Report of Web Engineering

GitHub: HerrAugust

23.09.2017

1 Table of contents

Contents

2 Introduction

The chosen project is *Course Web*. A website that basically reports a list of courses for the University is the core of it. It is also possible to have a look at the detail of courses and add users and courses in the backoffice. The source code of can be found at GitHub; the complete document of this project can be found under the folder *documentation*.

This report is divided into 2 parts. The first one is about design and the second one about the implementation. There, you can find details about technical aspects, like some related to frameworks and to the use of HTML5.

3 Part 1 - Design

3.1 Use case diagram

Given the text of the project (see above for a reference), only the mandatory parts are considered (i.e., everything but *optional* parts). You can see the Use Case Diagram in Figure 1. It is also available as an image (PNG) in the *documentation* folder in GitHub, so that the reader can more easily see it.

3.2 Data Model

The Data Model is reported in Figure 2 as a Class Diagram. A description of each entity/class follows.

3.2.1 Course

Sentences related to the entity Course are:

1. *Each course has [...] basic data: name, code, scientific sector (SSD), language, semester* which are the attributes of the entity Course

2. Lectures list (thus, the link to entity Teacher is inserted)

3. *Course description: prerequisites, learning outcomes, assessment method, teaching method, syllabus[...]* which become attributes of the entity Course

4. *Textbook* and consequently the connection to Book is inserted

5. *Relation with other courses: introductory courses, same-as courses, modules* and consequently the relations from Course to Course are inserted

6. *External links: course homepage, external resources, forum/eLearning* for this, course homepages and forum become attributes and for external resources a connection to a third entity is added

7. *Notes* becomes an attribute

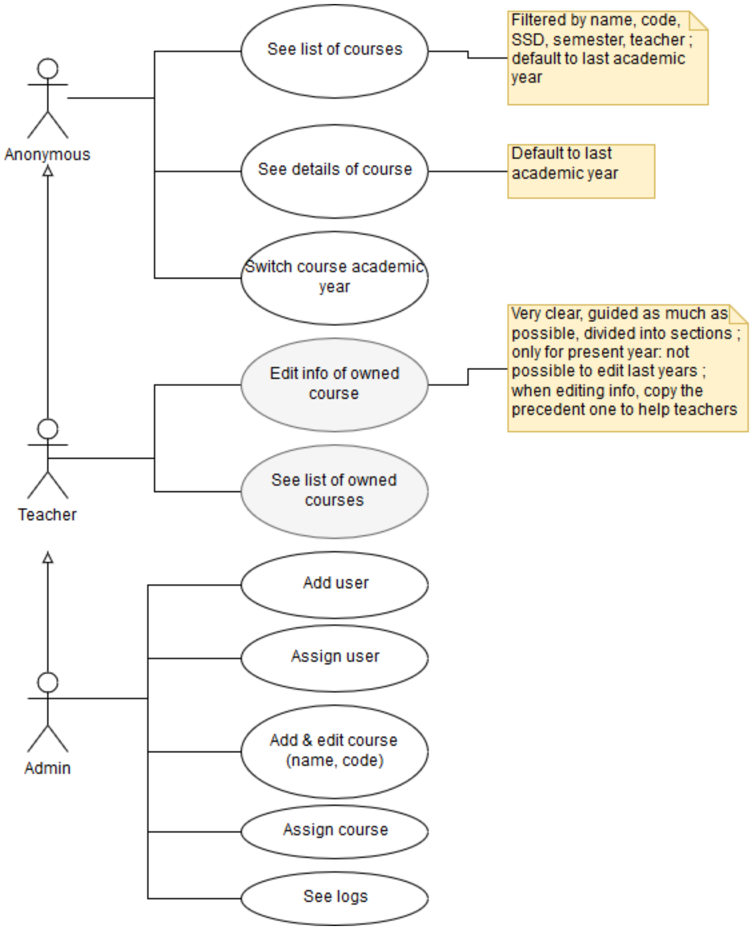

Anonymous
Teacher
Admin
See list of courses
See details of course
Switch course academic year
Edit info of owned course
See list of owned courses
Add user
Assign user
Add & edit course (name, code)
Assign course
See logs
Filtered by name, code, SSD, semester, teacher ; default to last academic year
Default to last academic year
Very clear, guided as much as possible, divided into sections ; only for present year: not possible to edit last years ; when editing info, copy the precedent one to help teachers

Figure 1: Use case diagram

8. *The site should support bilingual publication [...] and you design the site so that it allows to enter information in both languages in the back-office [...] and display in the front-office*

9. *In order to create a "virtual guide" the system must collect all this information year by year. This means that you should create a system where all the information above is associated with an academic year, and there can be several copies for different academic years.* and consequently the attribute "academic year" is inserted to Course.

Notice that for each attribute but id, code, SSD, name, language, semester, photo there should be the translation in both Italian and English, which is not reported to restrict the image.

