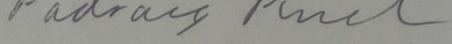
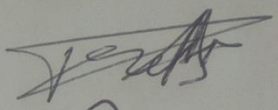
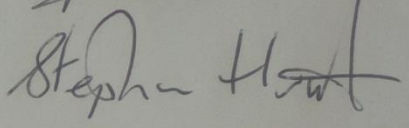
Design Documentation Deliverable

Date 03/03/2017

Group Name/Number Group 18

Student ID	Student Name	Signature
8487103	Padraig Punch	
15523108	Alam Syed Hasnanin	
16136616	Tresor Twendimbadi	
11106751	Stephen Hughes	

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

Date **03/03/2017**

Group Name/Number **Group 18**

Student ID	Student Name	Signature
8487103	Padraig Punch	
15523108	Alam Syed Hasnanin	
16136616	Tresor Twendimbadi	
11106751	Stephen Hughes	

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**

Purpose of the website

This is an open source project that aims to produce an interactive web platform to facilitate the proofreading of student theses, dissertations, assignments, research papers alike among students and staff. The main idea behind the website is to allow students to publish their academic documents and get them proofread/reviewed by peers.

User Types

There are two us user roles: Students and moderators. A moderator is any user whose reputation has reached a score of 40 or above. Students are allowed to submit and claim tasks and report tasks as in- appropriate. A moderator will check tasks that have been flagged as inappropriate and can un-publish these tasks and ban the user.

Justification of Functionality

The website allows users to register and submit proofreading tasks to be claimed by any users. The user who wishes to claim a task can browse a list of tasks look at their details and download a three page sample of the document before claiming. He also knows the deadline for completing the task. Once a task is claimed he emails the task creator for the full document. On completion of the task he updates the task status and sends the completed task to the creator. The task creator can browse through the tasks he has created and check their status – pending, claimed, unclaimed, completed, cancelled, and inappropriate.

Detailed Description

The specification as given is listed below:

• User roles:

- Two user roles desired: Students and Moderators

• Landing page:

- Initially, the website should display a landing page requesting users to register or login to use the system. None of the (proofreading/review) tasks should be displayed until a user is logged in

• User registration/sign-up

- At least the following profile information should be collected during the user-sign up process:
  - First Name
  - Last Name
  - Student or Staff ID
  - email
  - Major subject stream/discipline (select from a pre-populated list)
  - Password and Re-enter password

- Only emails from specified domain(s) should be allowed during the sign-up (e.g. emails from ul.ie)

• Create tasks

- Registered students should be able to create (proofreading/review) tasks
- When creating a task, at least the following information should be collected:
  - Task title
  - Task type (e.g. MSc thesis, BSc dissertation, project report, PhD thesis, Assignment, Conference Research Paper etc.)
  - Brief description of the task
  - Tags to describe the task (e.g. computer, html, project-report, cs4014) Maximum 4 tags allowed per task. This is an essential feature.
  - Number of pages in the document
  - Number of words in the document
  - Source file format (e.g. specify .docx, doc, open office, txt, pdf)
  - Sample of the document (e.g. three random pages from the thesis as a pdf) for preview purpose
  - Deadlines:
    - Deadline for claiming the task: task will be automatically unpublished after this date and time (i.e. the task will no longer appear in the website after this date/time, then the user may either create a new task or re-publish the task with a new claim date).
    - Deadline for task completion/submission (this date is the date that you wish to receive the proofread/reviewed document)
    - The above set up gives the task owner a reasonable timeframe to see whether anyone is interested in claiming the task; if not they can look somewhere else for a potential reviewer.

• Task list/stream

- Registered students should be able to browse available/unclaimed tasks published by the other students. A page should display a list of tasks best suited for the logged in user (e.g. tasks from the same discipline, user's favourite tags etc., user's subscribed tags). Ideally, try to identify/learn the characteristics of the most suitable tasks for the user from user's browsing habits of the website.
- Tasks with the task claim deadline closest to the current date should appear first.

