Development of an Educational SharePoint portal for coding students

Author: Gerard Colomer Castelló
Director: Lluís Solano Albajes
September 2016

Escola Tècnica Superior
d'Enginyeria Industrial de Barcelona
Summary

The project will explain what is SharePoint, why is used and who does use it. It will also expose some alternatives, compare them to SharePoint and expose the pros and cons. The main objective of this project will be developing an educational SharePoint portal.

Using this portal, coding students will be able to share their solutions to different exercises, vote the best solutions and comment them.

This portal will be developed using only software and servers obtained legally without any cost. As can be seen in this project, this will mean that all the coding will be at the client side.

This project will use mainly SharePoint, ASP, HTML, CSS and JS. Some third party libraries with creative commons will be also used.

This project also pretends to set the bases to expand this portal to other subjects to create a more collaborative studying environment in a future. Finally, an approximation of how much could the development of a portal with these characteristics cost.
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>SUMMARY</td>
<td>1</td>
</tr>
<tr>
<td>1. GLOSSARY</td>
<td>5</td>
</tr>
<tr>
<td>2. PREFACE</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>2.1. Motivation</td>
</tr>
<tr>
<td></td>
<td>2.2. Previous requirements</td>
</tr>
<tr>
<td>3. INTRODUCTION</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>3.1. Project objectives</td>
</tr>
<tr>
<td></td>
<td>3.2. Project Scope</td>
</tr>
<tr>
<td>4. PROJECT DEVELOPMENT PLAN</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>4.1. Iteration 1  (Starting date: 22/04/2016)</td>
</tr>
<tr>
<td></td>
<td>4.2. Iteration 2  (Starting date: 10/05/2016)</td>
</tr>
<tr>
<td></td>
<td>4.3. Iteration 3  (Starting date: 22/06/2016)</td>
</tr>
<tr>
<td></td>
<td>4.4. Final followed process</td>
</tr>
<tr>
<td>5. SHAREPOINT</td>
<td>16</td>
</tr>
<tr>
<td></td>
<td>5.1. What does SharePoint offer? Why SharePoint?</td>
</tr>
<tr>
<td></td>
<td>5.2. SharePoint Look and Feel</td>
</tr>
<tr>
<td></td>
<td>5.3. Alternatives and different versions of SharePoint</td>
</tr>
<tr>
<td></td>
<td>5.4. Portal Architecture</td>
</tr>
<tr>
<td></td>
<td>5.5. SharePoint 2016</td>
</tr>
<tr>
<td></td>
<td>5.6. SharePoint components that could be added by users</td>
</tr>
<tr>
<td>6. DESIGN</td>
<td>25</td>
</tr>
<tr>
<td>7. PAGE LAYOUTS</td>
<td>28</td>
</tr>
<tr>
<td></td>
<td>7.1. Home Page Layout</td>
</tr>
<tr>
<td></td>
<td>7.2. General Page Layout</td>
</tr>
<tr>
<td>8. PAGES</td>
<td>29</td>
</tr>
<tr>
<td></td>
<td>8.1. Home</td>
</tr>
<tr>
<td></td>
<td>8.2. Theory</td>
</tr>
<tr>
<td></td>
<td>8.3. Exercises</td>
</tr>
<tr>
<td></td>
<td>8.4. Exams</td>
</tr>
<tr>
<td></td>
<td>8.5. Tests</td>
</tr>
</tbody>
</table>
# 1. Glossary

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apache Cordova</td>
<td>Apache Cordova it’s an open source mobile application development framework.</td>
</tr>
<tr>
<td>CSS</td>
<td>CSS stands for Cascading Style Sheets. It’s a styling language that’s used for describing the look and feel of a webpage written in a markup language (HTML).</td>
</tr>
<tr>
<td>Embedded</td>
<td>Embedded code is code that’s inside other element. For example, we could have JavaScript code embedded inside an HTML web page.</td>
</tr>
<tr>
<td>Framework</td>
<td>A software framework provides generic functionalities that can be customized in order to develop products without having to start coding from zero.</td>
</tr>
<tr>
<td>HTML</td>
<td>HTML stands for Hypertext Markup Language. It’s a system to organize the content of a web page. It also provides identifiers for the behind code and the CSS.</td>
</tr>
<tr>
<td>Ionic</td>
<td>Ionic is a free and open source library of HTML, CSS and JS components that enables the developers to build web apps.</td>
</tr>
<tr>
<td>JQuery</td>
<td>JQuery is a JavaScript library that helps simplifying the HTML client-side. It’s widely used and it’s a free open-source software.</td>
</tr>
<tr>
<td>JS</td>
<td>JS stands for JavaScript: It’s a computer language used to create interactive effects within web pages.</td>
</tr>
<tr>
<td>Mobile Friendly/Responsive</td>
<td>A web site is mobile friendly or responsive when it’s easy to use on a mobile device (mobile phone or tablet) as well as in larger screens.</td>
</tr>
<tr>
<td>Office365</td>
<td>Office365 is a group of software and services that Microsoft sells as a pack to its subscribers. This software usually works online and includes Office365, Outlook, SharePoint…</td>
</tr>
<tr>
<td><strong>Out of the box</strong></td>
<td>It’s said of the immediate usability of functionalities that a product possesses.</td>
</tr>
<tr>
<td>-------------------</td>
<td>--------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>SharePoint</strong></td>
<td>SharePoint is a web application platform in the Microsoft Office server suite. It allows its users to create and share web pages, documents…</td>
</tr>
<tr>
<td><strong>SharePoint Feature</strong></td>
<td>SharePoint features make it easy to activate and deactivate functionalities. A feature can enable or disable a set of different functionalities of a SharePoint portal.</td>
</tr>
<tr>
<td><strong>Web Part</strong></td>
<td>A Web Part is a server control that usually just does one particular function. Web Parts are inserted in Web Part zones. Adding a web part to a web part zone can be done without any coding involved.</td>
</tr>
</tbody>
</table>
2. Preface

2.1. Motivation

When I studied the first subject where we had to code at the university I enjoyed every second of it but that was not the case for everyone. I found out that while some people enjoyed the subject and they find it really interesting others were struggling with it. Since currently I’m developing web portals at work I decided that it will be a good idea to develop a portal that could enable students to help each other with their doubts and problems. Right now at our university we have a “Moodle” that allows some communication between students but it’s social capabilities are limited and nearly anyone uses it to ask for help.

I decided that I wanted to build the portal using SharePoint since I’ve been working with SharePoint since the beginning of my internship at Sogeti.

