Design Documentation Deliverable

# Date 3rd March 17

# Group Name/Number Group 16

|  |  |  |
| --- | --- | --- |
| **Student ID** | **Student Name** | **Signature** |
| **16118928** | **Matthew Riley Neil Timothy Paul McCarthy Richard O’Connell** |  |

Please use the checklist below to ensure that your report contains all the items required.

* **Purpose of the website (a brief introduction)**
* **User Types**
* **Justification of Functionality**
* **Detailed Description**
* **Appendices:** 
  + **Web Pages (mock ups of main pages)**
  + **Process Descriptions**
  + **Table Design**
* **Technologies**

**Proofreading Website - PlayByPeer**

1. **Purpose of Website**

The purpose of PlayByPeer is to provide an interactive web platform to facilitate proofreading services for users. The service is available for all academic papers, such as theses, dissertations, assignments and research papers.

Currently if students wish to avail of a proofreading/review service, they may find themselves restricted in their ability to do so; either financially, or by subject specialisation. By creating a dedicated proofreading platform, such as PlayByPeer, users can experience the benefits of a collaborative effort from a community consisting of users and peers.

In its simplest form, PlayByPeer will give registered users access to a secure platform on which they can upload, browse, or select academic documents which require proofreading or review.

* 1. **User Types**
     1. Students (basic user). Once registered, students will have the ability to upload a sample of their documentation which they require to be proofread/reviewed by the user community. Additionally, all students will be able to browse documents (tasks) which have not yet been ‘claimed’ by other users. If suitable, users
     2. Moderators (enhanced user). In addition to the functions available to a student user, moderators have the additional functionality of being able to view a list of tasks which have been ‘flagged’ by other users as potentially being inappropriate; and also remove such tasks if necessary. Moderators will also have the ability to permanently ban other users from accessing the website.
     3. Administrator. Administrators will have the required permissions allowing them to alter/change any aspect of the site’s appearance, functionality or database records. Initially, the 4 group members listed above will be the only users with Administrator privileges.

**1.2 Justification of Functionality**

The 5 core functions of the system will be:

* Register as a User
* Browse the Catalogue of Available Tasks
* ‘Claim’ Tasks
* Create Tasks
* Leave Reviewer Ratings

1.2.1 Register as a User

On the landing page, visitors will only have 2 options: to Register, or to Log In. When registering, a user will be required to provide standard profile information, such as name, age, email address, etc. Additionally, users will be required to select their area of study/expertise/interest to be used by the system to display dynamic content that is user-targeted and relevant. Such registration processes are common and have been widely implemented by a large number of profile-based user applications, such as Facebook, LinkedIn, Amazon, etc.

Furthermore, In order to be eligible to register, a proposed user must hold a valid email account from a domain(s) specified by the site administrator; e.g. .ul.ie. By only allowing holders of email accounts from a specified domain(s) to register, some level of verification can be performed to check that users are indeed affiliated to the selected organisation or institution, e.g. Colleges, Universities, etc. This verification method is common in organisations where affiliation to a particular organisation is a prerequisite to obtaining a user account, where members are permitted to access particular information, such as library documents, employee information etc.

1.2.2 Browse the catalogue of available tasks

To have visibility of the task catalogue, all visitors to the website must first be registered (see 1.2.1) and be logged-in to the system. The justification of this is primarily to maintain security and confidentiality of users’ personal data; including uploaded academic data. This particular aspect of functionality is a common feature in sites such as dating websites, online (private) media catalogues, company-employee information, etc.

Once logged in, users can browse the entire catalogue of tasks, with the tasks most relevant to the users’ field of expertise displayed first. In addition to the areas of expertise selected in their profile, there will be the added functionality to modify the entries of the catalogue which are presented to the user based on their session/browsing habits of the site. This will be manifested by dynamically displaying tasks to the user on their homepage and/or when browsing the catalogue.

1.2.3 ‘Claim’ tasks

If a user would like to complete a task listed in the catalogue, they can do so by ‘claiming’ the particular task. Once claimed, a task is removed from the catalogue, and assigned to the user that claimed it. The website then facilitates contact (email) between the user who claimed the task and the and the creator (see 1.2.4).

