**Design Documentation Deliverable:**

**Proofreaders Website**



|  |  |
| --- | --- |
| Date | 27/02/2017 |

|  |  |  |
| --- | --- | --- |
| **Student ID** | **Student Name** | **Signature** |
|  |  |  |
| 15148823 | Colm Le Gear | signature_vemmjjge3c1bhpxl0e |
| 13150154 | Ryan Harris | 17035507_1652033558145290_970497667_n |
| 14164736 | Liam Horgan | 17012455_1244944435588796_1099016412_n |
| 15169235 | David Kearney |  |

*1.0 Purpose of website*

The purpose of **Proofreaders** is to provide users with an interactive web platform where they can have their academic papers proofread by fellow students and staff.

Almost every student will need their work to be reviewed at some point during their studies. Some students may ask their friends to proofread their papers but they may lack the required knowledge to provide helpful, constructive criticism.

In some cases students may ask other students in their course to proofread their work. However some students may be hesitant to go down this route as they fear that their work will be plagiarised. They may also worry that the other students will not be honest with them about the faults in their papers due to competition when it comes to achieving the highest grades.

This puts students looking to improve their work in an awkward position. They might end up with papers that are poorly reviewed or worse, not reviewed at all. Some may seek professional help if they can afford it, but of course this isn’t possible for everyone, particularly the students.

**Proofreaders** provides users with a helpful, educated community that can offer assistance and advice. Users can create an account and submit their documents to be reviewed. Other users will see a list of tasks on the website’s homepage that are available for review. This list will be different for every user as it shows tasks that are best suited to each particular user.

*1.1 User Types*

There are two user types for the proposed website.

1. **The Students,** will be able to create tasks for other users to review. They will also be able to browse a list of available tasks.
2. **The Moderators**, will be able to access flagged tasks. They can immediately un-publish flagged tasks. They have the ability to ban users who created any of the flagged tasks.
3. *Justification of Functionality*

The functionality afforded to users of the proposed site will be in some respects limited to a few simple operations. Users will be able to log in using two text boxes, one for email and the second for a password. Once both fields have been completed a sign in button will be clicked granting the user access to the system if the supplied information is correct.

Once logged on, users can view a list of available tasks that are best suited to them. When the user clicks on a task, further details are shown about the task to the user. Users can also use a search box to input their own parameters.

Clicking on a user’s name will direct the user to that profile.

*1.3 Potential Ramifications*

The potential ramifications of such a site on users could be the emergence of a new way for students to source assistance with their academic papers. The site will provide students with an educated community that can offer valuable assistance with their college work. The site could eliminate many issues that students currently face when trying to get their work reviewed, such as plagiarism and uninformed peers proofreading their work. Instead of students having to ask their friends to review their work they can instead use the site and have it reviewed without ever having to leave their room.

Scope also exists for the proposed site to be expanded in future to allow users to offer a financial reward for satisfactory completion of their task. This feature would attract users to the website that are looking to make some extra money while they pursue their studies. The option to extend the communication power of users is also available through the implementation of chat rooms, forums and instant messaging.

*2.0 Detailed Description*

When the proposed website is launched a homepage similar to figure 3 (see Appendix 1: Webpages TODO) will be displayed. No processes are required to generate this page. From the homepage users can click buttons to navigate to other pages and initiate processes.

P1 (Registration) is executed when a user wishes to register their account. Users will supply information to a webform on the homepage and submit it. The input is validated and if it is correct the information is passed to the database and the new entries are added in the tables and the user is directed to their new profile page (see figure 4 TODO Appendix 1: Webpages). If incorrect information is supplied the appropriate error message is displayed. E.g. the incorrect fields will be highlighted.

P2 (Login) when a registered user wishes to login they supply their email and password. This process is executed when these details are submitted at the login area on the homepage. The details are validated against the details in the database. If the user is authentic the process directs them to their profile page. If the supplied details were incorrect an appropriate error message will be displayed informing the user their details were incorrect and to re-enter them.

P3 (View task details) is executed when the user clicks on a task. This shows more details of the task to the user. From here the user should be able to download the preview document attached to the task.