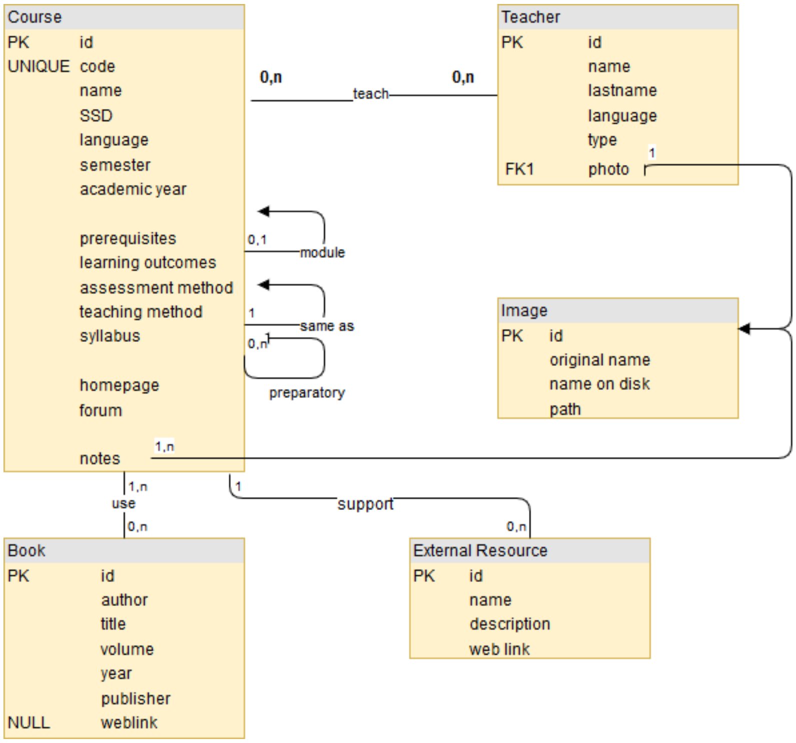


Figure 2: The Data Model

3.2.2 Teacher

Sentences related to the entity Teacher are:

1. *The system has three types of users: anonymous, teacher and administrator. Only administrators can register new users and assign them a type.*, thus the attribute *type* is inserted and it can only be either *anonymous, teacher or administrator.*

3.2.3 Image

There is no reference to images in the text, but it can be a good idea to add it to complete it a little bit.

3.2.4 Book

Sentences related to the entity Book are:

1. *Each course has [...] textbooks (author, title, volume, year, publisher, web link if available).* Thus, Course is associated with Book and Book gets the attributes in brackets.

2. *the system must collect all this information year by year. This means that you should create a system where all the information above is associated with an academic year, and there can be several copies for different academic years.* For this, the attribute "academic year" of Course is enough to find the courses.

3.2.5 External Resource

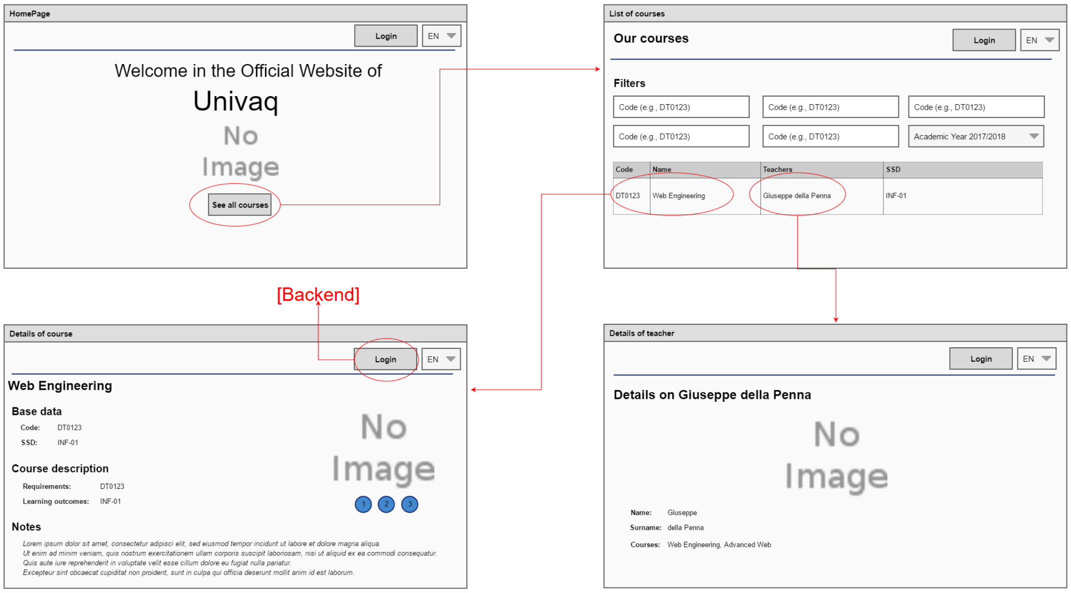
Each course has [...] external links: course homepage, external resources, forum/eLearning. It seems that a course has many external resources (and so it is an entity connected to Course) and that each is a link. Thus, an External Resource has a link, a name and a description.

4 Navigation Model

The navigation is quite simple but, according to who is writing, also accessible and usable. Moreover, it is not simplistic, it is simply what is needed.

An overview of the frontend is in Figure 3 and in Figure 4 there is an overview of the backend.