SharePoint provides a lot of “Out of the box” functionalities. These functionalities are well designed and are really efficient but they are not mobile friendly. That means that the portal is not well designed for little screens like the ones you can find in mobile phones, tablets... That’s one of the reasons why most of the projects developed with SharePoint don’t use most of those elements.

In SharePoint you can use a lot of different web parts. This project will be centered mostly with the newsfeed and documents web parts. It’s a good practice to use this out of the box elements, so I wanted to make these elements mobile friendly using only styles (CSS and JS). It was a challenge.

Finally, I wanted to write this project in English so more people could consult it and I could practice my English in order to improve it.
2.2. Previous requirements

The portal has to be easy to use in any device since the main purpose of this project is to enable students to help each other.

Even though SharePoint is not a free product the portal must be developed at no cost. Accomplishing that objective with classical SharePoint (SharePoint2013 or previous versions) would have been impossible since a big server would been needed. That’s why for this project the SharePoint used will be SharePoint Online which is part of Office365. Without server we can’t use any C# code. That’s why it also has use the out of the box web parts that SharePoint offers. To accomplish that requirements, the portal will need to be responsive while still using all the out of the box web parts.

The portal has to be scalable and usable in a future. That means that if the number of users increases exponentially the portal has to stay fast and easy accessible.

The look and feel of the portal has to be completely customized since the design will be an important part of the project. Because of that all images will be designed exclusively for this project.
3. Introduction

3.1. Project objectives

The main objective of this project is to develop a SharePoint portal that can be used for students to share their solutions, ask questions…

The specific objectives are:

- Design a portal that’s easy to use, intuitive and user friendly.
- Use agile methodology during the whole process.
- The portal has to be responsive. It has to be designed with all the different screens sizes and devices in mind.
- It has to be free to develop (the licenses of all the software, servers…).
- The portal has to be scalable (if the number of users increases the portal has to still work properly).
- Finish the project within the established time period.

3.2. Project Scope

Developing an Educational portal from the ground could be a project that would need a lot of investment and professionals involved (Designers, Architects, Developers, a Project Manager…). Since this project has been done by only one student it’s focused on only one subject and one group of students. It also uses some of the “out of the box” capabilities that SharePoint offers. This project has been developed with a possible real implementation in mind so with some extra work hours more features could be added.
4. Project Development Plan

The project is complex and with only one developer and a fixed timeline there was not much spare time. That’s why during the development of this project the Agile methodology was used (despite not being the minimum three people required). This was the chosen methodology because with it you know exactly what’s the state of the project all the time and you have a working product at the end of each iteration.

Agile methodology is a project management methodology that’s usually used in software development. It’s useful to respond to unpredicted problems. It accomplishes this by doing all the work in an incremental way. That means that you try to finish an early product and you keep improving it. This way if you have an unpredicted problem you will still have a working product.

In Agile methodology iterative work cadences are used in order to organize the development of the project. Those cadences are named springs. Each spring is divided into different short tasks. This way is easy to know what is already done. If there is not enough time to develop some of the features the product will still work. For this project three different springs or iterations were planned:
4.1. Iteration 1  (Starting date: 22/04/2016)

//Beginning, definition of the project and first working Product

- Introduction
  - What is this project about?
  - Why did I choose this project?
  - Objectives
- Investigation
  - Who uses SharePoint?
  - Why SharePoint? What does SharePoint offer?
  - Alternatives to SharePoint, pros and cons
  - Different SharePoint payment possibilities
  - My choice for this project (SharePoint Online) Philosophy to program on the cloud, Azure...
  - Explore what is DreamSpark, different possibilities (Azure vs Office365) ...
- Organization
  - Definition of Agile
  - Different Software that allows us to program our different interactions
  - TFS, why did I chose it, how it works...
- First Design
  - Drawings in paper in order to identify the different parts that I will need to develop for this project.
  - First aspx pages and stylesheets
- Start Developing
  - Set up the developing environment
    - Install Visual Studio
    - Create an account on.microsoft (office 365)
    - Install PowerShell Management for SharePoint and SharePoint Designer
    - Advantages about SharePoint online
  - Choose a domain (free one for now)
  - First really early stage prototype using POP 2.0 (Prototyping on Paper)
  - Design general Architecture (Site, Subsite, lists...) of the site
  - Develop the Master Page
  - Develop Home Page Layout
  - Develop General Page Layout
4.2. Iteration 2 (Starting date: 10/05/2016)

//Main characteristics finished

- Fix minor bugs from the previous iteration
- Building the Pages architecture
  - Definition of which architecture is going to be used in each page.
  - Investigate how can we deploy in SharePoint online.
  - Prepare a deploy infrastructure ready to go
  - Investigate how to build the structure using PowerShell
    - Differences between PowerShell for SharePoint 2013 and SharePoint Online
  - Build a PowerShell script that builds all the necessary structure
- Developing the main page
  - All links and the main functionality
  - Change the ribbon
- Developing Theory
  - Documents Web Part
  - Newsfeed Web Part (This Web Part Should be reusable)
  - Investigate connection (mapping) between SharePoint and other platforms like DropBox and Google Drive.
- Developing Exercises
  - Documents Web Part
  - Newsfeed Web Part (This Web Part Should be reusable)
- Developing Exams
  - Documents Web Part
  - Newsfeed Web Part (This Web Part Should be reusable)
  - A user with no developing experience should be able to add a new exam.
- Developing Test page (This page will contain different questions that have only 4 possible answers)
- Building the exact look and feel
  - Look for examples online
  - Develop the necessary CSS and JS to provide the right user experience (Must be responsive and adjust well in a mobile device)
- Investigate to transform comments (Python code) to HTML with colors
  - Select the best service available
  - How to use the API
  - Choose between the default page or comments
    - Can be done with JavaScript or I will need a Web App hosted at Azure to do this task?
4.3. Iteration 3  (Starting date: 22/06/2016)

//Finished Product

- Fix minor bugs from the previous iteration
- Replicate and Save SharePoint Online Portal
  - Investigate different alternatives to Replicate or Save a SharePoint Online Portal
  - Choose one and use one
- Write down all the processes done during development
- How could this become a portal for different subjects
  - Explain SharePoint out of the box Permissions Projects
- Nice to have
  - Possible improvements that could be done with more development time.
  - Extra Features
    - Reporting services
  - Fix all the bugs
- Simulate a first client contact with the final product. (In this case we will use volunteers, family, friends…)
  - Using the client insides make all the necessary little arrangements
  - Gather information about the major arrangements that could be done
- Finalization
  - Conclusion
  - Write down some aspects that it might have been missed
  - Finish the Project document, print it...
  - Prepare the Project Presentation

