



---

# UCL CS Student News and Twitter Portal

---

## **Team Number: 26**

**Project Title:**UCL CS Student News and Twitter Portal

**Yuxuan Li**

yuxuan.li.17@ucl.ac.uk

**Tiefu Cai**

jeff.cai.17@ucl.ac.uk

**Pierre-Alexandre Gruman**

pierre-alexandre.gruman.16@ucl.ac.uk

April 23, 2018

This report is submitted as part requirement for the undergraduate degree at UCL. It is substantially the result of my own work except where explicitly indicated in the text.

The report may be freely copied and distributed provided the source is explicitly acknowledged.

Department of Computer Science

University College London

# Abstract

Our team was assigned two separate tasks for this module. So for almost every category, we'll present the two tasks of our project.

For the first task, our client, Lucy Tallentire, the UCL Department of Computer Science Communications and Marketing Manager, asked us to create an online portal for students to upload media on, such as articles, images or both. That would allow our client, the admin of our website, to have more news to share in the Newsletter and on Twitter. We were asked to use a recognizable UCL Template in order to harmonize the portal with all the other UCL websites and other online tools. In addition, our client required that she could login on the portal and manage the posts and be the only one to be able to do that.

Technically for the web page design, we used Bootstrap templates. We used PHP as server-side language and MariaDB for our database that store the information provided by the students for each post. As for the pdf generation, we used the third-party app (OpenOffice). Regarding the tweet button, we referred to Twitter's API [1]. The portal has three main pages, a homepage that displays a list of all the uploaded posts, an upload page where students can put up posts and an admin page where the admin can manage the posts.

The final website we built is a tool that will provide our client with information coming directly from the students. Students can share their activities by uploading posts on this portal. The admin can login using her password in order to manage the upload list and choose what to publish and what not to. She can choose to delete an article from the list, download an article as a pdf, or tweet the article directly from the portal. So all the "must have", "should have" and "could have" requirements have been fulfilled.

For the second task, our client Dr. Dean Mohamedally, asked us to work on the Industry Exchange Network (IXN). It is a website ran by three professors from the UCL Department of Computer Science that allows students to engage in real work-based learning projects with companies, charities or the public sector. We had to create a new page for the website that lists all the jobs available so that the students can easily access them and filter them.

We took the IXN styles to create a coherent webpage that fits into the website seamlessly. For back-end, we used similar technologies as the first task.

We did not have to build a whole website from the ground up but we had to add a complimentary page of an existing website. We did manage to fulfil the requirements of this project, so that students can access all the jobs in one convenient place and apply a deadline filter in order to find the jobs that match their schedules. We hope that it will enable students to have a better overall experience with the IXN.

# TABLES OF CONTENT:

---

<b>Abstract .....</b>	<b>2</b>
<b>1 Introduction .....</b>	<b>6</b>
1.1 Background of our project and problem statement .....	6
1.1.1 Background of the client and project .....	6
1.1.2 Problem goal and definition .....	7
1.2 Development Team .....	7
<b>2 Requirements .....</b>	<b>8</b>
2.1 Persons .....	9
2.2 Requirement of Task 1 .....	9
2.3 Requirement of Task 2 .....	9
<b>3 Research .....</b>	<b>9</b>
3.1 Related Technologies .....	9
3.2 Summary of final decision .....	9
<b>4 Design and implementation</b>	
4.1 Design .....	10

4.1.1 For the first task .....	10
4.1.2 For the second task .....	14
4.2 Development tools .....	16
4.3 Implementation of key functionalities .....	16
4.3.1 Update new information in homepage .....	17
4.3.2 How to automatically generate a link to a file .....	17
4.3.3 How to convert a text file to pdf file .....	17
4.3.4 Pagination.....	18
<b>5 Testing .....</b>	<b>19</b>
5.1 Testing strategy .....	19
5.2 Compatibility testing .....	19
5.3 Responsive design testing .....	20
5.4 Users acceptance testing .....	20
<b>6 Conclusion and future work .....</b>	<b>21</b>
6.1 Summary of achievement .....	21
6.1.1 Function Achievement table .....	21
6.1.2 Individual distribution table .....	22
6.1.3 Known bug list .....	22

6.2 Function achievement .....	23
6.3 Critical evaluation .....	23
6.3 Future work .....	24
<b>7 Appendix .....</b>	<b>24</b>
7.1 User manual .....	25
7.2 Development manual .....	25
7.3 Code citation .....	26
<b>8 Reference .....</b>	<b>26</b>

# 1. Introduction:

---

## 1.1 Background of our project and problem statement

### 1.1.1 Background of the client and project

Our first client is Lucy Tallentire, the UCL Department of Computer Science Communications and Marketing Manager. Her goal with this project is to improve the social appeal of the CS department. She noticed that students were not as involved in sharing their activities as the staff was, so she asked us to create this portal in order to establish a link between her and the students.

