**Open Alexandria**

Team Members: Bor Shiuan Chen, Brian Chan, Hongda Zeng, Xu Han

**Problem Statement**

Many file sharing and education sites cost money and are not open source. This often results in locked features and slower development. We want to provide a free, open-source, secure, in-depth education sharing platform where users can help each other in academia or even contribute code to the site.

**Background Information**

Almost every college student have a class they will search up for help with their homework and project. And often not students will find the information they need online, but it comes at a cost. Websites like Course Hero or Chegg require a premium or subscription and often provide you with a free preview of the first page or the question a particular student had. Most students are not willing to pay a monthly subscription just for a few documents. Open Alexandria will be a solution to that premium and allow users to share documents and have discussions without charging any user to access the database.

**Environment**

We will be developing our system using an HTML5 client web app front end, node js server and postgresql database. A RESTful api will also handle network communications between the client and server.

**Requirements**

**Functional:**

|  |  |  |
| --- | --- | --- |
| ID | Description | Hours |
| 1 | As a user, I want to post questions about a course | 30 |
| 2 | As a user, I want to answer questions about a course | 30 |
| 3 | As a user, I want to add a course | 30 |
| 4 | As a user, I want to upvote/downvote questions/answers | 15 |
| 5 | As a user, I want to upload documents | 40 |
| 6 | As a user, I want to access shared documents | 20 |
| 7 | As a user, I want to delete my uploaded documents | 15 |
| 8 | As a user, I want to review documents uploaded | 10 |
| 9 | As a user, I want to be able to contact the developer to report any issues | 10 |
| 10 | As a user, I want to automatically retrieve a list of classes as I type the course name | 20 |
| 11 | As a user, I want to register an account by entering basic user information | 10 |
| 12 | As a user, I want to log into my account | 10 |
| 13 | As a user, I want to be able to edit basic user information | 10 |
| 14 | As an administrator, I want to block certain users | 20 |
| 15 | As an administrator, I want to delete undesired documents | 10 |
| 16 | As an administrator, I want to delete undesired groups | 15 |
| 17 | As an administrator, I want to delete undesired questions | 15 |
| 18 | As a user, I want to add tags to a document | 5 |
| 19 | As a user, I want to add ratings to a document | 5 |
| 20 | As a user, I want to add comments to a document | 5 |

**Non-functional:**

1. As developers, we want our application to be simple, straightforward, and easy to use with little to no introduction.
2. As developers, we want our application to utilize a clean and professional user interface.
3. As developers, we want our application to run as lightly as possible reducing system resource costs.
4. As developers, we want our application to have no bugs if possible.
5. As users, we want our application to respond in 100ms or less without sacrificing features of the app.
6. As developers, we must be able to accommodate at least 10,000 users.
7. As developers, we want to store as little data on the client device as possible.
8. As developers, we want our database system to be scalable.
9. As a developer, I want to minimize click depth and have no more than 5 clicks for any function.
10. As a user, I want to always be able to access the app.
11. As a developer, I want to reduce server downtime as much as possible.
12. As a developer, I want the API to be as simple as possible.
13. As a developer, I want the database to be as compact as possible.
14. As a developer, I want there to be as few database as possible.

**Use Cases**

|  |  |
| --- | --- |
| **Case** | 1. As a user, I want to post questions |
| **Action** | **System Response** |
| 1. Search a course or topic in navigation bar |  |
|  | 2. Display courses/topics thumbnail if found |
| 3. Select course/topic |  |
|  | 4. Display all course/topic discussion outline and files |
| 5. Click on “new question” button link |  |
|  | 6. Question text box appears |
| 7. User fills out question in text and clicks “post” button |  |
|  | 8. Posted question appears |
|  | 9. Page and logs are updated |

|  |  |
| --- | --- |
| **Case** | 2. As a user, I want to reply to questions |
| **Action** | **System Response** |
| 1. Search a course or topic in navigation bar |  |
|  | 2. Display courses/topics thumbnail if found |
| 3. Select course/topic |  |
|  | 4. Display all course/topic discussion outline and files |
| 5. User clicks on current question links of course/topic |  |
|  | 6. Full discussion page appears |
| 7. User clicks “reply” button” |  |
|  | 8. Reply box appears |
| 9. User fills out his/her reply and clicks “submit” |  |
|  | 10. Text box appears |
|  | 11. Page and logs are updated |

|  |  |
| --- | --- |
| **Case** | 3. As a user, I want to add a course |
| **Action** | **System Response** |
| 1. Search a course or topic in navigation bar |  |
|  | 2. Course not found, so display “create course” button |
| 3. Click “create course” button |  |
|  | 4. Display new course page with options to create questions and upload documents |
|  | 5. Page and logs are updated |

|  |  |
| --- | --- |
| **Case** | 4. As a user, I want to upvote/downvote questions/answers |
| **Action** | **System Response** |
| 1. Search a course or topic in navigation bar |  |
|  | 2. Display courses/topics thumbnail if found |
| 3. Select course/topic |  |
|  | 4. Display all course/topic discussion outline and files |
| 5. Click on question in outline |  |
|  | 6. Show question and its answers |
| 7. User clicks up or down arrow on the question or answers |  |
|  | 8. Page and logs are updated |

|  |  |
| --- | --- |
| **Case** | 5. As a user, I want to upload documents |
| **Action** | **System Response** |
| 1. Search a course or topic in navigation bar |  |
|  | 2. Display courses/topics thumbnail if found |
| 3. Select course/topic |  |
|  | 4. Display all course/topic discussion outline and files |
| 5. User clicks on documents for the selected course |  |
|  | 6. Document page appears |
| 7. User clicks upload to upload document |  |
|  | 8. Upload window appears and user can select document to upload from computer |