1.2.4 Create Tasks

Once registered, users can create a proofreading/review task on the system. Details that users are required to provide in relation to the task include: Task title, description, deadlines, subject, etc. Full details required will be outlined in section XXX of this document.

1.2.5 Leave Reviewer Ratings

The ability for users to leave feedback or a rating system for other users is a widely-implemented methodology, examples of which include Amazon, boards.ie, etc. This process effectively allows users to ‘self-manage’ the online community of that site, highlighting to administrators and other users those who contribute to a high standard and, conversely, those who contribute to a low or negative standard. This functionally also has the potential to generate ‘prestige’ within the user community, thus an added incentive for users to make contributions to the service that are of a high standard.

**1.3 Potential Ramifications**

As with any online community collaboration platform, there will be potential benefits and ramifications to the users and, potentially, the wider community. It is therefore essential that benefits are maximised to their full potential where possible, whilst risks and adverse consequences are removed or mitigated to a level that is as low as reasonably practicable.

1.3.1 Benefits and advantages

The benefits of online community collaboration are evident with sites such as Wikipedia, boards.ie, etc. where vast amounts of information can be produced and maintained by an online community. The fact that people are volunteering to assist other users shows that, in the majority of cases, the intentions of the community are for the greater good and without ulterior motives.

Users that have their work proofread/reviewed will be the most obvious beneficiaries of the website, using the experience and abilities of other users to improve the quality of their work. Additionally, however, the users that perform the proofreading/reviewing of other users’ documents will also have the potential to benefit from participating in the community, as the site can help facilitate the development of their peer-review ability and critical thinking, whilst potentially building on their knowledge of new and existing fields of expertise.

1.3.2 Adverse and unintended consequences

By sharing academic documents with the site, or other users, there is the inherent risk to the confidentiality of the data. To address this issue, when registering, all users should have to read and agree to Terms and Conditions (T&Cs) of using the system. Within the T&Cs it should be stated explicitly that all media uploaded and/or shared on the site is done so at the user’s risk. Alternatively, it could be stated in the T&Cs that users are not to reproduce, use, publish etc any content obtained from the site or any other user. Whilst the latter suggestion may be the ideal approach, realistically the former would be the most suitable/feasible to implement.

As online plagiarism checks are becoming commonplace in Colleges and Universities, there is a possibility that a user who has uploaded a sample of their document onto the website could find themselves accused by their institution of plagiarism. The most effective ways to mitigate this issue would be to prevent online plagiarism checkers from accessing sample documentation, and/or inform the institutions that the PlayByPeer website should be excluded from the plagiarism search.

Although the intentions of users may be well placed, there is always the potential that an issue will arise regarding the quality of the work undertaken, missed deadlines, disagreements, etc. Other than the functionality of Reviewer Ratings (see 1.2.5), users have little to no recourse to address their grievances with another user.

Although the concept of a ‘self-regulating’ community using moderators (see XXX) can be a resource-efficient model, there are also inherent risks involved with this practice. The main risk would be of ‘rogue’ moderators, who act in their own, possibly malicious, interest. This could include banning users unnecessarily, banning other moderators to maintain dominance, removing legitimate documentation or tasks from the website, etc. In addition to the discontent this would generate within the users, the actions of a rogue moderator could significantly undermine the credibility and reputation of the website. To address this, control measures should be implemented, such as limiting the number of actions a moderator can perform in a certain time, e.g. the ability to only ban x number of users in a 24hr period. Furthermore, moderators should not be able to individually ban other moderators without the approval of other moderators and/or the site administrators.

**2.0 Detailed Description**

*Description of how proposed system will work. Include flowcharts to describe overall process; individual processed etc.*

*Referring to information in three Appendices - Processes, Web pages, Tables.*

When the Home Page of the PlayByPeer is launched, visitors are presented with a page similar to that shown in fig XXX.

As there are only 2 options available to visitors at this point, to sign up or log in, no processes are generated at this point.