Our second client is Dr. Dean Mohamedally Teaching Fellow, Director of Apps Engineering and one of the creators of Industry Exchange Network. IXN enables students to engage in real-world problem based learning through term-based client projects. A “job” function had to be added so that the students can find opportunities more easily.

### **1.1.2 Problem goal and definition**

We were assigned two tasks for this project.

First task:

Lucy Tallentire asked us to work on developing an online platform that would allow students to upload media on. We would then have to convert the media into a pdf format file.

Second task:

Dr. Dean Mohamedally asked us to work on the IXN platform, the Industry Exchange Network. We will have to develop a listing of all the industry placements available on the website.

## **1.2 Development Team**

Yuxuan LI(Project Leader):

With a high-level coding ability, Yuxuan Li was a natural fit for our project’s back-end design and back-end setting up (include server and database). As a leader, he always infiltrates into every task of the whole project with due diligence.

Team Roles:

- Front-end developer
- Back-end developer
- Report Editor
- Researcher

Pierre-Alexandre Gruman :

Subsequently, PA works as a copywriter, he contributed a lot in client liaison and video demonstration. Although he had little experience in writing PHP, he finally finished the problem he met and became familiar with the Twitter Login API. The twitter login function was finished by him.

Team Roles:

- Vedio maker
- Report Editor
- Tester

Tiefu Cai:

With a medium level of coding ability and design, he made some contribution, both into coding and some designing task, and he was good at extracting some useful information during research, the online PDF converter function was finished by him.

Team Role:

- Poster Designer
- UI designer
- Report Editor

Responsibilities were divided averagely according to each team member's ability and we can finish the work together, that might enable every member to gain a clear understanding of the whole project.

## 2.Requirements

---

### 2.1 Persons

The portal is exclusively for UCL students. But the administrator will be Lucy Tallentire, allowing her to have access to what is uploaded and filter the information.



The IXN platform allows students to get real job opportunities and accumulate work experience by learning through term-based client projects.

## 2.2 Requirement for task 1 requirement :

Requirements (Task 1)	
Must have:	Portal for students to upload media (desktop version ), pdf generation
Should have:	Tags for different subjects, an admin access for Lucy Tallentire, Conversion functionality of student posts into Tweets
Could have:	Login, Mobile version
Won't have:	Access to other students' uploaded media

## 2.3 Requirement for task 2:

Requirements (Task 2)	
Must have:	List of job offers, expire date, tags for job keywords
Should have:	Description of job offers
Could have:	Comments, Feedback (from both students and companies)
Won't have:	Private information (CV, motivation letter, etc)

# 3. Research

---

Both of two tasks are web development and share similar features, so our team decide to apply same web development technology on both tasks.

## 3.1 Related technologies

For the server-side language, potential solutions can be PHP, JavaEE, Node.js or Python.

After consideration, we choose PHP as our server-side language. Because unlike other languages that had to be adapted for use on the web, PHP can integrate well with html, css, js and Mysql when developing web apps. Also, PHP scripting is easy to learn. This is because the PHP is similar to C and Java to an extent. It is an open source and popular[2], there are many blogs, forums and PDFs that discuss the problems associated with PHP coding. Developers can take guidance from these sources.

For database, potential solutions can be Mysql, MariaDB, PostgreSQL, MongoDB... After consideration, we choose MariaDB as our server-side language. Because MariaDB 5.5 is a complete drop-in-replacement for MySQL and 5.5. MariaDB development is more open and vibrant. MariaDB has a much-improved query optimizer and many other performances related improvements. [3] Certain benchmarks show that MariaDB is radically faster than MySQL. We use xampp as a local server development tool, and it integrated MariaDB as default.

For server, potential solutions can be Microsoft Azure, AWS, Heroku, Apache web server or Nginx web server. After consideration, we choose Apache as local server for development step. Because it is a free local web hosting and "Apache servers are very compatible with other operating systems. It can be found in use with Windows, MacOS, Linux and Unix as well as most other popular options." [4]

At testing phase, we have to test our websites on different platforms and machines, so we also use Microsoft Azure as an online web hosting. Because it is fast and might be more secure than others. [5]

## **3.2 Summary of final decision**

Server-side language: PHP

Database: MariaDB

Front-End: HTML, CSS, JavaScript, jQuery and Bootstrap

Others: Openoffice(a third-party application to convert files to pdf format)

Server: Apache

# **4. Design and implementation**

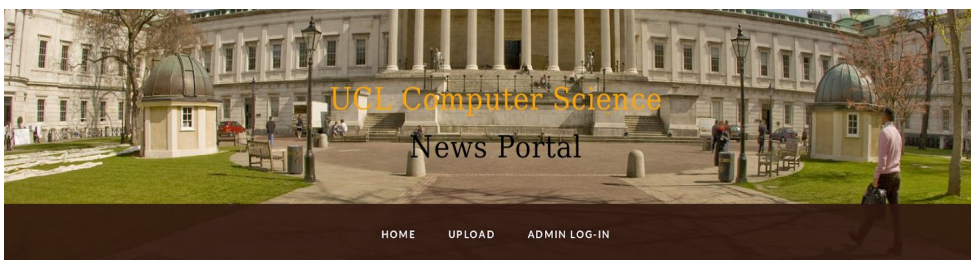
---

## 4.1 Design

### 4.1.1 For the first task (News portal)

We used templates to construct the main web page theme and made lots of modifications to fit our project.