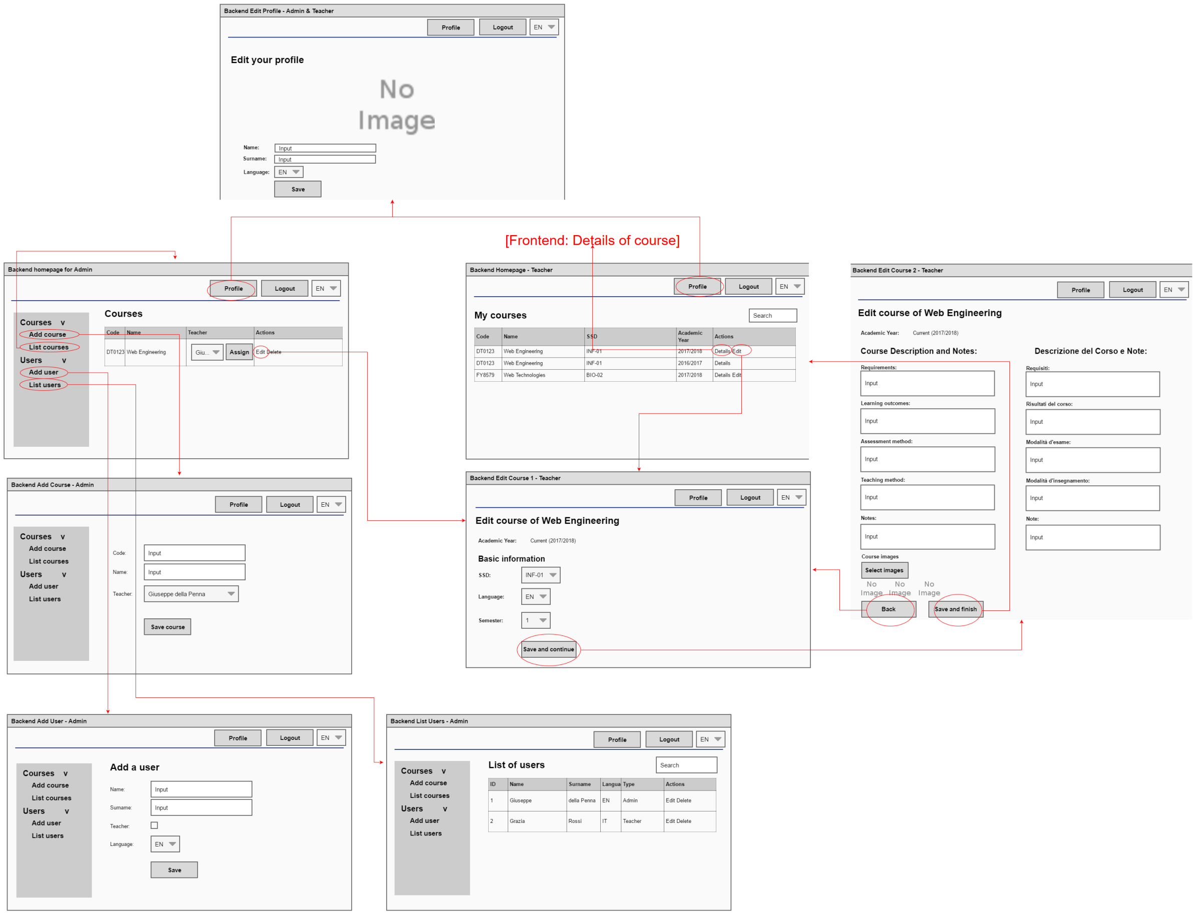
For all the pages, the following sentence is to take into account: *The site has to support bilingual publication(Italian and English) of all its content.* That's why there is a button in all pages that the user can click to switch language.



Code	Name	Teachers	SSD
DT0123	Web Engineering	Giuseppe della Penna	INF-01

*Lorem ipsum dolor sit amet, consectetur adipisci elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua.
Ut enim ad minim veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur.
Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.
Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.*

Figure 3: Navigation Model of the frontend



Code	Name	Teacher	Actions
DT0123	Web Engineering	Giu... Assign	Edit Delete

Code	Name	SSD	Academic Year	Actions
DT0123	Web Engineering	INF-01	2017/2018	Details Edit
DT0123	Web Engineering	INF-01	2016/2017	Details
FY8579	Web Technologies	BIO-02	2017/2018	Details Edit

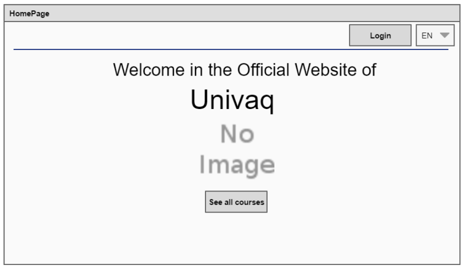
ID	Name	Surname	Langua	Type	Actions
1	Giuseppe	della Penna	EN	Admin	Edit Delete
2	Grazia	Rossi	IT	Teacher	Edit Delete

Figure 4: Navigation Model of the backend

4.1 Frontend Pages

This section shows all the pages of the frontend.