|  |  |
| --- | --- |
| **Case** | 6. As a user, I want to access shared documents |
| **Action** | **System Response** |
| 1. Search a course or topic in navigation bar |  |
|  | 2. Display courses/topics thumbnail if found |
| 3. Select course/topic |  |
|  | 4. Display all course/topic discussion outline and files |
| 5. User clicks on documents for the selected course |  |
|  | 6. Document page appears |
| 7. User can choose which document to access |  |
|  | 8. Selected document will open up |

|  |  |
| --- | --- |
| **Case** | 7. As a user, I want to delete my uploaded documents |
| **Action** | **System Response** |
| 1. Select view account detail |  |
|  | 2. Display account detail for the current user logged in |
| 3. Select my uploaded documents |  |
|  | 4. Display all documents uploaded by the user |
| 5. User clicks delete for the particular document |  |
|  | 6. Alert for document successfully or failed to be deleted |

|  |  |
| --- | --- |
| **Case** | 8. As a user, I want to review documents uploaded |
| **Action** | **System Response** |
| 1. Search a course or topic in navigation bar |  |
|  | 2. Display courses/topics thumbnail if found |
| 3. Select course/topic |  |
|  | 4. Display all course/topic discussion outline and files |
| 5. User clicks on documents for the selected course |  |
|  | 6. Document page appears |
| 7. User clicks upload to upload document |  |
|  | 8. Upload window appears and user can select document to upload from computer |
|  | 9. Review documents page loads |
| 10. User can accept or reject document uploaded |  |
|  | 11. Proceeds to all documents page if accepted and the new document will appear in the list |
|  | 12. Deletes the entry and proceeds to all documents page if rejected, no changes will be made |

|  |  |
| --- | --- |
| **Case** | 9. As a user, I want to be able to contact the developer to report any issues |
| **Action** | **System Response** |
| 1. User selects Contact Us | 2. Systems displays contact information with spaces to put a quick comment` |
| 2. User press submit | 4. System stores information for administrator to view later |

|  |  |
| --- | --- |
| **Case** | 10. As a user, I want to automatically retrieve a list of classes as I type the course name |
| **Action** | **System Response** |
| 1. User starts typing in class name | 2. Systems contacts Purdue Class Listing to pull up a list of classes |
|  | 3. System filters class listing based on what the user has typed |
| 3. User selects correct class name | 4. System changes text field to display class name |

|  |  |
| --- | --- |
| **Case** | 11. As a user, I want to register an account by entering basic user information |
| **Action** | **System Response** |
| 1. User selects Create Account | 2. System loads up new page giving fields for basic account information |
| 3. User enter information |  |
| 4. User press submit | 5. System creates new entry in database for user |

|  |  |
| --- | --- |
| **Case** | 12. As a user, I want to log into my account |
| **Action** | **System Response** |
| 1. User selects Login | 2. Systems displays new page with username and password field |
| 2. User enters login information |  |
| 3. User press login | 4. System authenticates against database to confirm the user exist |

|  |  |
| --- | --- |
| **Case** | 13. As a user, I want to be able to edit basic user information |
| **Action** | **System Response** |
| 1. User selects Edit Profile | 2. Systems displays new page with field to edit name, email, phone, password, and etc |
| 2. User enters new information |  |
| 3. User press submit | 4. System updates user entry with new information |

|  |  |
| --- | --- |
| **Case** | 14. As an administrator, I want to block certain users |
| **Action** | **System Response** |
| 1. Admin selects User List | 2. Systems displays new page with a list of users |
| 3. Admin press delete next to a user | 4. System block users from actions on the site |

|  |  |
| --- | --- |
| **Case** | 15. As an administrator, I want to delete undesired documents |
| **Action** | **System Response** |
| 1. Admin selects Document List | 2. Systems displays new page with a list of documents |
| 3. Admin press delete next to a document | 4. System remove user access to document |

|  |  |
| --- | --- |
| **Case** | 16. As an administrator, I want to delete undesired groups |
| **Action** | **System Response** |
| 1. Admin selects Group List | 2. Systems displays new page with a list of groups |
| 3. Admin press delete next to a group | 4. System remove user access to group |

|  |  |
| --- | --- |
| **Case** | 17. As an administrator, I want to delete undesired questions |
| **Action** | **System Response** |
| 1. Admin selects Group List | 2. Systems displays new page with a list of groups |
| 3. Admin selects Question List next to group name | 4. System displays a new page with a list of questions |
| 5. Admin press delete next to a question | 6. System remove user access to question |

|  |  |
| --- | --- |
| **Case** | 18. As a user, I want to add tags to a document |
| **Action** | **System Response** |
| 1. User clicks “+” button next to tag icon on document page | 2. Systems displays new tag dialog box |
| 3. User enters desired tag names | 4. System displays document tag names on page |

|  |  |
| --- | --- |
| **Case** | 19. As a user, I want to add ratings to a document |
| **Action** | **System Response** |
| 1. User clicks hovers over stars bar of document and selects the amount of stars to give a document | 2. Systems displays updated rating for document. |
| 5. Admin press delete next to a question | 6. System remove user access to question |

|  |  |
| --- | --- |
| **Case** | 20. As a user, I want to add comments to a document |
| **Action** | **System Response** |
| 1. User clicks “comment” button on document page | 2. Systems displays new comment dialog box |
| 3. User enters desired comment | 4. System displays document comment at top of page |