P4 (Claim task) is executed when a user wishes to claim the task after viewing the task’s details. Once a user claims the task will be removed from the task list so that two users cannot claim the same task.

P5 (Flag task) is executed when a user chooses to flag a task that they deem to be inappropriate. This sends the task to a list of flagged tasks that the moderators can review. The user will be awarded two reputation points for flagging a task.

P6 (Upload task details) is executed when a registered user has supplied the information needed to a webform on the homepage to add a document to be reviewed. A user will provide a task title, description of the task, type of task (e.g. project report), tags to describe the task (e.g. html5), number of pages and number of words in the document, three sample pages of the document, source file format and a deadline for claiming the task as well as a deadline for task completion. These details are validated and if they are in the correct format the new information is added to the relevant tables in the database.

P7 (Cancel task) is executed when a user wishes to cancel a task. They will be deducted fifteen reputation score for this action.

P8 (Task completion) is executed when a user desires to mark completion of a task. They will be required to provide a brief textual review.

P9 (Access flagged tasks) is executed when a moderator wishes to view the list of flagged tasks. This shows a list of tasks that have been flagged by registered users.

P10 (Un-publish task) is executed when a moderator wants to remove a flagged task from the list of available tasks for users to review after verifying the task details.

P11 (Ban User) is executed when a moderator decides to ban a user that has uploaded a flagged task.

*3.0 Benefits and Limitations*

The benefits of the proposed website include having a knowledgeable community that are willing to help students with their academic papers. The website will act as a means for students to seek assistance with their college work and assist their peers with their papers.

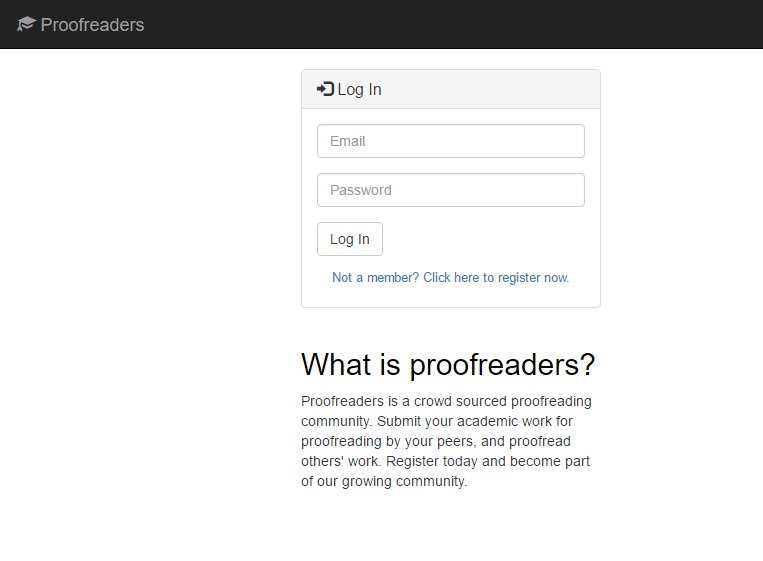
The limitations of the proposed website include only allowing the end user to upload specific information, mostly task related details.

*4.0 Technologies*

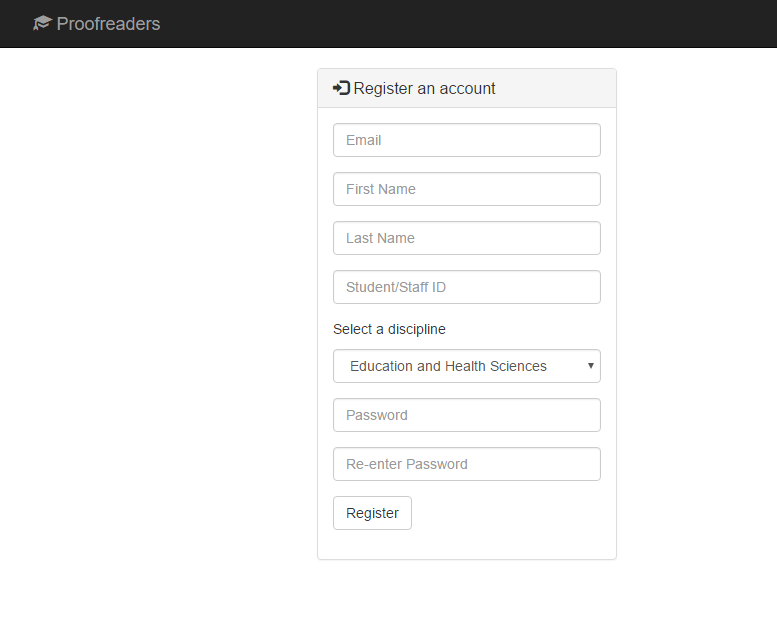
Several languages and technologies will be utilised during the development of this project.