To produce this page several processes are executed. Process **P1** is executed to produce a unique list of restaurant names that are displayed at three locations on this webpage (so that a user can select a restaurant when he/she wishes to write a review, when he/she opts to read reviews, and when he/she chooses to visit a restaurant home page).

**P2** is executed to display the reviews for a restaurant.

**.....................................................................................................**

**.....................................................................................................**

**.....................................................................................................**

**4.0 Technologies**

*You should describe what kind of technologies, e.g. operating system, PHP libraries, frameworks and so forth you wish to use.*

...

**Appendix 1: Web Pages**

**Figure 1:**

**Appendix 2: Tables**

*For each table state its name and its primary identifier. In addition for each attribute state its name, its data type, its length and any constraints.*

**Table Name:** Restaurant

**Primary ID:** RestaurantID

**Description:** The Restaurant table stores for each restaurant its name, the restaurant manager's name, a brief description of the restaurant, its location, its opening hours and the name of the image file to display for this restaurant.

|  |  |
| --- | --- |
| a1.jpg | * **RestaurantID** each restaurant is uniquely identified by a number, permissible numbers are in the range 1 to n where n is the number of restaurants * **RestaurantManagerName** each restaurant is managed by a manager, a manager may manage more than one restaurant. This attribute is alphabetic and up to 30 characters in length   ... |

***Table 2 details***

***Table 3 details***

***Table 4 details***

***Table 5 details***

***Table 6 details***

***Table n details***

**Show relationships between tables**

|  |  |
| --- | --- |
| Many reviews can be written for the same restaurant. A Review is written for a restaurant, each Review is assigned a ReviewID, a review happens on a particular date and is written by a reviewer. Each review has a helpfulness indicator and a not helpfulness indicator. A Review may have Review Text and a Review must have rating of quality, value for money, service and range of dishes.  ...  ... | a1.jpg |

**Appendix 3: Processes**

|  |  |
| --- | --- |
| **Process Number** | P1 |
| **Process Title** | Registration |
| **Brief Description** | Brings user to new profile if supplied with correct information, rejects information if not suitable to field that it is inputted in. |
| **Inputs** | User’s information for creation of new profile inputted via registration form. |
| **Detailed Description** | In order for new profile to be created, registration form must be submitted. Users most fill out form and put correct information in various fields of form. If correct input supplied to all fields, form can be successfully submitted. If incorrect information provided, users will be asked to resubmit their details. The information requested by the form is First Name, Second Name, Student or Staff Id, Email, Major Subject, Password and Password re-entry. If these details correct user table is then uploaded. |
| **Output** | Creates and bring users to their new profile. |

|  |  |
| --- | --- |
| **Process Number** | P2 |
| **Process Title** | Login |
| **Brief Description** | Validates username and password provided from the user, either restricts or provides access to user’s account. |
| **Inputs** | Username & Password. |
| **Detailed Description** | User will enter their unique username along with associated password into two fields. If details are incorrect then no access to account. Process validates input from two fields and if they are correct provides user with access to his/her account. |
| **Output** | Brings users to their profile. |

|  |  |
| --- | --- |
| **Process Number** | P3 |
| **Process Title** | Create task |
| **Brief Description** | Registered students create tasks to be proof read by users. |
| **Inputs** | Task details and documentation information submitted by user in task creation form in order to successfully publish new task. |
| **Detailed Description** | User enters details responding to task creation form, must correctly provide all task details – Task title, task type, task description, tags, number of paragraphs, word count, file format, file sample, deadline for claiming task/deadline for task completion. Process validates input and if correct task is created and published to reviewed. If input not correct user will be asked to resubmit task details. If the input is correct the task database is updated. |
| **Output** | Creates a task, publishes users documents to be reviewed by peers. |

|  |  |
| --- | --- |
| **Process Number** | P4 |
| **Process Title** | Browse tasks |
| **Brief Description** | Retrieve and generate a list of tasks that have not yet been claimed. |
| **Inputs** | User’s discipline taken from profile |
| **Detailed Description** | Query tasks database. Retrieve and generate a list of tasks published by other students that are available/unclaimed. Task’s appear which are relevant to student’s discipline and in order of nearest to deadline first. |
| **Output** | A list of available related tasks for user in order of nearest to deadline first. |