In order to keep track of all the tasks and iterations of this project the Visual Studio Team Services were used. As can be seen from the next figure, this tool allows the different users to plan their tasks, keep track of them, inform when a bug or a stopper appears and when this one is solved or closed. It's a really easy to use tool, that's the main reason why this tool was the chosen one.
Development of an Educational SharePoint portal for coding students

Figure 1 Visual Studio Team Services Board
4.4. Final followed process

- First contact with the tutor about the project (21/11/2015)
- Decide the title of the project
- Define the objectives of this project
- Investigate about SharePoint and related technologies
- Elaborate the project development plan
- Create a free Microsoft Developer account
- Download and install all the necessary software
- Set up the Office Online environment
- Enable the publishing features of SharePoint online sites and subsites
- Sketch the portal look and feel
- Investigate about how to implement Bootstrap to SharePoint online
- Install the solution that enables Bootstrap to run on SharePoint
- Create the SharePoint architecture (Sites, Subsites…)
- Develop the master page
- Develop a basic page layout
- Develop a solution in order to make some of the most used SharePoint web parts responsive
- Edit the current styles in order to have a common look and feel
- Create the custom icons of the home page
- Develop different page layouts for each page
- Test the portal with different devices
- Fix some minor bugs
- Fully functional product
- Write part of the project memory
- Partial presentation of the project (13/04/2016)
- Investigate how to create a test page inside SharePoint
- Create the Test page
- Develop the code highlighter for the standard SharePoint newsfeed.
- Adapt the code highlighter for mobile devices
- Create more users
- Final testing. Ask people to test the portal
- Solve some minor bugs regarding the security trimming of the portal
- Add some messages and documents showing the portal functionalities
- Write the user and administration manuals
- Finish writing the project
- Prepare the final presentation
- Final presentation (06/06/2016)
5. SharePoint

5.1. What does SharePoint offer? Why SharePoint?

SharePoint is a tool developed by Microsoft, it’s commonly used for intranets, document management and collaboration, public sites… The first edition was released 15 years ago and the popularity of this tool has been rising ever since. Right now SharePoint is being used by 78% of Fortune 500 companies. Some of the most important features that SharePoint offers are the security that provides to all the data stored, the users can easily create new pages...

Despite all of the exposed above this project could have been developed using a different technology. One of the main reasons why SharePoint was the chosen tool to build this portal was that I develop for SharePoint on a daily basis so I already have part of the know-how and I wanted to improve my knowledge about this amazing tool. Another important reason is that SharePoint online as well as SharePoint 2013 with the servers on the cloud (Azure) are both really easily scalable. That means that since your servers are on the cloud you can change the hardware configuration that is using the portal in order to adapt to a changing number of users.

5.2. SharePoint Look and Feel

SharePoint was designed with big screens in mind. The current SharePoint version works perfectly with a computer but it’s complicated to use with a mobile device. SharePoint 2016 is supposed to be more mobile friendly.

Currently SharePoint’s Look and Feel with mobile devices is not really good. Since the web portal developed in this project is aimed at computers as well as mobile devices the design will be a big part of the project.

The following figure shows different out of the box SharePoint pages displayed in mobile phones.
This pages contain some components that will be used in this project like the newsfeed for example. As can be seen in the figure, most of these pages and components are not mobile friendly. This means that these pages were not designed with a mobile device in mind. Because of that, a lot of development time during this project was invested into modifying these components. These modifications enable them to work properly with mobile devices.

5.3. Alternatives and different versions of SharePoint

This project has been developed using SharePoint Online. SharePoint Online is part of the Office365 program that Microsoft is trying to promote. Right now the stable versions are SharePoint 2013 SP1 and SharePoint Online. The first one is an on premises SharePoint with three different versions available: Foundation, Standard and Enterprise. Even though for SharePoint 2013 you need a server you can use Azure services to have a scalable server on the cloud. This is the recommended set up but as mentioned before this project will be developed with SharePoint online. One of the main reasons that made this choice easier is that you can have a free developer account with SharePoint Online for a year.
On the following lines SharePoint will be compared with some of the most popular alternatives:

- Dropbox and Google Drive: These two products offer a cloud storage service. They are really easy to use and a lot of people already use them for their personal storage. They are not fully customizable and there are not meant to build web portals.

- Slack: It's a team collaboration tool that's becoming really popular. It started as a tool for developers to share their code with people in the same project but right now is already being used in a large variety of business. Slack has been one of the most successful startups and right now it has a value over 4.5 billion dollars. Despite being already designed to share code does not offer the possibility to develop custom pages like SharePoint.

- Moodle: It is free and it has been widely used for education purposes. In a similar way that SharePoint it has different plugins, themes… On one hand some of Moodle themes are already mobile friendly, on the other hand it does not have Microsoft Office integration unlike SharePoint, or the security features that SharePoint provides.

### 5.4. Portal Architecture

SharePoint Online has a by default a site collection with an address like this: https://sharepointtfg.sharepoint.com/. A site collection is basically a collection of different sites. Each site can have different pages.

One important feature that will be used in this project is that, each site has a different newsfeed. The newsfeed is a SharePoint component that allows users to post different comments and enables them to give “likes” to comments. This feature can be implemented into a page using the Newsfeed web part. This provide us with a user interface that’s really good if you are interacting with the web portal using a big screen but is not responsive. This means that the page was not designed with a mobile device in mind.

The data stored in a site is organized inside lists and libraries. In a list you can store raw data, for example a list will be used to store all the newsfeed comments of this portal. A library on the other hand is used to store documents, images… Those documents can have metadata information that can be really useful for search purposes.
In order to have a separate newsfeed and documents web parts for each part of the portal (theory, exercises, exams and tests) four different sites had been created, each one with the Publishing feature enabled and the custom Master Page enabled.

Each site has one page but it will be possible to implement more in a future. The page layouts were developed so in a future the portal could be expanded with more subjects and without any extra coding needed.

The following figure shows the SharePoint Online infrastructure in a simplified way:

As can be seen in the figure the portal is completely in the cloud, that means that no servers were owned during this project.
5.5. SharePoint 2016

During the development of this project SharePoint 2016 was presented. Keep in mind that despite using SharePoint online, this is actually based on SharePoint 2013.

Since this project is being developed completely in the client side in a future this project could be easily converted to SharePoint 2016.

This new version of SharePoint offers a wide variety of functionalities, some of the most useful for this web portal could be:

- OneDrive Redirection: OneDrive is a Microsoft product that allows users to store their documents and information in the cloud easily and reach this information using any device. This functionality will allow to redirect a user to his Office 365 my site when he or she clicks on OneDrive. This might not sound like a big deal but will make the user experience much better.

