Group 40

*Studioruum*

Mission Statement

We aim to produce a comprehensive learning application that can help with the study of various different topics, both academic and personal (eg. Learning university modules or foreign languages). Our system will revolve around scholars (the general user) and educators (users with more capabilities to share material with groups) who can both who can both create flashcards, notes, dictionaries and other types of content to aid in the learning process, which can be reviewed in various different ways.

Mission objectives

1. Application must be used via a GUI
2. **Users** must be able to create resources like flashcards, revision notes, quizzes, dictionaries.
   1. *Application must save resources in the database*
3. **Users** must be able to log into the system. Authentication system will use username and password.
4. **Scholars** must be able to ask questions to a public domain which can be answered by educators.
   1. ***Scholar*** *must be able to take part in quizzes (Tests) published onto the public domain.*
   2. ***Scholar*** *must be able to receive feedback on their attempts to these quizzes*
5. **Educators** must be able to upload resources onto a public domain
   1. *A User must be able to search and access resources uploaded by other users.*
   2. *A Users must be able to rate resources*
   3. *Application must order search results from highest to lowest rated.*
6. **Educators** must be able to create a class
   1. An **Educator** must be able to invite users to their class
7. **Educators** must be able to distribute quizzes to users in their class.
   1. An ***Educator*** *must be able to review, mark and provide feedback for the completed quizzes,*
   2. *Application must save quizzes, and their results to the database*
8. **Educators** can share resources like flashcards, notes and posters to all Scholars in their class.

User Views

Our project aims to aid particular users (mainly academic related) in revision, whether that being creating flashcards and documents or to help in learning a new language which our app supports with the help of dictionaries for language to language comparisons. The application has two different types of users: **Scholar and Educator.** An essential feature of our project is a **public domain** where Educators can share their own notes, release quizzes and answer Scholars questions which were asked to the public domain. The Educators notes and quizzes can be accessed/completed by Scholars. Another feature of our project is a class service. A class can be created by an Educator where they can invite Scholars to join their “classroom”, which acts as a private domain where the Scholar to Educator relationship is much more closely related. This service makes it easier for Educators to send Scholars notes/questions/revision advice and also allows Educators to give Scholars personal feedback/marks/reviews on their quiz attempts.

Scholar

Scholar is the largest and most regular user group of our application.

* Create flashcards
* Create notes (documents)
* Create dictionaries
  + These can all be saved and edited at a later date.
  + These resources must be saved to the database
* Ask questions to a public domain
  + However, Scholars cannot publish their own notes to the public domain. But they can access already shared notes on the domain.
* Scholars will log into the system using a username and password

Once invited by an Educator, Scholars can

* Join Educators classes
* Answer Educators quizzes
* Receive feedback on their attempts.

Educators

* Inherits all aspects/attributes of a Scholar

These are closely related to Teachers within schools, but with a little bit more implementation. They act as somewhat administrators to their personal classrooms which contain Scholars.

* Create Classes
* Invite Scholars to their classroom
  + Publish quizzes and notes to these classrooms
  + Give Feedback on the Scholar attempts
* Answer Scholar questions in the public domain
* Publish public notes to the domain
* Educators will log into the system using a username and password

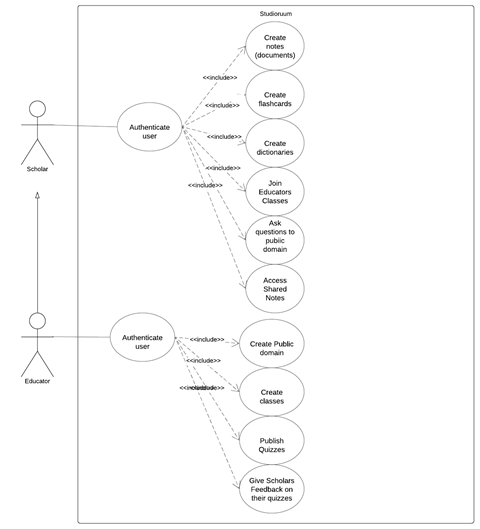
Transaction Requirements

|  |  |
| --- | --- |
| **Data Entered** | **User Views Involved** |
| Create User/Account | Scholar + Educator |
| Make notes | Scholar |
| Create quizzes | Scholar + Educator |
| Create comment | Scholar + Educator |
| Make Class notes | Educator |
| Create Class | Educator |
| Join Class | Scholar |

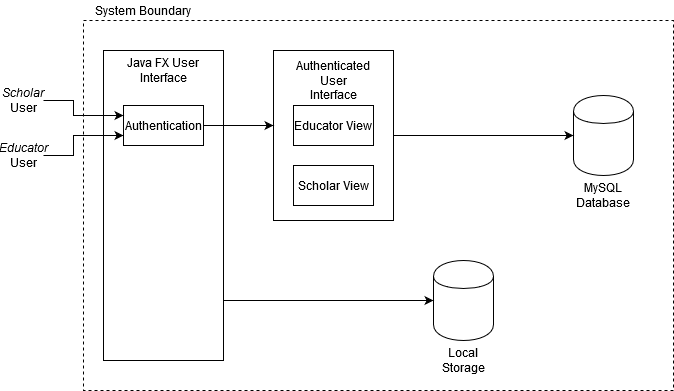
|  |  |
| --- | --- |
| **Data deleted or updated** | **User views involved** |
| Notes | Scholar |
| Quizzes | Scholar + Educator |
| Comments | Scholar + Educator |
| Class notes | Educator |
| Class | Educator |

|  |  |
| --- | --- |
| **Data Queried** | **User views involved** |
| Notes in personal database | Scholar + Educator |
| Notes in class | Scholar + Educator |
| Link to join class | Scholar |
| View Quiz | Scholar + Educator |
| List of public notes | Scholar + Educator |
| List of public quizzes | Scholar + Educator |
| Search for notes/quiz | Scholar + Educator |
| Load comments | Scholar + Educator |