|  |  |
| --- | --- |
| **Process Number** | P5 |
| **Process Title** | Get more details of task |
| **Brief Description** | Click onto task to read more details and get access to all information of task. |
| **Inputs** | No user input |
| **Detailed Description** | The registered user clicks onto a task, the task database is queried more details relating to the task ie number of pages, word count, associated tags ect. Should appear. The user is then able to download preview of documentation associated to that task. |
| **Output** | All information and data relating to that task, preview documentation of task. |

|  |  |
| --- | --- |
| **Process Number** | P6 |
| **Process Title** | Claim task |
| **Brief Description** | Registered users can claim an unclaimed/available task. |
| **Inputs** | No user input |
| **Detailed Description** | The registered user clicks onto a task, the registered user is then able to claim task. Once this user claims task, the task database is updated, the task is removed from the task list to prevent another user claiming same task. Once task is then claimed the user is rewarded 10 marks to his/her reputation score. |
| **Output** | Task removed from task list, user rewarded 10 marks to reputation score. Task claimed message appears on screen to user. |

|  |  |
| --- | --- |
| **Process Number** | P7 |
| **Process Title** | Flag/report task |
| **Brief Description** | If task seen as inappropriate by registered user, a task can be flagged/reported. |
| **Inputs** | No user input |
| **Detailed Description** | If a user discovers a task when browsing to be inappropriate they can flag/report it to the moderator. They are then awarded points of 2 marks to their reputation score. Task and user database is updated. |
| **Output** | Marks rewarded to user. Message saying flagged task reported appears on screen. |

|  |  |
| --- | --- |
| **Process Number** | P8 |
| **Process Title** | View all claimed tasks |
| **Brief Description** | Registered users can view a list of their claimed tasks |
| **Inputs** | No user input |
| **Detailed Description** | Task database queried. If registered user is logged on, the user can view a list of claimed tasks, sorted in order of nearest deadline first. |
| **Output** | A list of available tasks claimed by the user in order of nearest to deadline first. |

|  |  |
| --- | --- |
| **Process Number** | P9 |
| **Process Title** | Select an option to action with claimed task |
| **Brief Description** | Registered users can click into a task from claimed task list. Then user is provided with options to do with that task. |
| **Inputs** | The user must select option from dropdown list. |
| **Detailed Description** | When registered user is logged on and clicks into view a task, If the task deadline has not been reached the user can pick one of the options 1) request full file from task owner 2) Mark task as completed 3) Cancel task claim. If the task deadline is gone the status of task is altered to cancelled and can not pick any of these options, the claimant is then penalized 30 marks from their reputation score. Update task and user database. |
| **Output** | If task within deadline, A personal email is sent to task owner requesting full documentation of task or task marked as completed or task marked as cancelled and claimant deducted 15 marks. If deadline for task has gone status of task changed to cancelled and task claimant deducted 30 reputation points. |

|  |  |
| --- | --- |
| **Process Number** | P10 |
| **Process Title** | My tasks |
| **Brief Description** | Registered users can view a list of their created tasks. Contact and rate claimant. |
| **Inputs** | No user input |
| **Detailed Description** | If registered user is logged in and has created tasked the user can view a list of their tasks. The retrieved list will show current status of either pending, unclaimed, claimed, cancelled or completed. If task marked as complete, the user cant provide feedback (Happy or Unhappy). Task and user database queried and updated. |
| **Output** | Retrieve a list of users published tasks, if claimed, show claimants email address, first and last name. Reward/Penalise claimant with points based on users feedback rating. |

|  |  |
| --- | --- |
| **Process Number** | P11 |
| **Process Title** | Promote user to moderator |
| **Brief Description** | User gets automatically promoted to moderator based on reputation score. |
| **Inputs** | No user input |
| **Detailed Description** | If registered user gets a reputation score of 40 or more, they automatically get promoted to moderator. User database updated. |
| **Output** | Change status to users profile to moderator. |