- Hybrid Cloud Search: With this new feature, you will be able to have a unified search for all the information that’s shared with you. That means you will be able to search to your educative e-mail, SharePoint, Office365 sites, Documents…

- User Interface changes: With SharePoint 2016 Microsoft is trying to promote even more Office365. With this rework of the user interface, SharePoint and Office365 will have the same look and feel, so the user won’t even notice any difference.

- Performance Improvements: SharePoint is a powerful tool but is usually criticized because is not the fastest tool of the market. SharePoint 2016 wants to change this, because of that, Microsoft put a lot of effort into improving SharePoint performance.

- With SharePoint 2016 there will be zero downtime while patching. This was a major issue in SharePoint 2013 for some companies that needed their services to be online all the time.

- Increased the maximum File Size: Now, files can be bigger than 2 GB. This could allow to store long videos for teaching purposes for example.
- SharePoint 2016 will have a touch friendly interface. As can be seen in the figure below. This will mean that a user could navigate through the different libraries without having to zoom in and out.

![SharePoint friendly interface](image)

---

**Figure 4 SharePoint friendly interface**

SharePoint 2016 comes with many more features but these are the ones that could be more useful for this particular web portal.
5.6. SharePoint components that could be added by users

SharePoint enables the users (with permission) to create, personalize and edit sites and pages without the need of coding anything.

This web portal has been created with a purpose in mind but purposes usually change over time. That’s why a lot of thought went into designing the page layouts so the users could edit them in an easy way. The user should be able to edit pages, add some new components and these components should be responsive and automatically adapt to the web portal look and feel.

In order to add this components (web parts) the user has to open the page in edit mode. Only some users (teachers and administrators) can do this.

These are some of the components that users (teachers or portal administrators) could add in order to improve the portal:

- Embedded video: The users could add embedded videos from, SharePoint, Video library of Office365, YouTube or other video platforms. The figure below shows an embedded YouTube video.

![Figure 5 Embedded video](image-url)
- Calendar: As can be seen in the figure below a calendar can be introduced into a page. This calendar could show the future major events of the subject (exams, extra classes…). In order to add a calendar, the user simply has to add this new web app.

Figure 6 Calendar
Images, links, text... Editing the content editors already existing in the site the teachers will be able to add text, documents, images... The following figure shows all the different content that can be easily added using the content editors of this web portal.

Figure 7 SharePoint Online in "Edit Mode"

The Administrator Manual of this project shows how a user can easily obtain this results using this portal. SharePoint offers many more components; this are the simpler ones that could be more useful in a teaching environment.
6. Design

The portal had to be responsive, because of that, designing was a big part of this project. The designing process started with some hand drawings. The first drawings were drawn with a mobile phone screen in mind. While drawing the different screens or views of the portal those were photographed and updated to the P.O.P. (Prototyping On Paper) platform. This tool allows you to connect one image to another and you can have the first approximation of how the look and feel of the website will be. It can be seen in the image below the approximate look and feel that this first prototype had.

![First Prototype on Paper](image-url)

---

Figure 8 First Prototype on Paper
With this prototype some possible improvements were detected. Then the real design process started. The look and feel of the site was supposed to be mobile friendly. In order to accomplish that a new masterpage was installed. This new masterpage basically allowed Bootstrap to run with SharePoint with no issues.

Basically bootstrap divides the page in 12 columns and you can indicate the width of your components using these columns. It’s really helpful in order to make the webpage responsive.

The icons and the styles applied to the web parts were designed to share the same look and feel. The figure below shows the actual look and feel of the portal, where some changes were made.

All the buttons and web parts are perfectly adaptable to a little screen.

Figure 9 Web portal design on a mobile phone
In a computer screen the portal looks like in the figure below:

Figure 10 Web portal design on a laptop
7. Page Layouts

Two different page layouts designs were developed in order to provide more flexibility to the users. Each page layout is responsive and contains different web part zones. Using this web part zones, the user can add or delete web parts without needing any code.

These page layouts are written in ASP since this is the language used in SharePoint.

7.1. Home Page Layout

As it can be seen in the image below the home page layout has a header, the “Learnit” Home button, the four different buttons, and four web part zones. In two of them there are the Newsfeed (General discussion) and the Documents (General documents).

7.2. General Page Layout

The general page layout is the one being used at all the pages except from the homepage. The only difference is that the general page layout doesn’t have the four buttons.
8. Pages

This pages use one of the two page layouts explained previously. For more information on the page layouts please consult the annexes.

The figures below show the pages seen by a user with no administration permissions (a student for example). This user won’t be able to edit any of the content shown in this page. An administrator will see the same pages plus a configuration bar at the top of the page.

8.1. Home

The Home page is the first page the user will see once he or she logs in the web portal. The main purpose of this page is to show general information about the subject.

As can be seen in the figure below, the header of the page (1) shows the title of the web portal, that could be easily changed. This part is common in all the pages and once the user clicks on it, this component redirects him to the home page.

The next row contains four buttons that link to different pages. This buttons were designed to be minimalistic and still represent the page they redirect the user to. The buttons (2,3,4,5) redirect the user to different pages that will be explained with more detail later on.

The last row has a Newsfeed (6), where the users can share comments, recommendations, possible improvements regarding the subject in general.

Finally, we have a content editor (7). All the administrators can edit this area adding more text, documents… They can even add embedded videos, links to other pages…

This is a description of all the content that’s currently in the page. In the section “SharePoint components that could be added by users” there are descriptions of some of the components that should be simple to add to this portal. For example, in the home page, a calendar could be an interesting component that could really help the students.
8.2. Theory

The Theory page uses the general page layout. It has the header that redirects the users to the home page and two main columns. As can be seen in the figure below, the left column has a content editor (1). Using the content editor, the teachers or administrators can add and edit, text, documents… This area was designed in order to make it easy for teachers to share the official documentation of the subject, comments, recommendations…

Below the content editor, a document web app can be find. This web app allows users (students), to share documents, create folders… This will be the designed area to share summaries, useful documentation… The administrators can delete some of the documentation if it’s considered to be incorrect, offensive…

Finally, we have the newsfeed (3). Here, students and teachers can comment the theory of the subject, ask for advice, recommendations…
8.3. Exercises

This page also uses the general page layout. Because of this, the general structure is really similar to the theory page. In this case the main purpose of this page is providing the students with a tool to help each other solve the subject exercises.

The content editor (1) can only be edited by the teacher or portal administrator. This part has been designed to contain all the official exercises of the subject. The actual website of the subject with all the information could be embedded here if desired. Currently this part has an example (text and images) that show how the students can share their code using the tag [code]. This page has also a document web app (2) that allows students to share their documents (mostly .py in this case).