* **Microsoft Windows 7 Operating System** will be used as the operating systemfor developing the proposed website.
* **PHP** will be used to generate dynamic webpages.
* **HTML5 & CSS3** are languages used to generate the appearance of webpages.These languages are used together. HTML5 consists of tags which dictate where information appears on a webpage. CSS3 is used to assign styles to these tags such as colour, font, positon, margins etc. Both these languages will be used to design the appearances of the webpages.
* **MySQL** is a relational database management system.Itwill be used as theprimary means of creating a database for the proposed website.
* **Git/GitHub** (<https://github.com/CS4084/CS4084>) as the main version control repository.
* **Bootstrap** HTML, CSS, and JS framework for developing responsive, mobile first projects on the web. It will be used for designing the front end of the site.

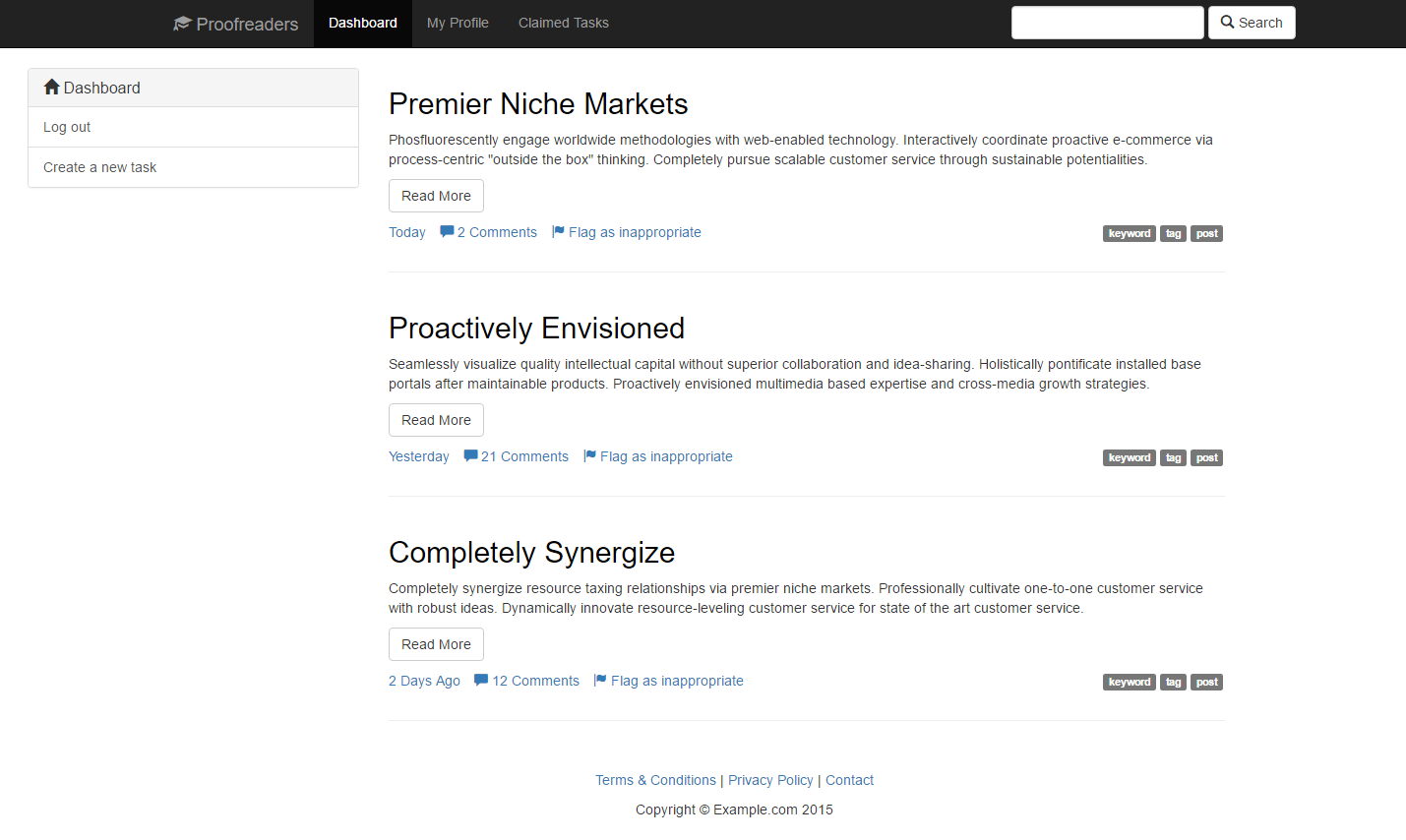
*Appendix 1: Web Pages*



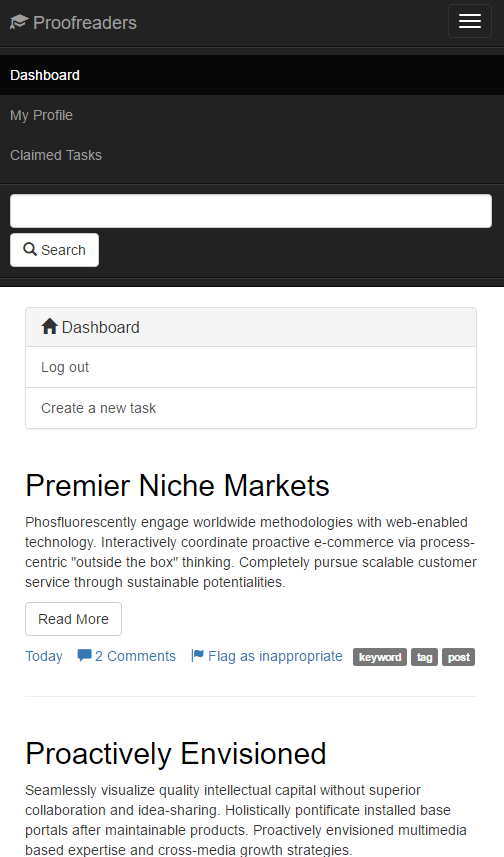
**Log in Page**



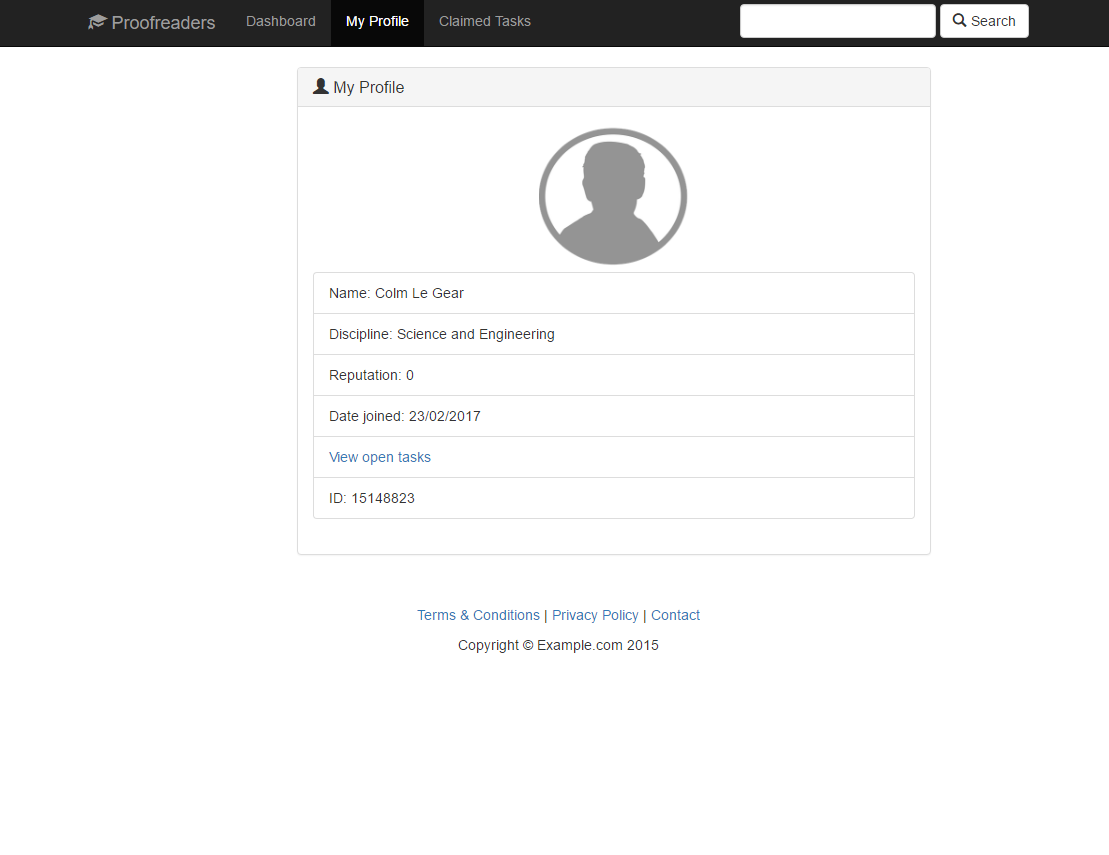
**User Registration Page**



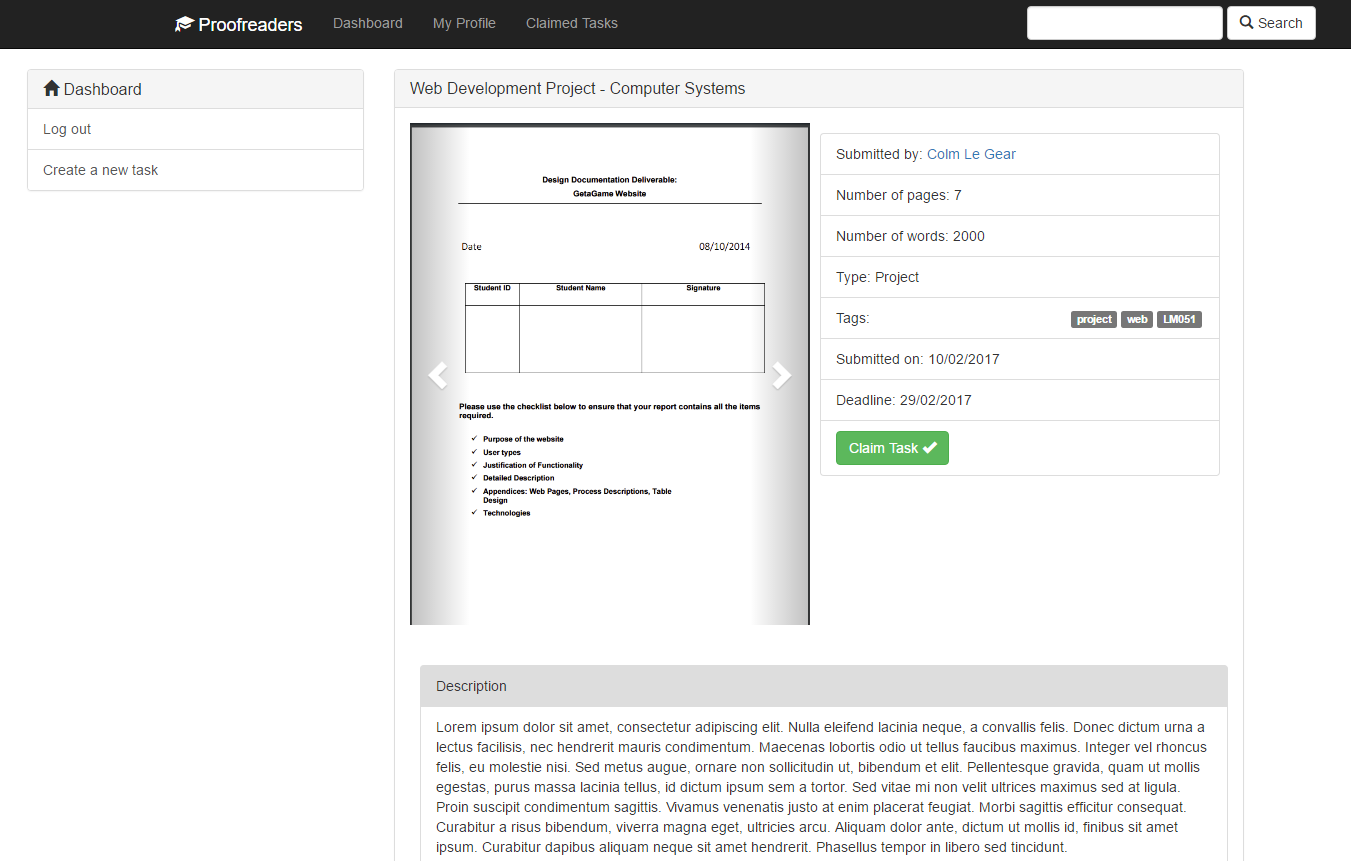
**Dashboard: Logged in homepage**



**Dashboard – Mobile View**



**User Profile Page**



**Task Details Page**

***Appendix 3 : Database Tables***

|  |
| --- |
| CREATE TABLE `students` ( |
|  |  |
|  | `student ID` int(11) DEFAULT NULL, |
|  |  |
