximum. It could be a good option in future versions to add some visual and more attractive features. The tool usability has almost the maximum average rate, which means that it is easy to use and intuitive. In addition, it is confirmed that the web tool is an improvement over the previous situation since all the answers were affirmative.

Some of the errors/bugs found and solved later are the following:

- Accents: when uploading the CSV file of professors or subjects, when there was an accent it was not displayed correctly.
- Intro: in some Excel versions, when uploading the files an intro space was added at the end of the words, causing errors when trying to log in or other functionalities.
- Excel: the Excel was not well displayed due to a rounding error, when the subject has 1.5 ECTS credits the value was rounded to 1.
- Mobile: when accessing the web application through a mobile phone, the navigation bar on top did not work.

7. Budget

To calculate the project budget, the cost of the work if it has been done as it would be in a real company and according to the salary of the workers needed to perform the different tasks that make up the project and the number of hours assigned to each of them.

It is taken into account that, since the material and software required to develop the project are conventional and do not require any license or device with special characteristics, the salary of the employees already includes the cost of depreciation of the used hardware material and the energy consumed, since this methodology represents the way in which companies act when budgeting a project. However, these costs will be mentioned for purely informative purposes and will also be presented. The total budget will be calculated only with the human resources and software costs.

7.1. Human resources

The completion of the project in a company requires at least 3 workers:

- A project manager and analyst: is responsible for managing the project, maintaining contact with the client and performing a functional analysis.
- A senior programmer and architect: is responsible for choosing the technology used in each part of the program and programming the most difficult parts.
- A junior programmer: is responsible for programming the server.

Despite the fact that in this project all the roles are assumed by the same person, the 3 roles are different in order to calculate the approximate cost of the work performed. The following table shows the tasks performed and the number of hours spent by each worker:

Role	Task	Hours	Total hours
Project manager and analyst	Collect requirements	5	130
	Functional analysis	30	
	Functional tests	15	
	Preliminary validation	20	
	Write documentation	60	
Senior programmer and architect	Design architecture	25	35
	Set up	10	
Junior programmer	Previous server knowledge	40	140
	Programming server	80	
	Test	20	
Total			305

Figure 24: Dedicated hours table.

The following is the total cost of differentiated human resources per worker. The cost of the worker has been calculated taking into account that he works 1800 hours/year, as the agreement usually applies, and adding to the salary he receives an extra 33% that the company has to pay to social security.

Role	Working hours	Cost [€/h]	Total [€]
Project manager and analyst	110	26,6	3458
Senior programmer and architect	35	24,38	853,3
Junior programmer	140	13,3	1862

Figure 25: Project members salary table.

7.2. Hardware

The depreciation cost of the material used to develop the project is shown in the following table.

Product	Price [€]	Shelf life [years]	Usage time [months]	Amortization cost [€]
Laptop	800	6	4	44.44

Figure 26: Amortization cost table.

The router amortization cost is neglected due to its multiple utilities and long service life.

The calculations were made using the following formula:

$$\text{Amortization cost} = \text{Price} \cdot \text{Usage time} / \text{Shelf life}$$

7.3. Software

The cost of the different software installed is shown in the following table:

Software	Cost [€]
Windows	0*
XAMPP	0
NetBeans	0
JavaSDK	0
Github Desktop	0
Google Chrome	0

Figure 27: Software cost table.

* Windows has an additional cost if you want to install it separately, but it is not taken into account as it was installed with the computer and, therefore, its cost has not been paid and in any case would be included in the amortization cost of the computer.

It can be seen that the cost is € 0, since all software used (except Windows) are open source.

7.4. Indirect costs

The indirect costs of this project are basically the costs of the electrical energy of the appliances used. The costs associated with the consumption of the router and Internet access are not taken into account, since both concepts are associated with multiple uses not related to the present project and the number of daily hours in which it is used for this purpose it is not very high, which is why they are considered negligible.

Thus, taking into account the average price of electricity in Spain (about 0.14 € / kWh), the time of use of each device and its electricity consumption, the average cost of electricity consumption of this project is calculated through the following formula:

Cost of electricity consumption [€] = power consumption [kW] • usage time [h] • cost of electricity [€ / kWh]

It is assumed that the computer consumes 50 W and has been used in all tasks performed, that is during 305 hours in total.

Device	Usage time [h]	Total cost [€]
Computer	305	2,14

Figure 28: Electrical energy cost table

7.5. Total

The total costs can be seen in the following table:

Concept		Cost [€]
Human resources	Project manager and analyst	3458
	Project manager and analyst	853,3
	Junior programmer	1862
Software		0
Total		6173

Figure 29: Total cost table.

The total cost of the project amounts to 6173,3€.

8. Environmental and social impact

The environmental impact of the project is very low since it has been developed from beginning to end virtually. Therefore, its impact is linked to the materials used for the construction of the computer used for developing the web app and the energy it has spent during its usage. However, since any new device has been purchased for doing so, and the energy impact is not significant, the environmental costs of the project could be neglected.

Regarding the social impact of the project, it is positive since it provides a new tool that makes a task easier for the people. Not only it makes easier for the professors to choose their teaching groups, it lets them change their decision at any time and notifies them when they can do it, it also helps the person responsible of managing the teaching assignment by automatizing the process.