Finally, the most interesting part of this page is the newsfeed. This newsfeed has been modified (all the code can be found at the annexes) in order to convert all the text after the “[code]” tag to an HTML code. This is done using JavaScript and allows the students to easily recognize the elements of the code. The figure below contains some examples.
8.4. Exams

This page has the same components as the exercises page. Despite that, the main purpose of this page is to enable the students to practice with previous exams and ask questions if they don’t understand something.

This page has a contend editor (1), where all the official exams could be posted. A documents web app (2), where students could upload some exams with solutions, modified… Finally, the custom newsfeed that will allow them help each other in an easy way.
8.5. Tests

The test page allows students to test their knowledge about the subject. SharePoint allows the users to publish surveys but not tests. Because of that this page uses an external test embedded. The webpage providing the test is called “Poll maker”. This page also contains CSS code that enables to “clean” the user interface. Using this code, the page maintains the same look and feel of the hole web portal.

Poll maker allows users to create tests, display them as embedded content, and even see the results. In this demo the test doesn’t record which user is taking it so it couldn’t be used as an exam.

Despite that, poll maker allows the user to set up tests that are only available for a short period of time, where users need to have a personal code… With this type of tests this web portal could be also used to do exams.

The final result obtained using this page can be seen in the figure below. The Script Editor (1) contains the embedded code (annexes).
LearnIt

Test 1

Question 1
Answer A  Answer B  Answer C

Question 2
Answer A  Answer B  Answer C

Figure 17 Test Page
9. Newsfeed Syntax Highlighting

Some pages of the web portal (Exercises and Exams) display the comments on the newsfeed that have the tag “[code]” highlighted. This functionality allows the students to see the code better and to identify each element in an easier way.

This is one of the main features that makes this portal different from all the other SharePoint portals that already exist. Because of that, developing this feature was an important part of the project.

There are already some solutions that highlight the syntax of the code. The difficult part was to find one based on JavaScript or something that could run in the client-side of the application. The chosen one was code-prettify from google. Adding a script like the following one to any site will convert any HTML “pre” with the class “prettyprint” to a highlighted HTML.

```html
<script src="https://cdn.rawgit.com/google/code-prettify/master/loader/run_prettify.js?autoload=true&amp;skin=sunburst&amp;amp;lang=py" defer="defer"></script>
```

The important parameters of this script are:

- **src**: The url to where the JavaScript is stored.
- **autoloading**: Setting this parameter to true enables the script to run when the page is loaded.
- **skin**: This JavaScript library has different configurations of colors. The sunburst has a black background and clear colors. This will allow the students to see the code easily.
- **lang**: The “lang” parameter specifies the language that the code is written. In this case py refers to Python.

It was mentioned before that this JavaScript looks for the tag “<pre class="prettyprint">”. This is a long tag and it’s not an easy one to remember. In order to fix that issue we could have downloaded the JavaScript code and edit it changing the tag. This could be a nice function but it will be more difficult to update the script, to change the tag…

Instead another JavaScript was developed. This script searches all the newsfeed looking for the tag “[code]” changes the tag to <pre class="prettyprint">.
The next step was to ensure that this new highlighted code would stay inside the newsfeed and behave as a responsive component. In order to achieve that some extra CSS was added to the page. This CSS can be seen in the annexes.

Finally, an example of the final script working can be seen at the figure below:

---

Figure 18 Example of how the script works
10. Administration manual

This section pretends to help the administrator of this portal to perform some simple tasks. For more information, please visit the official SharePoint online forums or the Office365 support center.

All the following figures and information are from the development environment (sharepointtfg) but would work with any other domain.

The first thing an administrator needs are his credentials. In this case the user name will be something like: administrator@sharepointtfg.onmicrosoft.com

The admin center can be accessed through this route:
https://portal.office.com/AdminPortal/Home

Here you will need to introduce your credentials. Once you are verified as an administrator a page like the one shown in the figure below will appear.

![Office 365 Administration center](image_url)
If your setup is incomplete office will guide you through it (1). From this page you can easily manage the current users (2), manage the billing (3) ...

At the left part of the screen there are some navigation buttons. The most important ones are: Users, Billing, Settings and Report.

Form Users you can manage the current users, add new ones, add new administrators... The billing page will allow you to see your subscriptions and licenses. Using this page, you will also be able to purchase new services. Finally, the reports page will show you some data about the portal in simple graphs.

All of the above were settings and tasks that could be completed at the Office365 administration center. This center is really useful but in order to edit our pages and do some other tasks we will need to go to the SharePoint portal.

Once inside our SharePoint portal you can click the gear shown in the figure below:

![SharePoint Online page options](image)

Figure 20 SharePoint Online page options

The most important tabs of this dropdown are: “Edit page” (1), “Site contents” (2) and “Site settings” (3).
10.1. Edit Page

If you click “Edit page” the portal will show you the actual page in edit mode. Using this mode, you will be able to edit any web part of the actual page. You could even add new web parts, delete existing ones…

In order to add a new web part just click the “Add a Web Part” button at the desired zone. Then at the top of the page the “Insert” tab will become available. Once clicked this tab will show you all the web parts available. Take into account that not all the web parts have been customized to be mobile friendly.

Some pages like “Theory” have a “Content Web Editor” that will allow you to easily edit text, add videos, images, documents…

Once you are in edit mode you will see that some parts disappear and some new buttons appear. Don’t worry this is not the final look of the page. In order to see the result of your changes check in the page.

One important point to take into account is the state of the page. Once you start editing a page a yellow header will appear informing of the state of the page. This can be seen in the figure below.

- The first state is “Checked out”. This means that this page is being edited by you and that nobody else would be able to edited while the page stays in this state.
- Once you click “Check in” you can see a preview of the modified page but all the other users won’t be able to see any changes.
- Once you click “Publish this draft” the modified page will be visible by all the users.
10.2. Site Contents

An example of the site contents page can be seen in the figure below. This page shows us all the libraries and list that uses SharePoint to store the information. For example, the “MicroFeed” list will show us all the conversations that occurred in this page with all the details.

The library “Pages” contains all the webpages of the site. This library can be used to add different pages, edit the code of the existing ones... Activating features or adding some web parts to your pages could add new lists or libraries.

For example, the “Calendar” list will appear when a calendar is created. This list will enable us to create and manage calendar appointments.

There are many libraries and list but this are the ones that could be more useful for a SharePoint administrator. For more information, please visit the SharePoint online support website.

![Site Contents](image_url)

Figure 22 Site Contents
10.3. Site Settings

Site Settings is a page that allows the SharePoint administrator to edit a wide variety of parameters. This Administration manual will only explain some of its parts. For more information, please visit the SharePoint online support site.