Use-Case Diagram



System Boundary Diagram



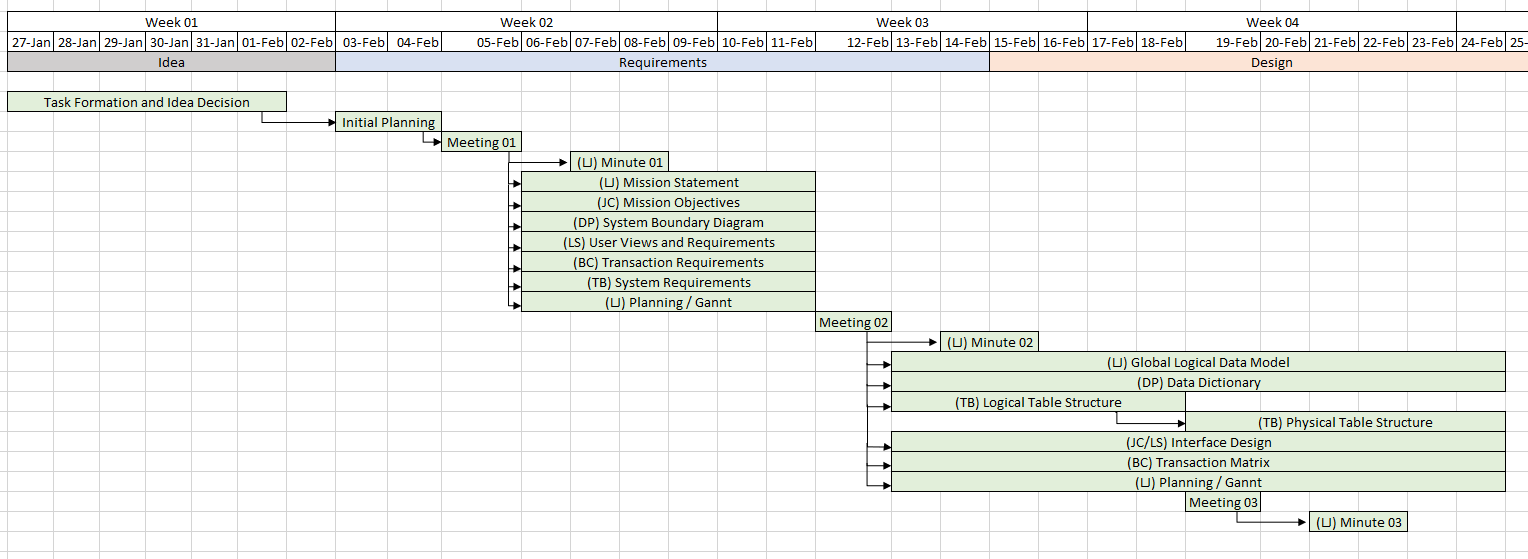
System Requirements

* The system must allow for multiple users concurrently;
* The system must allow for the user to make notes in document formats such as pdf and txt;
* The system must allow for the user to share the notes they have made through email and social media, as well as being able to export them as a file;
* The system must support the following character sets: UNICODE, UTF-8;
* The system must allow for the use of different fonts;
* The system must allow for the formatting of text and paragraphs to make notes easier to read;
* User account should be secured using usernames and passwords;
* No two users may have the same username, and passwords must be at least 8 characters in length containing at least 1 number;
* Storage of passwords as hashes;
* In the event of a forgotten password an email link will be sent to the user. The user can reset their password through this link;
* The users will search the database multiple times a week for their records to check for notes they need to revise or for the educator to check which students have completed which tasks;
* The system will only require network and database access when retrieving files from other users;
* Rate of growth – the database must have the capability to expand for more users and documents to be stored;
* The system and its data must comply with the UK data protection act 1998.

## Planning

## Gannt Chart

We have created an initial Gannt Chart to show some rough estimations of how we expect the product to be conducted. Whilst earlier sections have more detail (such as task allocations), the later sections are currently estimating the time needed for general phases of development and don’t have individual tasks underneath them. Throughout the development of the project, we will use the tools shown below to add to and refine upon this Gannt Chart and the planning of the project as a whole.



## Meetings

We aim to meet every week (excluding the first week) for an hour review session to check the quality of work produced and allocate tasks to be completed over the next week(s). Currently, we have had two meetings (see the minutes for more detail), with the general topics discussed being:

1. Allocating Tasks for the Requirements and General Planning
2. Allocating Tasks for the Design and Finalising the Requirements Document

## Other Tools

As well as the above methods, we have been using a *Discord Server* that allows us to communicate quickly with each other and share files for feedback. The main use thus far has been to schedule meetings, brainstorm ideas, check requirements of the system and mention any important information needed for the project.

As with many larger software development projects, we are using a *GitHub Repository* to aid in source control, and host all of the project files. All planning updates and important information are listed on this page to be easily accessible, and all work completed by group members are quickly available in a centralised place.

## Planning Choices

Though the schedule given in the lectures for the project completion is to be mostly adhered to, we have decided that it would be beneficial to try and complete the design in *two or three weeks* as opposed to *four* to try and give ourselves more time for implementation. Combining that with the time over the break should allow us to have more than enough time as is needed for the coding needed to implement the project in its entirety.

# Signatures

In the Order - Lloyd Jones, Daniel Ponturo, Tyler Bunsie, Bobby-Chase Davies, Lachlan Smith, Jordan Clewlow.