On the home page, there is a navigation bar which contains “home”, “upload” and “admin login” icons. When clicking on “home”, the home page will refresh. Click on upload, the webpage will jump to upload.php page. And click on “admin login”, it will jump to admin login page, on which the admin can enter the username and password to manage the website. Below the navigation bar is the main content of the page, including all the news and posters. At each end of the news, there is a tab. When clicking on the tab, the user can view and download a pdf version of current news. To achieve these features, we use php, mariaDB and OpenOffice(third-party app). The php file reads data from the database and displays the information on the homepage.



**Title1**

Category: Computer  
Author: Author1  
Date: 2018-04-14  
Description: It is an online platform that would allow students to upload media on. We would then have to convert the media into a pdf format file.  
Email: email@email.com

[MORE](#)

**UCL CS NEWS PORTAL**

**Title1**

Category: Computer  
Author: Author1  
Date: 2018-04-14  
Description: It is an online platform that would allow students to upload media on. We would then have to convert the media into a pdf format file.  
Email: email@email.com

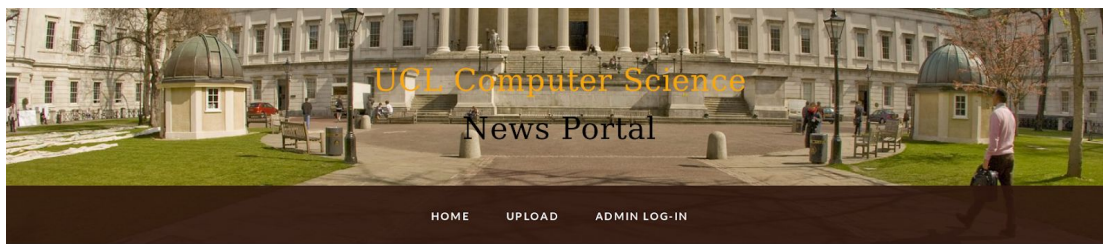
[MORE](#)

**Awdaawd**

Category: Computer

On the upload page, users can upload a local file to this website, which will be shown on the home page once finish uploading. Users have to fill the form before finish uploading, which has title, author, category, description and email information. We use php to handle file and form information. Saving form information as records

in database table and save files in a specific folder.



## NEWS UPLOAD

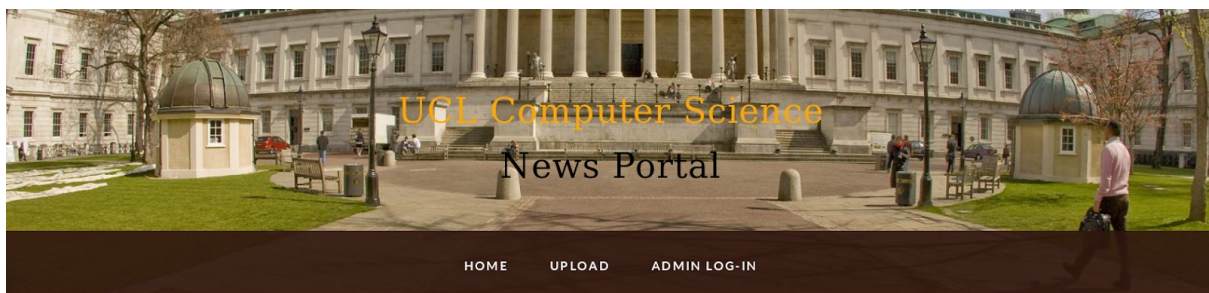
Title:  Category:

Author:  Description:

Email:

Add file:  No file selected.

On the admin login page, if admin entered correct username and password, they can then manage this website. Admin can delete any inappropriate news or posters. And the admin can push some relevant news into UCL CS official twitter account.