4.1.1 Homepage



Welcome in the Official Website of

Univaq

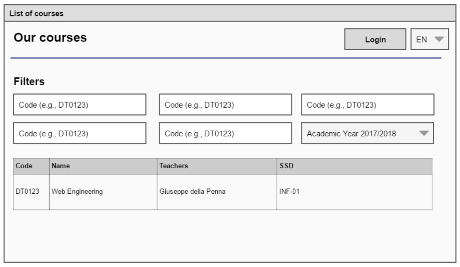
Figure 5: Homepage of frontend

Sentences related to the homepage are:

1. *The system has three types of users: anonymous, teacher and administrator.* That's why there is the login button.

The page is minimalist but it works: to present the website and show its main actions, which in this case is just to show courses.

4.1.2 List of courses



Code	Name	Teachers	SSD
DT0123	Web Engineering	Giuseppe della Penna	INF-01

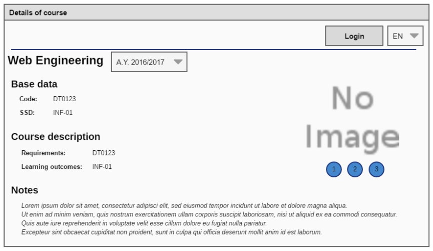
Figure 6: Page with all courses of frontend

Sentences related to this page are:

1. *All the users can view the complete list of courses, possibly filtered by name (also partial), code, SSD, semester, teacher, language and degrees for which it is available (if you have modeled such information). Clicking on a course will display its full record..* Thus, there are some inputs that the user can fill to filter courses.

Only the essential information about courses are shown. The user can get more by clicking on one of them.

4.1.3 Details of course



Web Engineering

Base data

Code: DT0123

SSD: INF-01

Course description

Requirements: DT0123

Learning outcomes: INF-01

Notes

Lorem ipsum dolor sit amet, consectetur adipisci elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua.
Ut enim ad minim veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur.
Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.
Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Figure 7: Page with details of a course of frontend

Sentences related to this page are:

1. *the system must collect all this information year by year. This means that you should create a system where all the information above is associated with an academic year, and there can be several copies for different academic years.*. Thus, there is the button of the Academic Year that users can click.

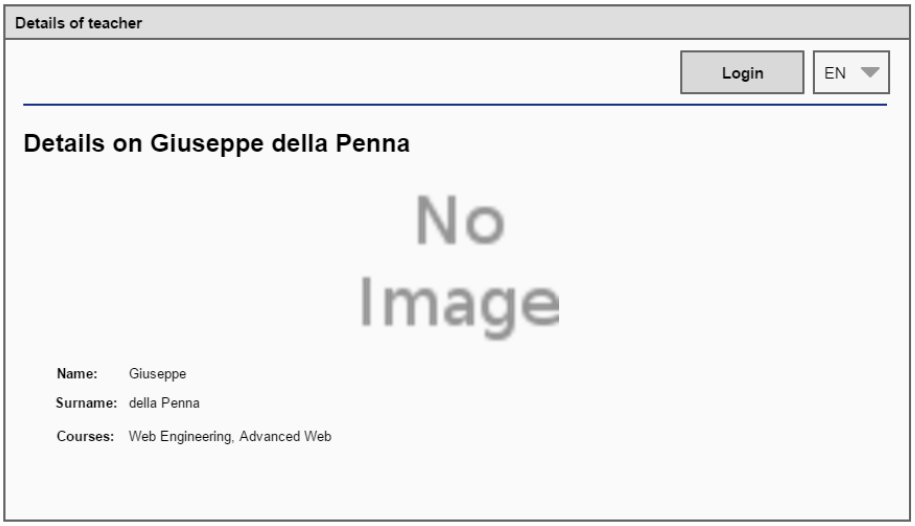
2. *Of course, the course page must be very well-finished, to make the contained information as clear and accessible as possible. You can split the data across multiple pages, remembering, however, that the essential information must be accessible with a small number of clicks.*. Thus, sections (Base data, Course Description, Notes) try to guide the user. Also images could be useful to help students to understand the content of the course.

3. *the information presented in a course page are, by default, those related to the last academic year (which must be indicated on the page) where the course has been updated.*

4. *the user should be able to choose an academic year and read the course information related to such academic year, if available (e.g., through controls on the course page, or using suitable filters on the course list).*

4.1.4 Details of teacher

There is no sentence in the text for this page. It was added as an additional content.



Details on Giuseppe della Penna

Name: Giuseppe

Surname: della Penna

Courses: Web Engineering, Advanced Web

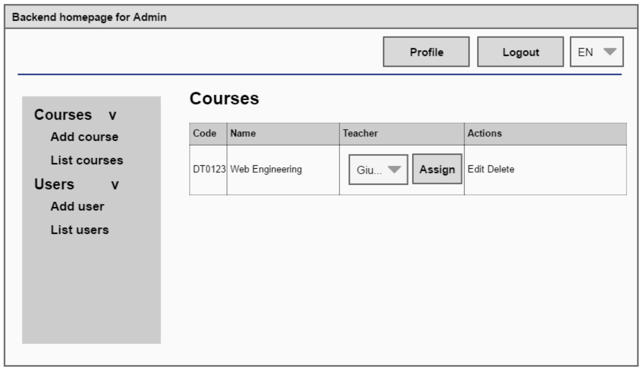
Figure 8: Page with the details of a teacher of frontend

4.2 Backoffice Pages

This section shows all the pages of the backoffice.