9. Future improvements

Since it is a first version of the web application, there are still some aspects that could be improved in the future. First, the Department of Electronic Engineering will try the tool and if the test is successful the following implementations could be added:

- Use it in other departments: it would also be very useful for other departments of the school and it would not be necessary to make a lot of changes.
- Multilanguage: at the moment, the website is only in Catalan, it would be helpful to also have it in Spanish and English.
- Develop an Android/Ios application: currently the web application can be accessed through mobile phone but the interface is the same as the laptop version but adapted by Bootstrap for mobile phone. An application would be a better option since the interface would have been thought for mobile phone, specifically so it would be more intuitive and practical.
- Improve interface: as the results in the survey of the preliminary validation showed, the interface could be a little more user-friendly. A different format or color distribution would be an option.

Conclusions

Once the design, implementation and preliminary validation of the web application have been completed, it can be concluded that the project objectives have been largely achieved. It has been obtained a functional website, with simple but careful and coherent aesthetics, that is a clear improvement compared to the method used before.

Although developing such a web application is not a major technological difficulty in the professional world, it is a great challenge for the student to learn a new programming language from scratch (PHP) and the technologies required to create it as well as learn and understand the server, the database and make the connections with the different web pages.

Some subjects from the Industrial Engineering Bachelor Degree have been useful for completing this project. First of all, although Python is the programming language learned through two different Computer Science subjects, having some prior knowledge in programming is fundamental in order to understand and learn faster other programming languages, PHP in this case.

Then, thanks to the student's choice in Project I and II, some basic knowledge of HTML and CSS languages and server connections were previously acquired. All of them essential in web development. Finally, an internship of 6 months in a company dedicated to digital marketing provided learning of SQL.

The project, therefore, has achieved its goals and has been very beneficial for the student, who has gained very valuable knowledge for his professional future, as well as at a personal level, making a small contribution in the ETSEIB School.

Acknowledgments

To my family, for their help and involvement in this project. It would not have been possible without them.

To my tutor, for his dedication, patience and guidance through all the process and for being the first tester of it.

To the 9 friends and familiars that have tried the web application and answered the survey sincerely.

Bibliography

Bibliographic references

- [1] European Commission, *European credit transfer and accumulation system* *ects*. Available at: https://ec.europa.eu/education/resources-and-tools/european-credit-transfer-and-accumulation-system-ects_en
- [2] BEN SHAPIRO, Website vs. Web Application: What's the Difference?, April 16th 2013. Available at: <https://www.seguetech.com/website-vs-web-application-whats-the-difference/>
- [3] NICOLE FERGUSON, *What's The Difference Between Frontend And Backend Web Development?*, updated on July 17th 2018. Available at: <https://careerfoundry.com/en/blog/web-development/whats-the-difference-between-frontend-and-backend/>
- [4] LAUREN STEWART, *Front End vs Back End Development*, January 14th 2019. Available at: <https://www.coursereport.com/blog/front-end-development-vs-back-end-development-where-to-start>
- [5] MAHESH CHAND, *Most Popular Databases In The World*, updated on July 26th 2019. Available at: <https://www.c-sharpcorner.com/article/what-is-the-most-popular-database-in-the-world/>
- [6] SANDRA PARKER, *Top Backend Programming Languages of 2019*, June 13th 2019. Available at: https://medium.com/@Sandra_Parker/top-backend-programming-languages-of-2019-dc8d81836b2d
- [7] PHP documentation. Available at: <https://www.w3schools.com/php/>
- [8] Netbeans documentation. Available at: <https://netbeans.org/kb/>
- [9] XAMPP documentation. Available at: https://www.apachefriends.org/faq_windows.html
- [10] Bootstrap documentation. Available at: <https://getbootstrap.com/docs/4.3/getting-started/introduction/>

- [11] Java SDK documentation. Available at:
<https://www.oracle.com/technetwork/java/javase/documentation/index.html>
- [12] PhpMailer documentation. Available at:
<https://github.com/PHPMailer/PHPMailer/wiki/Tutorial>
- [13] PhpSpreadsheet documentation. Available at:
<https://phpspreadsheet.readthedocs.io/en/latest/>
- [14] Github desktop documentation. Available at: <https://desktop.github.com/>

Complementary bibliography

- YOUTUBE, Java Dev One, Curso diseño web principiantes 00 - introducción - HTML, PHP y Bootstrap, 2015. Available at:
<https://www.youtube.com/watch?v=9d2BJsxff-q&list=PLN9W6BC54TJl6mlpAT-b7nDkpHbZp73i4&index=1>

In order to start learning PHP and for the web application in general, a course of Youtube videos of web development for begginers was very useful. This course has 80 videos in total but the first 15 were enough for this purpose.

- YOUTUBE, Cool Inventor, PHP MySQL login form with XAMPP, 2018. Available at:
<https://www.youtube.com/watch?v=G9gkh4GRUAY&t=>

This video was used for implementing the login functionality correctly.

- Stack Overflow. Available at: <https://stackoverflow.com/>

Stack overflow is a very large online community for developers to learn and share their programming knowledge. There, people ask for help in different programming issues to other users of the community. Almost every little problem that a programmer can face during a project is already solved in this community.