|  | `student name` varchar(45) DEFAULT NULL, |
|  |  |
|  | `students college` varchar(45) DEFAULT NULL, |
|  |  |
|  | `course code` varchar(45) DEFAULT NULL, |
|  |  |
|  | `student email` varchar(45) DEFAULT NULL, |
|  |  |
|  | `student password` char(64) DEFAULT NULL, |
|  |  |
|  | `user id` int(11) NOT NULL AUTO\_INCREMENT, |
|  | UNIQUE KEY `user id\_UNIQUE` (`user id`), |
|  |  |
|  | UNIQUE KEY `student ID\_UNIQUE` (`student ID`), |
|  |  |
|  | UNIQUE KEY `student email\_UNIQUE` (`student email`) |
|  | ) ENGINE=InnoDB DEFAULT CHARSET=utf8 |
| |  | | --- | | CREATE TABLE `tasks` ( | |  |  | |  | `Title` varchar(20) NOT NULL, | |  |  | |  | `Type` varchar(45) NOT NULL, | |  |  | |  | `Description` varchar(100) NOT NULL, | |  |  | |  | `Tags` varchar(100) DEFAULT NULL, | |  |  | |  | `Number of Pages` int(11) NOT NULL, | |  |  | |  | `Number of Words` int(11) NOT NULL, | |  |  | |  | `Source File Format` varchar(45) DEFAULT NULL, | |  |  | |  | `Deadline Date - claiming` int(11) NOT NULL, | |  |  | |  | `Deadline Date - completion` int(11) DEFAULT NULL, | |  |  | |  | `Claimed Tasks` tinyint(4) NOT NULL AUTO\_INCREMENT COMMENT 'Boolean', | |  |  | |  | `Flagged Tasks` tinyint(4) DEFAULT NULL COMMENT 'Boolean', | |  | `Completed Taks` tinyint(4) DEFAULT NULL COMMENT 'Boolean', | |  |  | |  | PRIMARY KEY (`Claimed Tasks`,`Title`,`Number of Pages`,`Number of Words`) | |  | ) | |  | ENGINE=InnoDB DEFAULT CHARSET=utf8 | |  |