• Task details

- Registered users should be able to click on a task to view more details (such as associated tags, description, number of pages, numbers words etc.) about the task
- User should be able to download the preview document attached to the task

• Task claim

- After viewing the details of a task, a user should be able to claim the task. Once someone claims a task, that task should disappear from the task list/stream (so that the same task cannot be claimed by two users).
- User Reputation
  - For each task claimed, the user should be awarded 10 marks to the user's reputation score

• Flagging tasks

- Any registered user should be able to flag a task as inappropriate
- User Reputation
  - When a user flags a task as inappropriate, the user should be awarded 2 marks towards the reputation score

• Claimed tasks (for task claimants)

- If the logged in user has claimed tasks, they should be able to view a list of claimed tasks, sorted by the task's completion deadline (task with the deadlines coming up soon should appear first)
- Clicking on a claimed task should provide the following options:
  - Full file request
    - Display a personalised email template requesting the task owner to send the full document for review (under no other circumstances should the system display other users' profile information, including emails due to privacy concerns).
  - Task completion (mark as complete)
    - System should allow a task claimant to mark the completion of a certain task (e.g. by proving a brief textual review)
  - Task cancellation
    - Task claimants are allowed to cancel tasks at any stage, but their reputation score should be reduced by 15 marks
  - Failed submission
    - If the task claimant fails to submit a task before the task's submission deadline, the status of the task should be changed to cancelled and a penalty of 30 marks should be deducted from the claimant's reputation score

• My tasks (task owners)

- If the logged in user has created any tasks, the user should be able to retrieve a list of created tasks, with their current status: pending claim, unclaimed (task claim deadline expired), claimed, cancelled (by the claimant) or completed.
- Task owner should be able to view the email address of the claimant (in addition to claimant's first name and the last name).
- Reviewer rating

        - Task owners should be able to provide feedback about the task claimant for each of the completed tasks, with simply two options: Happy (with their work/review), Not happy (with their work/review)
        - If chosen
            - 'Happy', 5 marks will be added towards the claimant's reputation
            - 'Not happy', 5 marks will be deducted from the claimant's reputation

• When a user gets a reputation score of 40 or more, the user should be automatically promoted as a moderator

• Moderators role:

    - Access flagged tasks
        - Moderators should be able to access a list of 'FLAGGED TASKS'
    - Un-publish task
        - Moderators should be able to immediately un-publish flagged tasks (after verifying the task details)
    - Ban users
        - Moderators should be able to ban users who created any of the flagged tasks; the banned users should never be allowed to login or create account with the same email address.

Design

The design of the tables is listed in appendix A with the ER diagram in appendix B. Two tables are required one for the user and the other for the tasks. Appendix C lists mock-ups of the webpages and are listed C1 to C13.

The processes are listed in appendix D and are listed P1 to P7. They follow closely the flow of the specification and page mock-ups in appendix C.

- P1 and C1 are the user login.
- P2 and C2 are register new user.
- P3 and C3 are create new task. P4 and C4 are display the task list with C5 mock-up displaying all the details of the task.
- P5 and C6 display the tasks claimed by a user, additional popups C7 allow a task to be marked as completed, cancelled or request the full document in which case an email template C8 is displayed and can be sent to the task creator.
- P6 and C9 display my tasks with popups C10 showing details of who claimed the task and C11 allowing completed tasks to be marked as satisfactory or un-satisfactory.
- P7 and C12 displays flagged tasks with popup C13 allowing the moderator to view details of the task, download the sample, ban the user and un-publish the task.

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 system for 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.

- JavaScript will be used to add functionally and effects to the webpages. It is and object orientated language. JavaScript has the ability to change HTML pages, styles and validate data, or produce an event on a click. It will be used as deemed fit throughout the project.

- JQuery is one of the JavaScript libraries. It is the most used JavaScript library and is open source. JQuery allows for changes to be made to HTML pages in a similar way to JavaScript. The key difference is that JQuery requires much shorter and simplified pieces of code. Changes such as colour or visibility of HTML elements can all be changed using JQuery. It will again be used as deemed fit throughout the project.

- MySQL is a relational database management system. It will be used as the primary means of creating a database for the proposed website.

- Github will be used as the main version control repository:

https://github.com/CS4014group18/web.