## ADMIN LOG-IN

USERNAME:

PASSWORD:

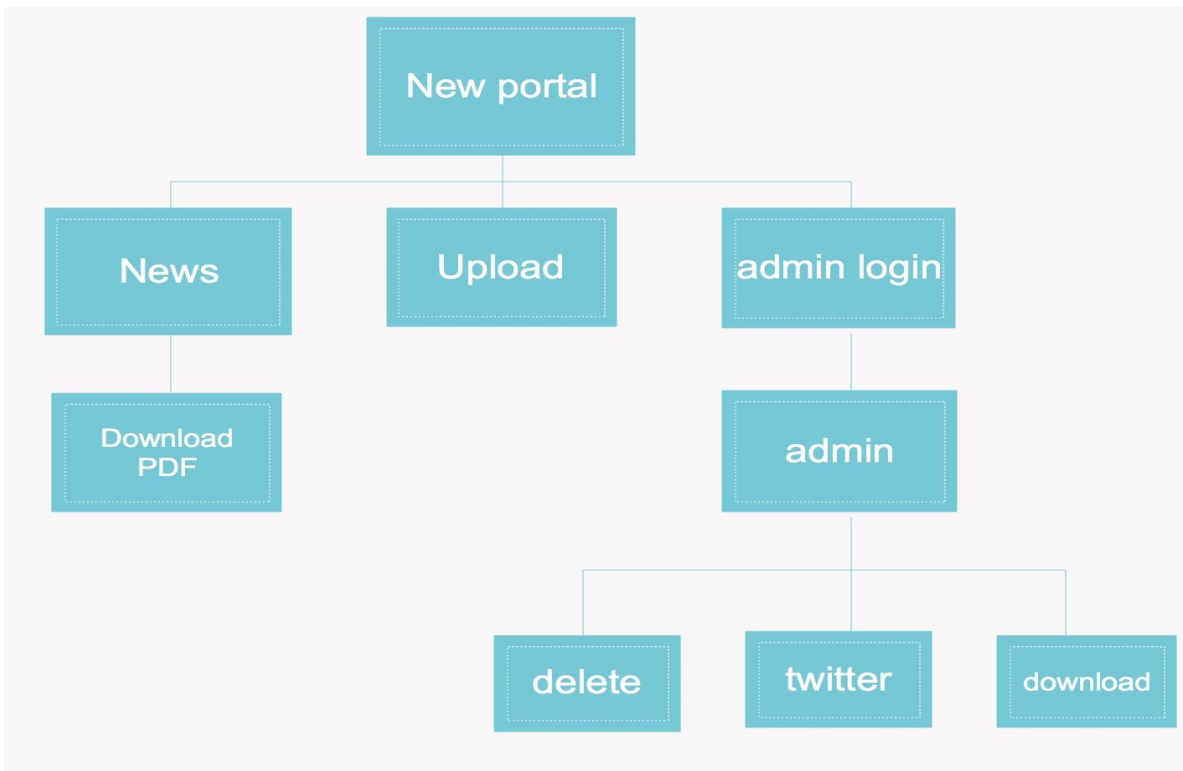


Show 10 entries Search:

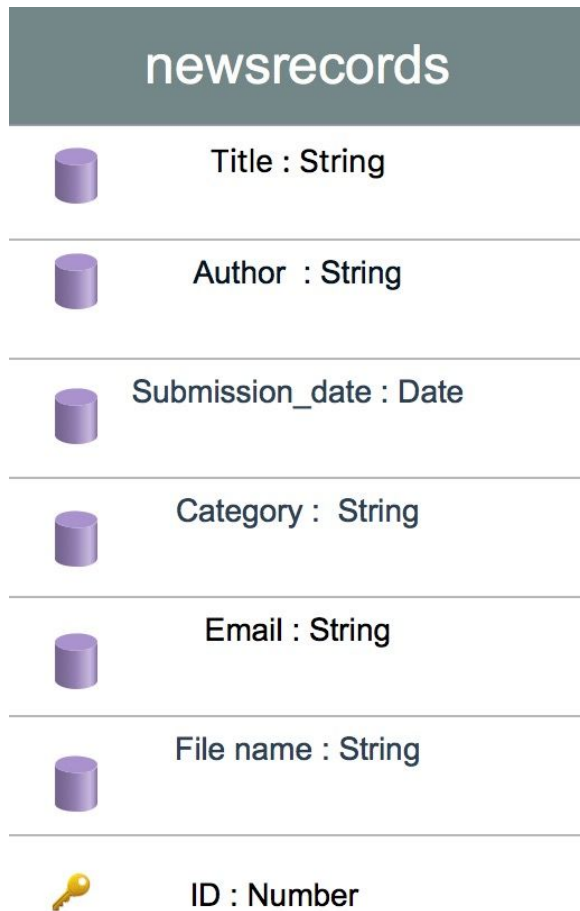
ID	Title	Author	Category	Submission date	Email	Original File	Tweet	Delete
70	asdawd	asd	Art	2018-04-09	12@as.com	asdawd_asd_2010 Answers.txt	Tweet	Delete
71	asdawq21	awd	Computer	2018-04-09	123123123@qq.com	asdawq21_awd_2015.pdf	Tweet	Delete
72	aswd	asdffsefsdf	Computer	2018-04-09	asda@awda.com	aswd_asdffsefsdf_Untitled.odt	Tweet	Delete
73	awdaawd	asda	Computer	2018-04-11	123123123@qq.com	awdaawd_asda_test.docx	Tweet	Delete
74	Title1	Author1	Computer	2018-04-14	email@email.com	Title1_Author1_test.docx	Tweet	Delete