- From the “Look and Feel” the administrator can change the master page, the page layouts, the title, description, logo… This tab will enable him or her to slightly change the look and feel of the web portal.

- The “Users and Permissions” tab enables the administrator to edit the people and groups permissions. The administrator will be able to accept new users into the web portal, give them exclusive permissions… For example, the teachers can see the suit bar but the students can’t. This type of permissions can be changed using this tab.

- From the “Site Administration” tab, the administrator could change the regional settings of the web portal as well as the language settings, user alerts…

- The “Site Actions” tab allows the administrator to manage the site features. This features can enable or disable some functionalities of the web portal.

- The “Site Collection Administration” tab has a lot of links. Some of the most important ones are:
  - Recycle bin: All the documents deleted at the web portal go to the recycle bin. This way, if a document is deleted by mistake can be always recuperated.
  - Site collection features: Some features affect all the site collection and not only the current site. This features can be accessed by clicking this tab.
  - Popularity and Search Reports can help the SharePoint administrator identify the documents that have been more viewed by the students.

The following figure shows the current site collection settings.
Development of an Educational SharePoint portal for coding students

Figure 23 Site Settings

This section pretends to be a little user manual to show how this web portal works. In order to enter the web portal, you will need your credentials. These should be provided by your teacher or SharePoint administrator.

Once you log in into the portal you will see the home page. All the pages have a header with the “LearnIt” logo. This logo redirects you to the home page.

The following figure shows the actual home page.

![Actual home page](image)

The four buttons enable you to navigate through the web portal. The web portal has 5 pages. Home Page, Theory, Exercises, Exams and Tests.

After the buttons you will see a Newsfeed that will be used to discuss general things about the subject. The right part of the screen will show you the content that the teacher or SharePoint administrator puts there. In this case a description and an embedded YouTube video about “Why python is the future”. After this content there is a section where students can upload their documents regarding the subject in general.

The Theory page contains information and all the official theory documentation of the subject. Apart from that, students can upload their own documents, summaries… A newsfeed is also provided. This newsfeed could be used by the students to help each other.
The Exercises and Exams pages have the same structure. The main difference is that one page is used for exercises and the other for exams. This pages have the official information, exercises and exams. A web part that allows the students to share their own solutions and a customized newsfeed.

As can be seen in the figure below, the newsfeed highlights the code syntax. This is really useful to read the code in an easy way.

In order to do so the code has to be introduced with the tag “[code]” before it. The figure below shows an example. Then, after the page is reloaded the newsfeed will highlight your code syntax.
Finally, the test page will show different tests that you can complete in order to practice your coding abilities. If a test is mandatory the teachers or SharePoint administrator will inform you with previous notice.

12. Software used

In order to develop this project a lot of specific software was used as it can be seen in the picture below. Some of this programs have been obtained thanks to the UPC-Dreamspark program.

- GitHub is really useful to share code between devices, share code with other developers, look for examples…

- Project 2016: Project allows you to plan your project, count the hours dedicated to it, keep track of the progress done, have an idea how much could a project like this cost…

- SharePoint Designer 2013: It allows you to change the masterpage, pages, page layouts without deploying a new solution.

- SharePoint Online Manager and Windows PowerShell ISE: It allows you to automatizate some repetitive tasks, like deploying, creating the necessary structure in order to deploy our solutions…
- Visio2016: Visio allows you to easily create a wide variety of pictures and diagrams. This program was basically used during the designing phase of the portal.

- VisualStudio 2013: Puts all the code together. It also compiles all the documents needed into a solution ".wsp" file in order to deploy them later.

13. Budget

The budget of this project is mainly developing hours since one of the objectives was to build this platform without any extra cost (like buying software, servers…). Apart from the developing hours we could account other costs like all the disposables used, the amortization of the pc… The next table shows all the expenses disaggregated:

<table>
<thead>
<tr>
<th>Tasks</th>
<th>Professional</th>
<th>Duration (h)</th>
<th>Cost (€/h)</th>
<th>Cost (€)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Define the portal architecture</td>
<td>Business Analyst</td>
<td>25</td>
<td>35</td>
<td>875</td>
</tr>
<tr>
<td>Design the portal HTML and CSS</td>
<td>Designer</td>
<td>120</td>
<td>25</td>
<td>3.000</td>
</tr>
<tr>
<td>Define the portal architecture, chose SharePoint version, define the software that should be used</td>
<td>Software Architect</td>
<td>15</td>
<td>45</td>
<td>675</td>
</tr>
</tbody>
</table>
## Development of an Educational SharePoint portal for coding students

<table>
<thead>
<tr>
<th>Schedule all the tasks, develop documentation, test the portal and present it.</th>
<th>Project Manager</th>
<th>40</th>
<th>50</th>
<th>2,000</th>
</tr>
</thead>
<tbody>
<tr>
<td>SharePoint configuration, JS paint code, integration…</td>
<td>SharePoint Developer</td>
<td>120</td>
<td>35</td>
<td>4,200</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td></td>
<td><strong>320</strong></td>
<td></td>
<td><strong>10,750</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Other costs</th>
<th>Total cost</th>
<th>Useful life (h)</th>
<th>Used time (h)</th>
<th>Cost (€)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Paper, printing, presentation costs...</td>
<td>18</td>
<td></td>
<td></td>
<td>18</td>
</tr>
<tr>
<td>PC amortization</td>
<td>1,000</td>
<td>5*365= 1,825</td>
<td>320</td>
<td>172,602,797</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td></td>
<td></td>
<td></td>
<td><strong>10,940,60</strong></td>
</tr>
</tbody>
</table>

As can be seen in the table above the total cost of this portal is **10,940,60€**

The price per hour of this budget was calculated using online platforms like “glassdoor” as well as input from people who work at those positions.

The amount of hours invested in each task is the real amount of hours that as a young developer I have invested. A more experienced professional might have done the same work faster.
14. Possible future improvements to the portal and/or expansions

This project has been centered in developing a portal to help coding students share their ideas and solutions. This portal could be adapted to different subjects. Maybe they won’t need the “[coding]” feature but they might need some other functionalities that are not actually developed.

This portal is making use of the Social Part of SharePoint, a reporting services could be created in order to know which one of the users is being more active, which one is more useful to the other students (more likes), which parts of the subject need to be explained better…

Apart from the previous considerations this portal is being developed with only one group or class in main. A future development could be extending the functionalities of this portal to different classes, with some administrators (Teachers) that could be able to see some of them.

The major part of the design of this portal has been done with static CSS. It has been done this way in order to simplify it. An important improvement could be use more JQuery instead of some part of the CSS. This will enable the users to have an even better experience adapting better the different columns to different screen sizes.