4.2.1 Backend homepage for Admin

This page is specific for Admins.
This web page is intented to be a homepage, i.e. a page to have a summary of the main functionalities. These are presented in the Use Case Diagram (see Figure 1). Because *Teachers can enter and edit all information belonging to their assigned courses, while administrators can modify all the courses*, admins can decide to modify courses from this page (Edit button).

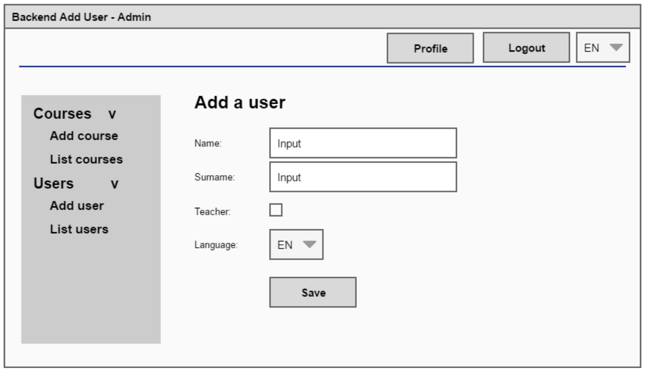


Code	Name	Teacher	Actions
DT0123	Web Engineering	Giu... Assign	Edit Delete

Figure 9: Homepage of backend for Admins

4.2.2 New User Page for Admin

This page is specific for Admins.
Sentences related to this page are:



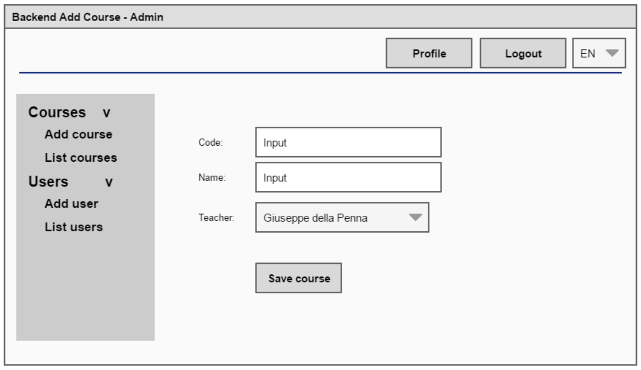
Name: Input
Surname: Input
Teacher: □
Language: EN

Figure 10: Page to create new users for Admins

1. *Only administrators can register new users and assign them a type.*

4.2.3 New Course Page for Admin

This page is specific for Admins.
Sentences related to this page are:



Code: Input

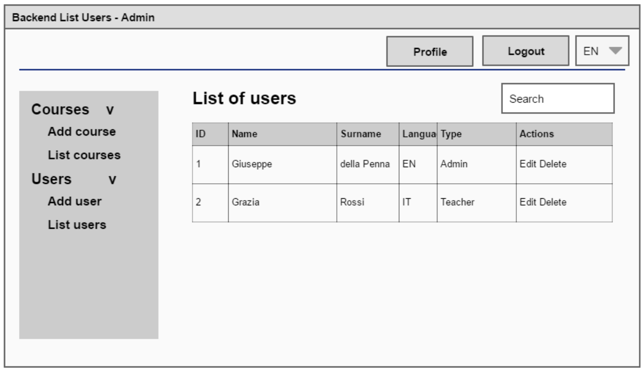
Name: Input

Teacher: Giuseppe della Penna

Figure 11: Page to create new courses for Admins

1. *Site administrators can create courses (with at least the essential information: name, code) and assign them to the teachers.* It is assumed that the names of courses can be in English only. This is especially useful to Erasmus students, and to local ones should not make a big difference.

4.2.4 Page List of Users for Admin



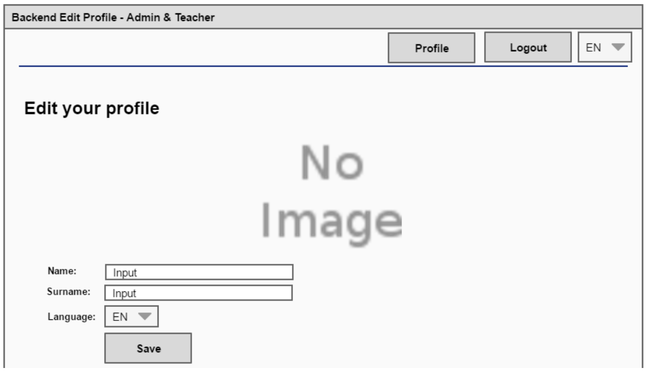
List of users

ID	Name	Surname	Langua	Type	Actions
1	Giuseppe	della Penna	EN	Admin	Edit Delete
2	Grazia	Rossi	IT	Teacher	Edit Delete

Figure 12: Page to list users for Admins

This page is specific for Admins.
There is no reference in the text to this page, but it is standard: the administrator may need to modify a user (teacher or admin).