Appendix A Database Tables

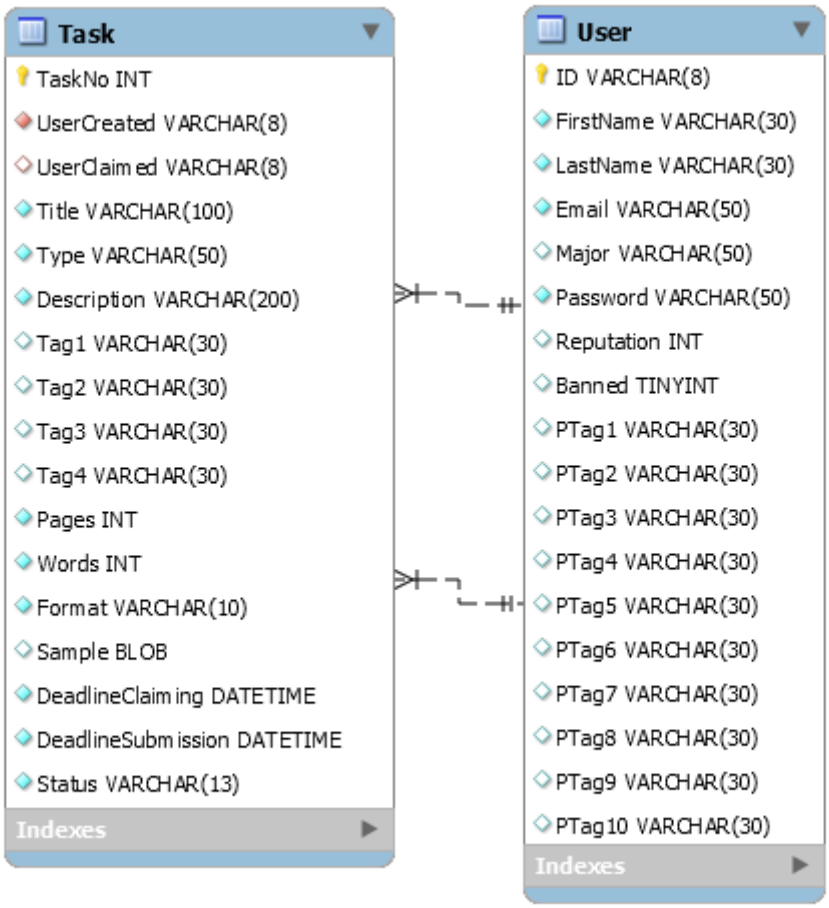
User Table

Field Name	Data Type	Comment
ID	VARCHAR(8)	Primary key
First Name	VARCHAR(30)	
Last Name	VARCHAR(30)	
Email	VARCHAR(50)	Unique
Major	VARCHAR(50)	
Password	VARCHAR(50)	
Reputation	INTEGER	
Banned	BOOLEAN	
PTag1	VARCHAR(20)	10 Preference tags
PTag2	VARCHAR(20)	
PTag3	VARCHAR(20)	
PTag4	VARCHAR(20)	
PTag5	VARCHAR(20)	
PTag6	VARCHAR(20)	
PTag7	VARCHAR(20)	
PTag8	VARCHAR(20)	
PTag9	VARCHAR(20)	
PTag10	VARCHAR(20)	

Task Table

Field Name	Data Type	Comment
TaskNo	INTEGER	Primary Key
UserCreated	VARCHAR(8)	Foreign Key: User.ID
UserClaimed	VARCHAR(8)	Foreign Key: User.ID
Title	VARCHAR(100)	
Type	VARCHAR(50)	
Description	VARCHAR(200)	
Tag1	VARCHAR(20)	
Tag2	VARCHAR(20)	
Tag3	VARCHAR(20)	
Tag4	VARCHAR(20)	
Pages	INTEGER	
Words	INTEGER	
Format	VARCHAR(10)	
Sample	BLOB or MEDIUMBLOB or TEXT or MEDIUMTEXT	
DeadlineClaiming	DATETIME	
DeadLineSubmission	DATETIME	
Status	VARCHAR(13) or ENUM	Values: PENDING, CLAIMED, UNCLAIMED, CANCELLED, INAPPROPRIATE, COMPLETED