Another possible improvement could be use Hammer.js to improve the user experience. Hammer makes the web page run faster, the delay from some actions is drastically reduced. It also adds some new functionality for touch screens like rotate, pinch, press, swipe…

This platform could be integrated with already existing platforms in our university such “S.O.S. ETSEIB” that currently uses Dropbox to share documentation between students.

Finally, this is a web portal but with the upcoming SharePoint 2016, iOS and Android apps could be developed. In order to do so, some JavaScript frameworks like Ionic or Cordova could be used.
15. Environmental Impact

Since this is a software project all the environmental impact can be divided between the electricity and the paper consumed during its development.

The CO₂ emissions from the energy consumed are shown at the following table:

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer (Laptop)</td>
<td>320h</td>
<td>120</td>
<td>38.40</td>
<td>24.96</td>
</tr>
<tr>
<td>Illumination</td>
<td>220</td>
<td>89</td>
<td>19.58</td>
<td>12.727</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td></td>
<td></td>
<td><strong>57.98</strong></td>
<td><strong>37.69</strong></td>
</tr>
</tbody>
</table>

*The laptop used was an Asus N56, and the illumination was provided by a fluorescent light of 89W.

The European Union estimates that in order to produce 1 kWh, 0.65 kg of CO₂ are thrown to the atmosphere.

This project has been printed, and approximately 135 A4 papers were used during it’s development, that means 8.42m² of paper. If we consider that the density of the paper used is 90g/m², then the total amount of paper used is 757.35 g.
16. Conclusions

After completing the project and thanks to following the Agile methodology the portal is a working product with a really good responsive look and feel. All of the project objectives have been accomplished.

During the development process of this web portal a lot of effort went into designing the different pages and modifying the out of the box styles of SharePoint in order to have a responsive product. The result is a custom SharePoint portal that runs perfectly in mobile devices. Despite that, some of this work won’t be needed with the new SharePoint Online version.

The final portal has been tested with more than 10 different devices. During the testing process some bugs appeared. They were all solved, but in order to solve some of them the initial structure of the portal was modified. This could have been avoided if I have had more experience with the SharePoint architecture and the profile configuration of Office365.

During this project I have discovered that the initial estimated time of some tasks was too low. I ended up spending a lot of time fixing bugs, setting up the Office365 environment, investigating... This was frustrating but also taught me to analyze better all the tasks before estimating them.

I’ve been able to finish the project within the established time period but there are still a lot of possible improvements that could be done. The biggest one could be translating this web portal into an Android or iOS app.

Currently the portal has only three users, one administrator and two students but is easily scalable since is part of Office365. I will love to see this portal or a similar one being used in the future!
17. Acknowledgments

I would like to express my gratitude to my tutor Lluis Solano. All the meetings were really useful and without his input this project wouldn’t be the same.

I would also like to thank all the team at Sogeti Barcelona. Starting with Jaime Arroyo and Alvar Rius. I’ve been working with them this whole course and I’ve learned a lot from them. Without them and all that they taught me this project wouldn’t even exist. I also have to thank Edin Kapik for all his support. Without all the information that he provided me with I wouldn’t be able to even set up a SharePoint environment. It also helped a lot to attend the SharePoint Saturday hosted in Barcelona which he organized.

Finally, I would like to thank my family and friends who supported me during the development of this project.
Bibliography

   [https://cordova.apache.org/, 23/05/2016]

[2] CODEPLEX. PROJECT HOSTING FOR OPEN SOURCE SOFTWARE. Responsive
    SharePoint August 19, 2014.
   [https://responsivesharepoint.codeplex.com/, 03/04/2016]

   [https://www.glassdoor.com/index.htm/, 15/05/2016]

   [http://hammerjs.github.io/, 10/05/2016]

   [http://ionicframework.com/, 16/05/2016]

    20/03/2016]

   [http://www.poll-maker.com/, 02/03/2016]
18. Annexes

This annexes contain the ASPX, CSS and JS code of the two page layouts developed for this project.

18.1. General Page Layout

This page layout is used by all the pages except the home page.

18.1.1. ASPX

The ASPX code is the skeleton of the page layout. This code is structured using Bootstrap. In this case we have two rows, one for the header and another one for all the web part zones. Despite only using one or two web part zones in the second row, there are up to four of them in case some new functionalities were added.

```csharp
<%@ Register Tagprefix="SharePointWebControls" Namespace="Microsoft.SharePoint.WebControls" Assembly="Microsoft.SharePoint, Version=15.0.0.0, Culture=neutral, PublicKeyToken=71ebce11e9429c" %>
<%@ Register Tagprefix="WebPartPages" Namespace="Microsoft.SharePoint.WebPartPages" Assembly="Microsoft.SharePoint, Version=15.0.0.0, Culture=neutral, PublicKeyToken=71ebce11e9429c" %>

<asp:Content ContentPlaceHolderId="PlaceHolderPageTitle" runat="server">
    <SharePointWebControls:FieldValue id="PageTitle" FieldName="Title" runat="server"/>
</asp:Content>
<asp:Content PlaceholderID="PlaceHolderPageTitleInTitleArea" runat="server">
    <SharePointWebControls:FieldValue FieldName="Title" runat="server"/>
</asp:Content>
<asp:Content ContentPlaceHolderId="PlaceHolderMain" runat="server">
    <div class="row">
        <div class="col-md-12">
        
```
Development of an Educational SharePoint portal for coding students
Development of an Educational SharePoint portal for coding students

```html
<div class="webparts">

<WebPartPages:WebPartZone runat="server" AllowPersonalization="false"
ID="bootstrapRow5Column1Sub1" Title="Left Footer Left" FrameType="None"
Orientation="Vertical"><ZoneTemplate"></ZoneTemplate></WebPartPages:WebPartZone>

</div>

</div>

<div class="col-md-2">

</div>

<WebPartPages:WebPartZone runat="server" AllowPersonalization="false"
ID="bootstrapRow4Column1Sub2" Title="Left Footer Left" FrameType="None"
Orientation="Vertical"><ZoneTemplate></ZoneTemplate></WebPartPages:WebPartZone>

</div>

</div>

<div class="row">

<div class="col-md-3">

</div>

</div>

</div>

<div class="col-md-4">

</div>

</div>

<div class="col-md-4">

<WebPartPages:WebPartZone runat="server" AllowPersonalization="false"
ID="bootstrapRow5Column1Sub0" Title="Left Footer Left" FrameType="None"
Orientation="Vertical"><ZoneTemplate></ZoneTemplate></WebPartPages:WebPartZone>

</div>

</div>

</div>

<div class="col-md-4">

<WebPartPages:WebPartZone runat="server" AllowPersonalization="false"
ID="bootstrapRow6Column1Sub1" Title="Left Footer Left" FrameType="None"
Orientation="Vertical"><ZoneTemplate></ZoneTemplate></WebPartPages:WebPartZone>

</div>

</div>

</div>

<div class="col-md-4">

</div>

</div>

</div>

</div>

</div>

</div>

</div>
18.2. Home Page Layout

This page layout is used at the home page.

18.2.1. ASPX

In this case we have three rows. The first one is the header, the second one has the four buttons and another one for all the web part zones. Despite only using one or two web part zones in the second row, there are up to four of them in case some new functionalities were added.
Development of an Educational SharePoint portal for coding students