4.2.5 Edit Profile for Admin and Teacher

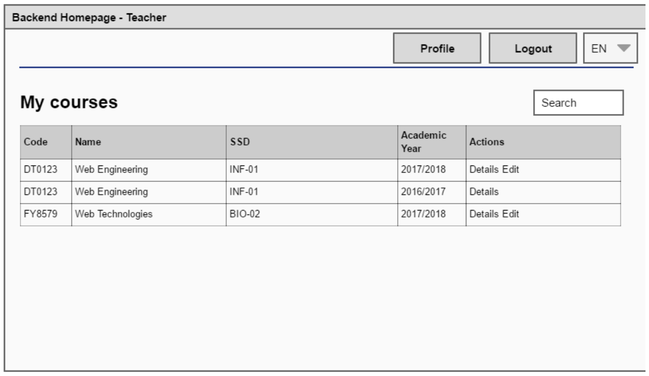


Name: Input
Surname: Input
Language: EN

Figure 13: Page to edit profile for Admins and Teachers

There is no sentence in the text regarding this functionality, but it is a standard one.

4.2.6 Backend Homepage for Teachers



Code	Name	SSD	Academic Year	Actions
DT0123	Web Engineering	INF-01	2017/2018	Details Edit
DT0123	Web Engineering	INF-01	2016/2017	Details
FY8579	Web Technologies	BIO-02	2017/2018	Details Edit

Figure 14: Homepage for Teachers

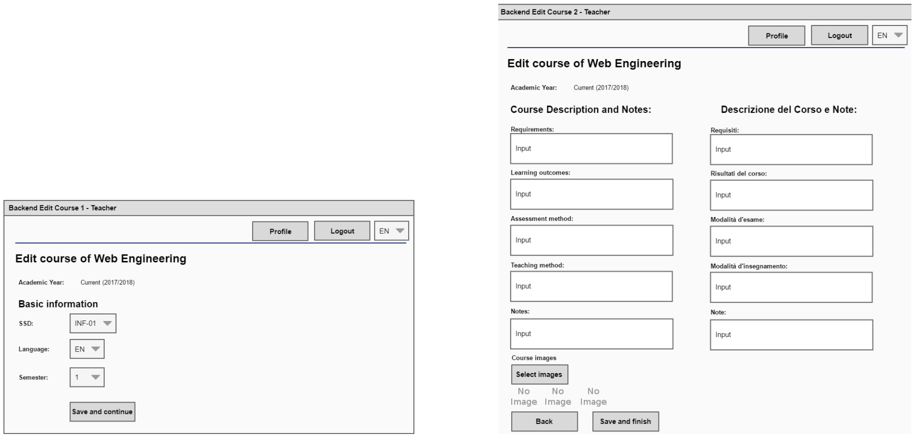
The idea of this page is that an admin creates the course, the teacher access his homepage and he can edit the information about his course. It could be useful to highlight the row of a course to be completed to enhance usability. Sentences related to this page are:

1. *Teachers can enter and edit all information belonging to their assigned courses, while administrators can modify all the courses.* Thus, a list of courses is provided

2. *In the back office, the information entered or edited by the teacher are always related to the current academic year.* Thus, teachers cannot use the action ”Edit” for courses of previous years.

3. *the system must collect all this information year by year. This means that you should create a system where all the information above is associated with an academic year, and there can be several copies for different academic years.* Thus, information of past versions of the same course are kept stored.

4.2.7 Page to edit a course for Teachers and Admins



Backend Edit Course 1 - Teacher

Profile | Logout | EN

Edit course of Web Engineering

Academic Year: Current (2017/2018)

Basic information

SSD: INF-01

Language: EN

Semester: 1

Save and continue

(a) Page 1/2

Backend Edit Course 2 - Teacher

Profile | Logout | EN

Edit course of Web Engineering

Academic Year: Current (2017/2018)

Course Description and Notes:

Requirements: Input

Learning outcomes: Input

Assessment method: Input

Teaching method: Input

Notes: Input

Course images

Select images

No Image No Image No Image

Back | Save and finish

Descrizione del Corso e Note:

Requisiti: Input

Risultati del corso: Input

Modalità d'esame: Input

Modalità d'insegnamento: Input

Note: Input

(b) Page 2/2

Figure 15: Page to edit a course for Teachers and Admins

Sentences related to these pages are:

1. *The back-office for teachers should also be very intuitive. It may be useful to provide contextual help to clarify the meaning of the various fields, and possibly split the course information into logical segments to be compiled in a wizard-style process. Where possible (e.g., in the selection of related courses, of languages, semesters, SSD, types of credits, etc.) the input should be assisted or guided.*. To make this page clearer, it was splitted into two. In the first one, only basic information are required, in the other one some others. All information about a course should be filled. Each input has a label and a placeholder (it will be an example of valid input).

2. *in the back office, the information entered or edited by the teacher are always related to the current academic year.* To make this part clear to teachers, the page will show the message *Academic Year: Current*

3. *The site has to support bilingual publication (Italian and English) of all its content. Therefore [...] you should design the site so that it allows to enter information in both languages in the back-office.* Therefore, two columns of inputs are shown. I think it is a convenient way to question information because this way it is easier for a teacher to have both version in a single page, thus making it easier to translate by reading the counterpart.