Showing 1 to 5 of 5 entries Previous 1 Next

Site map for UCL CS News Portal:



ER diagram for News Portal:



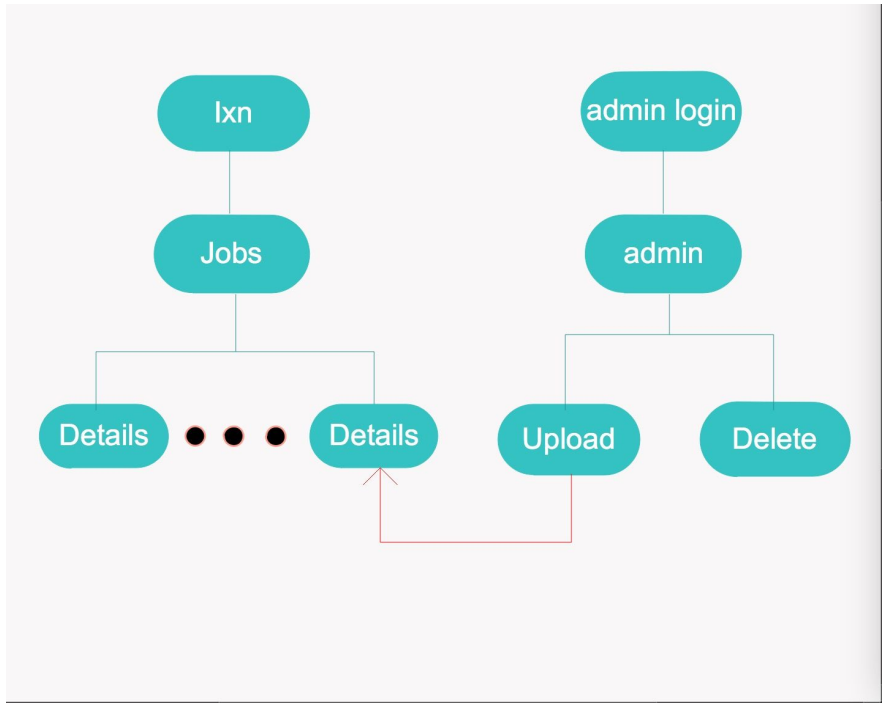
#### 4.1.2 For the second task (IXN)

On the home page, we add one “job” button in the navigation bar, and like any other sections on the IXN website, we add a section for “job” in the main content of the page, showing some information about job opportunities on IXN platform. And in this section, there is a “more” button, when you click on this button, it will jump to job page.











On the job page, there is a list of opportunities available on the IXN platform. We use php to read data from database to realise this feature.

As for upload, admin-login and admin pages, we applied similar design and technologies as the first task.

Site map for IXN job task:



ER diagram for News\_portal:

datalists	
	Job name : String
	company : String
	Submission_date : Date
	Deadline : Date
	Category : String
	Description : String
	Email : String
	File name : String
	Status : String
	ID : Number