```xml

<asp:Content ContentPlaceHolderID="PlaceHolderPageTitle" runat="server">
    <SharePointWebControls:FieldValue id="PageTitle" FieldName="Title" runat="server"/>
</asp:Content>
<asp:Content ContentPlaceholderID="PlaceHolderPageTitleInTitleArea" runat="server">
    <SharePointWebControls:FieldValue FieldName="Title" runat="server"/>
</asp:Content>
<asp:Content ContentPlaceHolderId="PlaceHolderMain" runat="server">
    <div class="row">
        <div class="col-md-12 toplearnit">
            <!-- First Row -->
            <div class="row">
                <div class="col-md-12">
                    <div>
                        <WebPartPages:WebPartZone runat="server" AllowPersonalization="false" ID="bootstrapRow1" Title="loc:Header" FrameType="None" Orientation="Vertical"><ZoneTemplate/></WebPartPages:WebPartZone>
                    </div>
                </div>
            </div>
            <!-- Second Row -->
            <div class="row">
                <div class="col-md-12 toplearnittitle">
                    <div>
                        <a href="/Paginas/Home.aspx">
                            <h6>LearnIt</h6>
                        </a>
                    </div>
                </div>
            </div>
        </div>
    </div>
</div>
</div>
</div>
</asp:Content>

<figure class="snip1445">
    <img src="/Style Library/LogoTeoria.png" alt="sample84" />
    <figcaption>
        <div class="textteoria">
            <h4>Theory</h4>
        </div>
    </figcaption>
    <a href="/Theory2/Pages/Theory2.aspx"></a>
</figure>
</div>
</div>
</div>
</div>
</asp:Content>
```
Development of an Educational SharePoint portal for coding students
Development of an Educational SharePoint portal for coding students
18.3. CSS

These are the styles used in order to provide the page with a user friendly look and feel. There are some styles which have the property "!important". This is because SharePoint uses JavaScript to change these styles. While working with SharePoint, in some cases the "!important" tag is relevant in order to ensure that these styles are applied to all the page. These styles are shared by all the pages of this web portal.

```css
/*----------------------*/
/*Custom CSS Functions for LearnIt*/
/*----------------------*/

.snip1445:hover h4{
  color:white;
}

div.webparts{
  width: 330px !important;
  display: block;
  margin-left: auto;
  margin-right: auto;
  background-color: #F3E7D3;
}
.webparts{
  margin:15px;
  border: 5px solid black;
  padding-left:10px !important;
  padding-right:10px !important;
}
.col-md-4{
  padding-left:50px !important;
  padding-right:50px !important;
  min-width: 350px;
}

#QCB1_Button3{
  display:none !important;
}

/*Changes the boxes text to white when the snip is hovered*/
```
Development of an Educational SharePoint portal for coding students

.snip1445:hover h4,
.snip1445:hover h4 {
  color: white !important;
}

/*Disables the horizontal scrollbar*/
#s4-workspace {
  max-width: 100% !important;
  overflow-x: hidden !important;
}

/*Custom microbloging mobile friendly*/
#ms-feedthreadsdiv{
}
#ms-microblogInputFocus{
  max-width: 300px !important;
}
#ms-mbRightSpan{
  padding-right: 150px !important;
}
.ms-microfeed-replyArea{
  max-width: 1px !important;
  margin-left: 0px !important;
}
.ms-microfeed-postReplyButtonSpan{
  padding-right: 0px !important;
}
.ms-microfeed-userThumbnailArea{
  display: none !important;
}
.ms-microfeed-rootBody{
  margin: 0px 0px 0px 0px !important;
}
.ms-microfeed-replyBody{
  margin: 0px 0px 0px 0px !important;
}
.ms-microfeed-rootText{
  min-width: 30px !important;
  max-width: 300px !important;
}
.ms-microfeed-replyBackground{
  background-color: #FBDCAB;
  max-width: 300px !important;
  min-width: 20px !important;
}
#ms-microblogInputFocus{
  background-color: #F9F2E7;
}
div.ms-microfeed-fullMicrofeedDiv button[disabled][type="button"] {
  cursor: pointer;
  border-color: #F88600;
Development of an Educational SharePoint portal for coding students

background-color: #F09609;
background-color: #F09609;
color: #fff;
}
.ms-microfeed-focusBoxNoFocus {
    background-color: #F9F2E7;
}
.ms-microfeed-replyText {
    max-width: 290px !important;
    min-width: 20px !important;
}
.ms-microfeed-focusBoxNoFocus {
    max-width: 290px !important;
    min-width: 20px !important;
}
.ms-webpart-titleText > a {
    font-weight: 900;
}
.ms-microfeed-replyBoxIndentDiv {
    margin: 0px 0px 0px 0px !important;
}
.ms-microfeed-feedPart {
    /*Centers the cross*/
    min-width: 20px !important;
    max-width: 300px !important;
    position: relative;
}
.ms-microfeed-microblogpart {
    min-width: 250px !important;
    max-width: 290px !important;
}

/*Documents Responsive*/
.ms-vb2 {
    display: none !important;
}
.ms-vh2 {
    display: none !important;
}
.ms-vh-icon{
    display:none !important;
}
.ms-vh{
    display:none !important;
}
.ms-vb-user{
    display:none !important;
}
.ms-cellStyleNonEditable{
    display:none !important;
}

/*LearnIt Top*/
18.4. JS

This section contains the JavaScript used for this page layout. The first function is used in order to hide some parts of the page that were not desired. The second one is used to change the tags “[code]” to <pre class="prettyprint">. This code is shared between all the pages of this web portal.

```javascript
function DisplayNoneStyle() {
    var $target = $("ul#zz13 RootAspMenu>li:nth-child(1)>a");
    $target.hide();
}
```
```javascript
$(document).ready(function() {
    var cusid_ele = document.getElementsByClassName('ms-microfeed-postBody');
    for (var i = 0; i < cusid_ele.length; ++i) {
        var item = cusid_ele[i];
        item.innerHTML = item.innerHTML.replace('[code]', '<pre class="prettyprint">');
    }
});
```