Appendix B Entity Relationship Diagram



Appendix C Web Page Mock-ups

C1 User Login Page

Login Not Registered

ID

Password

C2 Register User Page

First Name

Last Name

ID

Email

Major

Password

Re-Enter Password

C3 Create Task Page

Title

Type

Description

Tag 1

Tag 2

Tag 3

Tag 4

No of Pages

No of Words

Source Format

Deadline Claiming

Deadline Completion

Choose Sample

Upload

C4 Task List Page

Task

Title 1

Title 2

Title 3

Title 4

Title 5

Title 6

Title 7

Title 8

Title 9

Title 10

C5 View Individual Task Page or Popup

Title

Tag 1

Tag 2

Tag 3

Tag 4

Description

Pages

Words

Download Sample

Claim

Inappropriate

C6 Claimed Task Page

Task

Title 1

Title 2h

Title 3

Title 4

Title 5

Title 6

Title 7

Title 8

Title 9

Title 10

C7 Email, Complete, Cancel Popup

Request Document

Completed

Cancel Task

C8 Email Template

To

Subject

Regarding Task: Task Title

Text

Please reply with your document

Send

C9 My Tasks

Task

Title	Status
Title 1	Completed
Title 2	Claimed
Title 3	Pending
Title 4	Unclaimed
Title 5	Cancelled

C10 Status Claimed Popup

First Name

Last Name

Email

C11 Rate Task Popup

Rate Task

Satisfactory ☐

Un-Satisfactory ☐

Submit

C12 Flagged Tasks

Flagged Tasks

Title 1

Title 2

Title 3

Title 4

Title 5

Title 6

Title 7

Title 8

Title 9

Title 10

C13 View Flagged Task Page or Popup

Title

Tag 1

Tag 2

Tag 3

Tag 4

Description

Pages

Words

Download Sample

Un-Publish

Ban User

Appendix D Processes

Process Number	P1
Process Title	Login User
Brief Description	Login user
Inputs	User ID and password
Detailed Description	Logins user, verifies password is correct
Output	User

Process Number	P2
Process Title	Register User
Brief Description	Accepts registration requests from users. If supplied information is correct or rejects request if information supplied is incorrect.
Inputs	User data
Detailed Description	Users wishing to register fill out a form on the homepage. If the correct input is received for each field the user will have successfully registered. If incorrect input is supplied the user will be asked to resubmit their registration with the incorrect fields highlighted.
Output	Update database user table

Process Number	P3
Process Title	Create Task
Brief Description	User creates a new task
Inputs	User, Task data
Detailed Description	Allows user to enter data on a task and upload sample pages
Output	Update database task table

Process Number	P4
Process Title	Display Task List
Brief Description	Display a list of tasks that can be claimed.
Inputs	User, Database task table
Detailed Description	Allows user to download a sample, claim the task or flag as inappropriate.
Output	Display task list and details of individual tasks, update task table status, download sample

Process Number	P5
Process Title	Display Claimed Task List
Brief Description	Display a list of tasks that the login user has claimed
Inputs	User, Database task table
Detailed Description	Display a list of tasks that the login user has claimed, Allows e user to mark task as completed, cancelled or to request the document by email.
Output	Display task list, Update task table status, Email template

Process Number	P6
Process Title	Display My Tasks
Brief Description	Displays my tasks
Inputs	User, Database task table
Detailed Description	Displays a list of my tasks and their status. For claimed tasks the name and email address of claimant is displayed. A rate task popup allows completed tasks to be marked as satisfactory or un-satisfactory.
Output	Display task list, Display database name and email of user who claimed task, Database user table reputation is updated

Process Number	P7
Process Title	Flagged Tasks
Brief Description	Display flagged tasks
Inputs	Database user table users with reputation > 40, Database task table entries with status inappropriate and user that created the task
Detailed Description	Allows moderators to view tasks flagged as inappropriate and un-publish them and ban the user who created the task.
Output	Display flagged task list, Database update user table banned flag, delete inappropriate task from database task