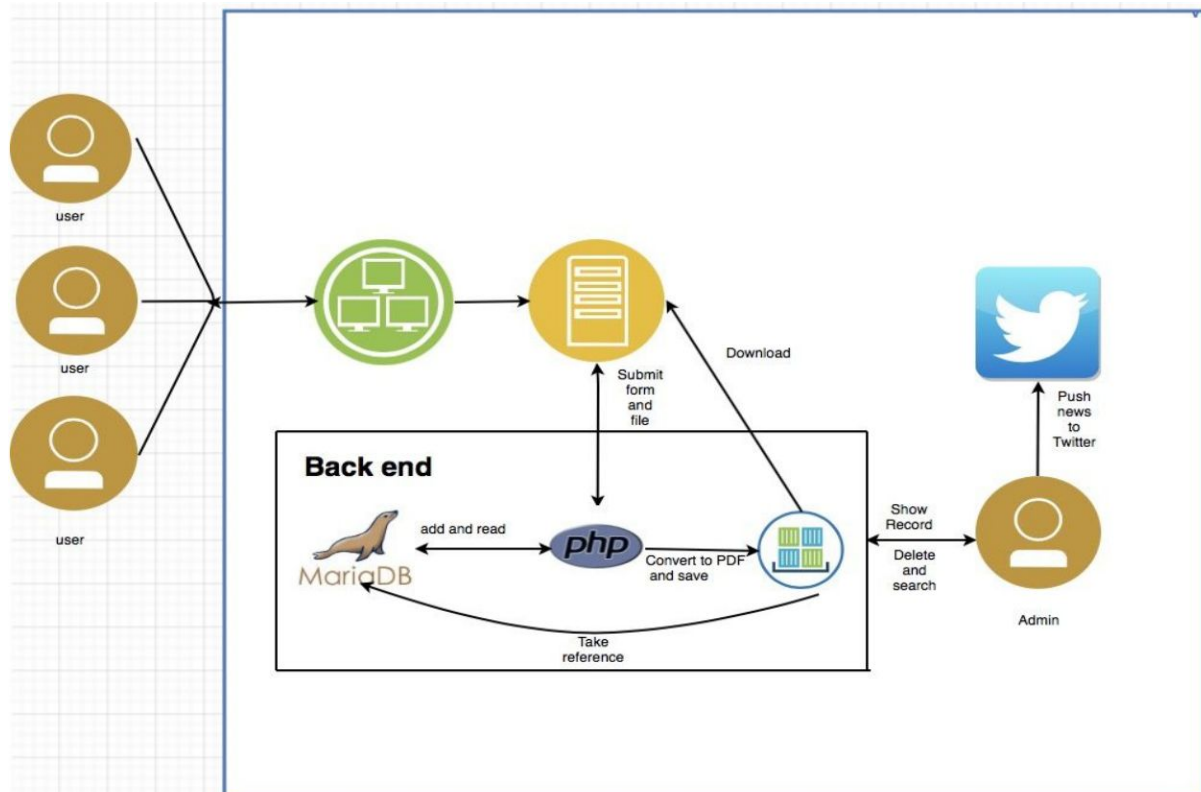
## 4.2 Development tools

- Communication: Facebook
- File sharing: Github, Google Drive
- Testing: Firefox, Chrome and Safari
- UI design: VS Code
- IDE: PHPstorm
- Poster editing tool: Edrawinfo
- Video editing tool: iMovie
- Local server: Apache

## 4.3 Implementation of key functionalities

Across the whole app, we believe the users' experience is the most important element, hence we have to let each component operate without any disturbance. Under this regard, a linear design is our priority.

The design can be divided into two parts basically, user interface and back-end. The graph below depicts how front-end and back-end connect to each other.





### 4.3.1 Update new information on homepage:

When someone uploads news or posters, how to update the information and generate a new section for the news automatically on news home page?

Solution:

On user upload page, data will be uploaded to a table in database. On home page, I wrote lines of php code which will read data from database, and show all the data in the same pattern using a while loop. Like this:

```
if ($result->num_rows > 0) {  
while($row = mysqli_fetch_array($result)) {  
echo '<div >';  
// insert html here  
echo '</div>'; }}
```

### 4.3.2 How to automatically generate a link to a file?

Firstly, rename the uploaded file according to author name and title, and save the filename as a string in database. At the same time, save the renamed file inside a folder called 'upload'. Then in php files, I can get a link as below:

```
echo '<a class="navLink" href= "'.upload/'.$row["filenames"].'"> Download </a>';
```

### 4.3.3 How to convert a text file into PDF format?

We use the third party application to convert a uploaded file into PDF format. We use openoffice app and put the application onto server. Then type in `soffice -headless -accept="socket,host=127.0.0.1,port=8100;urp;" -nofirststartwizard` in terminal command line to activate openoffice to listen to port 8100. Then in the upload.php code, the code fragment below convert a uploaded file into a pdf file and rename the file according to title, author name and file name. Then save the pdf file in a folder named "pdffolder".

```
$command = 'java -jar  
/opt/jodconverter-2.2.2/lib/jodconverter-cli-2.2.2.jar  
'./opt/lampp/htdocs/news_portal/upload/'.$filenames.'
```

```

/opt/lampp/htdocs/news_portal/pdffolder/'.current($firtitle)."_"
current($firauthor). "_" . $filename. '.pdf';
system($command);

```

#### 4.3.4 pagination

For some webpages, we have to do pagination to avoid hundreds of records in one page. Firstly, we find out the mathematical relation between record number and page number.

```

$result = mysqli_query($conn,"SELECT * FROM newsrecords");
$recordNum = $result->num_rows;
$record_each_page = 5;
$pageNum = ceil ($recordNum / $record_each_page);

```

Secondly, we display corresponding records in the current page.

```

$count = 0;
while($row = mysqli_fetch_array($result)) {
    if($count<$page*$record_each_page and
    $count>=($page-1)*$record_each_page)
    {
        // display corresponding records in pattern.
    }
    $count ++;
}

```

Then, for the pagination bar, we pass \$\_GET["page"] variable by generating a link when click on the page number.

```

echo'<ul class="pagination">
<li><a href="index.php?page='.$prevPage.'"><<</a></li>';
for($c = 1; $c <= $pageNum; $c++)
{
    if($c == $page)
    {
        echo '<li><a class="active"
href="index.php?page='.$c.'">'.$c.'</a></li>';

```

```
}  
else{  
  
echo'<li><a href="index.php?page='.$c.'">'.$c.'</a></li>';  
}  
}  
echo'<li><a href="index.php?page='.$nextPage.'">>></a></li>';  
echo'</ul>;
```

## 5. Testing

---

### 5.1 Testing strategy:

Considering our use case, we decide to test compatibility and responsive design of the web pages. And then we will ask some people to open our website on their devices and record the results.