4.3 Palette

Because the main color of the flag of the University of L'Aquila is red, this is also the main color of the palette, in Figure 16.

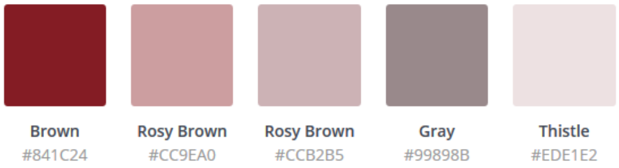


Figure 16: Palette of the website

5 Part 2 - Implementation

For the client, the following technologies were used: HTML5, CSS3, Javascript, JQuery, Bootstrap; for the server, instead, Java Servlet (v.8), MySQL, Tomcat, Freemarker as Template Engine.

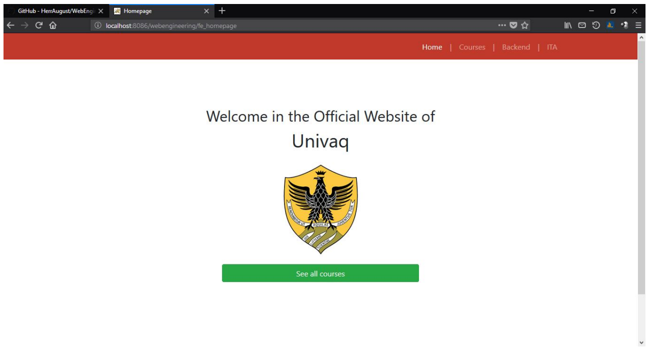
The site has been tested under Firefox 58, Chrome 64.0, Microsoft Edge 41.16299. The conformance to HTML5 polyglot has been tested under HTML-Validator for Firefox. HTML5 is the most robust solution for HTML5 pages.

Some attributes were added, like email and password for Teacher. These were necessary for Login. It is also assumed that nobody can insert a course of another academic year. You can only add a course for the current one.

Logging is also performed using the Java library. You can find them inside forlder web.

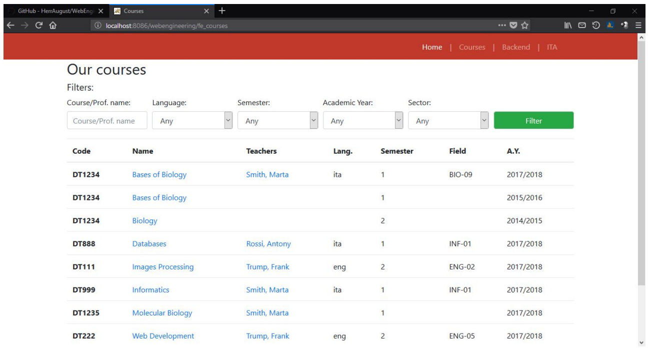
Notice how this website tries to avoid as much as possible Javascript, because someone may disable it for enhanced security or other reasons. Javascript is only used in the backend, for the function Search (searching in a table for general information is too complex to do with Java, because you have to manually search inside each single column) and for entering a course description (Login as you want - modify a course - press Continue), where FormData is used to group different information (textareas, text input and images). Thus, this website shows how you can do everything by using HTML5 forms only and CSS3/Bootstrap to arrange them appropriately.

Some previews of the website follow:


GitHub - HerrAugust/WebEngi
Homepage
localhost:8086/webengineering/fe_homepage
Home | Courses | Backend | ITA
See all courses

Welcome in the Official Website of
Univaq

Figure 17: Homepage of the website - frontend


GitHub - HerrAugust/WebEngi
Courses
localhost:8086/webengineering/fe_courses
Home | Courses | Backend | ITA

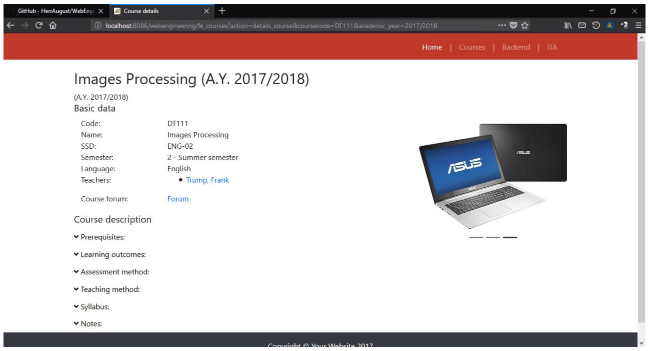
Our courses

Filters:

Course/Prof. name:	Language:	Semester:	Academic Year:	Sector:	
Course/Prof. name	Any	Any	Any	Any	Filter

Code	Name	Teachers	Lang.	Semester	Field	A.Y.
DT1234	Bases of Biology	Smith, Marta	ita	1	BIO-09	2017/2018
DT1234	Bases of Biology			1		2015/2016
DT1234	Biology			2		2014/2015
DT888	Databases	Rossi, Antony	ita	1	INF-01	2017/2018
DT111	Images Processing	Trump, Frank	eng	2	ENG-02	2017/2018
DT999	Informatics	Smith, Marta	ita	1	INF-01	2017/2018
DT1235	Molecular Biology	Smith, Marta		1		2017/2018
DT222	Web Development	Trump, Frank	eng	2	ENG-05	2017/2018