### 5.2 Compatibility testing

We choose 4 most popular browsers to test compatibility.

Browser	Version No.	Result
Chrome	65.0.3325.181(64-bit)	Great
Firefox	59.0.2 (64-bit)	Great
Safari	11.1	Great
Edge	41.16299.371.0	Great

### 5.3 Responsive design testing

We choose 4 most common screen size to test responsive design. First

Screen size	Result
Desktop (width 1920px on Chrome)	Great
Desktop (width 1440px on Chrome)	Great
iPad (width 2048px on Safari)	Great
iPhone 6s (width 1334px on Safari)	For admin.php page, the data in the table cannot be displayed fully.

We assume our client administrator will use desktop and laptop more often to manage data and information. So, the current version is acceptable.

## 5.4 User acceptance testing:

Functions	Tester 1	Tester 2	Tester 3	Tester 4
Upload file	Pass	Pass	Pass	Pass
Download file	Pass	Pass	Pass	Pass
Admin log-in	Pass	Pass	Pass	Pass
Delete records	Pass	Pass	Fail	Pass

Tester 1: Using chrome browser on 1920px laptop.

Tester 2: Using chrome browser on 2048px laptop.

Tester 3: Using an integrated browser(not popular) on 1920px android smart-phone. (On admin page, an unknown error occurs, the table cannot be displayed, and for other IXN pages, the css seems not working. It is a compatibility problem)

Tester 4: Using Firefox browser on 1920px desktop.

## 6. Conclusion and future work

---

## 6.1 Summary of the achievement :

The achievement of this project can be measured by comparing the functions we achieved in both tasks with our requirement list. At this point, we finished all the “must have”, “should have” and “could have” requirements at an acceptable level, there is no doubt that our project is successful to some extent. But we still have limitations on a few functions such as “push to twitter” and “pdf conversion”. The feedback from the client will be an important standard of judging the project as well.

### 6.1.1 Achievement table

Achievement table for task 1:		
Portal for students to upload media (desktop version )	Implemented	Must have
pdf generation	Implemented	Must have
Tags for different subjects	Implemented	Should have
admin access for Lucy Tallentire	Implemented	Should have
Conversion functionality of student posts into Tweets	Partial Implemented	Should have
Login	Implemented	Could have
Mobile version	Implemented	Could have
Achievement table for task 2		
List of job offer	Implemented	Must have
Expire date	Implemented	Must have
Tags for job keywords	Implemented	Must have
Description of the job offers	implemented	Should have
Comments	Implemented	Could have
Feedback	Implemented	Could have

### 6.1.2 Individual contribution table

Workplace	Yuxuan Li	Tiefu Cai	Pierre-Alexandre Gruman
Client liaison	40%	20%	40%
Requirement analysis	40%	30%	30%
Research	60%	20%	20%
UI Design	80%	20%	0%
Programming	70%	20%	10%
Testing	0%	0%	100%
Progress Reports	60%	20%	20%
Group Report	40%	40%	20%
Poster Design	0%	70%	30%
Video editor	0%	0%	100%
<b>Overall contribution</b>	60%	25%	15%
<b>Roles</b>	Front-end developer, back-end developer, Researcher	Poste Designer, UI designer, Report Editor	Vedio maker, Report Editor, Tester

### 6.1.3 Known bug list

ID	Bug description	Priority
1	On news portal upload page, when uploading a image, the image file will not be converted into pdf format.	Normal

## **6.2 Functional Achievement :**

“Must haves” are the most essential features of the whole project, and we implemented them successfully. In addition, different functions have a different difficulty levels, hence the relative easy items were our priority during the whole project developing period. But for a few functions, even though we managed to work them out, they might not be perfect in some cases.

PDF conversion function:

We use the third-party application, OpenOffice to convert an uploaded file to pdf format. But it has some limitation. For example, not every format file can be converted to pdf format by OpenOffice, like png, jpg and other image files.

Push a post to Twitter account:

We referred a lot to Twitter official API, but current function can only link to twitter login page We cannot pre-edit message in php file and then generate a tweet.

## **6.3 Critical evaluation:**

Although we finish all main functions and the outcome of the whole project is positive, it still has so many points of improvement that could further enhance. There are some negative appeals of our project.

UI(user interface):

The user interface we designed is not perfect. Only a few of color in the main pages, and pages are very simple. Furthermore, the script functions are very limited.

Functionality:

We almost finished all the functions, and all the function we finished have been tested, however, the twitter login function is partially implemented, so the test we implemented in this function might not be extensively enough. Except this, there still have many improvements we need to achieve, like the pdf generation function cannot convert the image file and the admin cannot delete the files on the web. But the app really can satisfy about 90% of the requirements the client gave us. Hence the functionality of this app is basically successful.

Compatibility:

Due to some shortcoming of this app, the compatibility might not be perfect, especially for the mobile version, so many situations we didn't take into account

during the test. So, the compatibility might not reach a very high level, but for the most common use cases, it is acceptable.

Maintainability:

It is really hard to determine the app's maintainability. It has all basic features that the client assigned to us, but it might not work perfectly under some situation we haven't expected. But the source code's structure is clear and it is not difficult to make modifications.

Project Management:

The project management is quite organic, everyone in the team could make contributions to what themselves good at.

For programming task, we allocated each team member tasks according to each one's ability. And we use google online document editor(PS: Google online file editor which allow different people to edit a same file at the same time) to build the project schedule, set inner-team deadline and function list, so as to plan our time on this project more effectively and once we have done a function, we will tick off it in the list, then everyone can know the process of the project. Then, we will merge our latest code on github, which is accessible to all of us.

## **6.4 Future Work:**

As the limited time only allows us to deliver a prototype of the app to the client, we still have so many chances in the future for us to improve the app. Firstly, if we got more time, we will work on improving the user interface of our web pages. Secondly, because the compatibility is not perfect yet, we would like to improve the compatibility for small screen devices. Moreover, we also plan to upgrade the security of admin login web pages. In the future, some bugs we haven't met might be exposed, and we might be more likely to implement some additional high-level functionalities using the professional knowledge we accumulated. The Tweeting functionality for the News Portal has yet to be fully implemented. A little more expertise on the matter could have helped us significantly. We will most definitely be able to achieve this later in our degree.

## **7. Appendix**

---



## 7.1 User Manual

For both tasks:

Username for admin\_login page: admin

Password for admin\_login page: password

## 7.2 Deployment manual

### 7.2.1 For both tasks:

To change admin username and password, please change file admin\_login.php file line 3 and line 4. `$usr = "admin"; $psw = "password";` Replace "admin" by your new username, and replace "password" as your new password.

To connect database and create a database table, please change cretable.php file line 2 to 5, replace `$servername = "localhost:3306"; $username = "root"; $database = "newsdb"; $password = "";` by your server name, username, database name and password. And then run this file in your server to create a table in your database. Then in config.php, please do the same thing to replace server name, username, database name and password by yours.

### 7.2.2 For News Protal task:

To use the third\_party app to convert uploaded file to pdf format, client should enter:~\$  
soffice -headless -accept="socket,host=127.0.0.1,port=8100;urp;" -nofirststartwizard  
in server command line. And in source code news\_portal/upload.php file, change line 79 from

```
$command = 'java -jar  
/opt/lampp/htdocs/jodconverter-2.2.2/lib/jodconverter-cli-2.2.2.jar  
'./opt/lampp/htdocs/news_portal/upload/'. $filenames.'  
/opt/lampp/htdocs/news_portal/pdffolder/'.current($firtitle)." ".  
current($firauthor). " ". $filename. '.pdf';
```

to

```
$command = 'java -jar  
/absolute_path_of_jodconverter/jodconverter-2.2.2/lib/jodconverter-cli-2.  
2.2.jar './absolute_path/news_portal/upload/'. $filenames.'
```

```
/absolute_path_of_news_portal/news_portal/pdffolder/'.current($firtitle)."  
_". current($firauthor). "_". $filename. '.pdf';
```

Where `absolute_path_of_jodconverter` is the absolute path of `jodconverter-2.2.2` folder in source code and `absolute_path_of_news_portal` is the absolute path of `news_portal` folder in source code.

## 7.3 Code citation

NO.	Function Name	Code File Name	Source
1	*	ixn/index.php	<a href="http://ixn.org.uk/">http://ixn.org.uk/</a>

Actually, we cited a lot style code from the IXN website, because of the task's nature.

## References:

---

[1] "Twitter Developer Documentation - Tweet Button", [online], available:

<https://dev.twitter.com/web/tweet-button>

[2] "12 'Must-Know' Advantages of PHP", 27/6/2015, [online], available:

<http://www.vandelaydesign.com/advantages-of-php>

[3] "10 reasons to migrate to MariaDB", 9/1/2015, [online], available:

<https://seravo.fi/2015/10-reasons-to-migrate-to-mariadb-if-still-using-mysql>

[4] "Top advantages with Apache servers", [online], available:

<https://www.instantssl.com/articles/top-advantages-with-apache-servers.php>

[5] “10 Microsoft Azure Business Benefits in One Convenient List”, 26/3/2018,  
[online], available:  
<https://www.redpixie.com/blog/top-ten-microsoft-azure-business-benefits>