Figure 18: List of courses - frontend


GitHub - HerrAugust/WebEngi
Course details
localhost:8086/webengineering/fe_courses?action=details_course&coursecode=DT111&academic_year=2017/2018
Home | Courses | Backend | ITA
ASUS

Images Processing (A.Y. 2017/2018)

(A.Y. 2017/2018)

Basic data

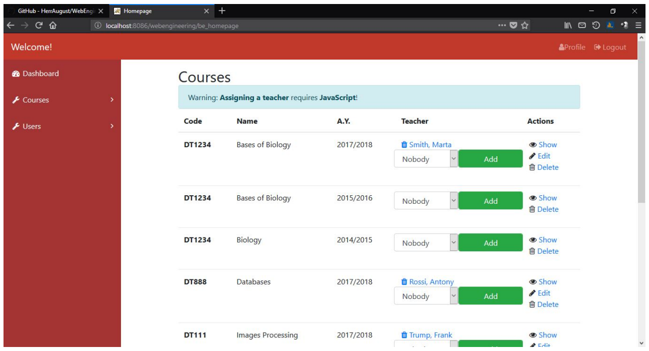
Code:	DT111
Name:	Images Processing
SSD:	ENG-02
Semester:	2 - Summer semester
Language:	English
Teachers:	• Trump, Frank
Course forum:	Forum

Course description

- Prerequisites:
- Learning outcomes:
- Assessment method:
- Teaching method:
- Syllabus:
- Notes:

Copyright © Your Website 2017

Figure 19: Course details - frontend

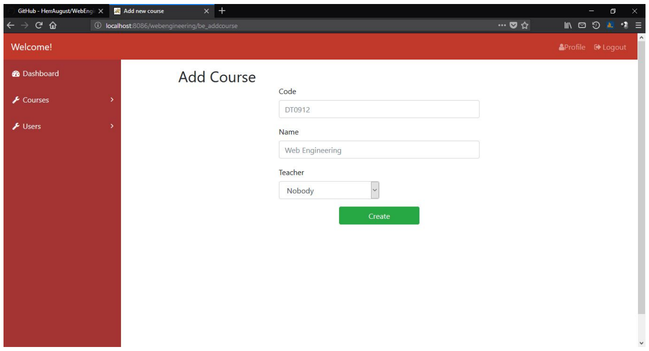

GitHub - HerrAugust/WebEngi
Homepage
localhost:8086/webengineering/be_homepage
Welcome!
Profile
Logout
Dashboard
Courses
Users

Courses

Warning: **Assigning a teacher** requires **JavaScript**!

Code	Name	A.Y.	Teacher	Actions
DT1234	Bases of Biology	2017/2018	Smith, Marta Nobody Add	Show Edit Delete
DT1234	Bases of Biology	2015/2016	Nobody Add	Show Delete
DT1234	Biology	2014/2015	Nobody Add	Show Delete
DT888	Databases	2017/2018	Rossi, Antony Nobody Add	Show Edit Delete
DT111	Images Processing	2017/2018	Trump, Frank	Show Edit

Figure 20: List of courses - backend


GitHub - HerrAugust/WebEngi
Add new course
localhost:8086/webengineering/be_addcourse
Welcome!
Profile
Logout
Dashboard
Courses
Users
Create

Add Course

Code

DT0912

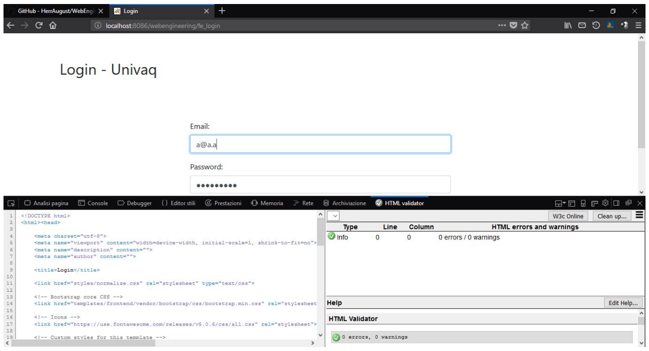
Name

Web Engineering

Teacher

Nobody

Figure 21: Form to add a new course - backend



Login - Univaq

Email:

a@a.a

Password:

●●●●●●●●●

```
<!DOCTYPE html>
<html><head>

    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <meta name="description" content="">
    <meta name="author" content="">

    <title>Login</title>

    <link href="styles/normalize.css" rel="stylesheet" type="text/css">

    <!-- Bootstrap core CSS -->
    <link href="templates/frontend/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet

    <!-- Icons -->
    <link href="https://use.fontawesome.com/releases/v5.0.6/css/all.css" rel="stylesheet">

    <!-- Custom styles for this template -->
```

Type	Line	Column	HTML errors and warnings
✓ Info	0	0	0 errors / 0 warnings

Help

HTML Validator

✓ 0 errors, 0 warnings

Figure 22: How the conformance to standards has been tested - HTMLValidator for Firefox

List of Figures