

Sprint 2 Planning Document - Open Alexandria

Team 4   
Xu Han, Brian Chan, Andy Chen, Hongda Zeng

# Sprint Overview:

In this sprint we want to get the rest of core functions completed: flashcard and document. In addition to core functionality, we investigating into the user experience or the flow of the website. Currently the website works as alpha with no consideration towards usability. In this sprint we are hoping to bring that ease of use to the user. We will also be using Spring Break to our advantage and get more done.

**Scrum Master:** Brian Chan

**Meeting Schedule:** Tuesday@18:00, Thursday@18:00, Saturday@14:00

**Risks/Challenges:** The team will be using a broad array of different web API's to implement a large chunk of the functionality being added in this sprint. This means a large amount of time could be eaten up just becoming familiar with the technologies. It will be important to manage

time well. We will also be attempting to handle files, which we have not done before.

[**Sprint Overview:**](#_clarkm3ifpid) **1**

[**Hours Allocated Per Member:**](#_q14odta5b5kj) **2**

[**User Story**](#_h67a54mcy5al) **3**

[5: As a user, I want to upload documents](#_lrcqw13yw8ri) 3

[6: As a user, I want to access shared documents](#_gf5lp5oxn9yw) 3

[7: As a user I want to delete my uploaded documents](#_yxehxfki1hso) 4

[13: As a user, I want to be able to edit basic user information](#_23j7prbig8fg) 4

[15: As an administrator, I want to delete undesired documents](#_wi3vfsyeohly) 5

[18: As a user, I want to add tags to a document](#_grf1xfnf2evu) 5

[19: As a user, I want to add ratings to a document](#_64rle6ymsvlv) 6

[20: As a user, I want to add comments to a document](#_kx7ey2kpc6qz) 6

[21: As a user, I want the logo to redirect to the home page](#_n56wbw1gve) 7

[22: As a user, I want login to change to profile when I logged in.](#_j49vike72nqs) 7

[23: Replace static header/footer with dynamic header/footer](#_7qibpcjmc6jx) 7

[**Remaining User Stories**](#_6xzu1louju49) **8**

# Hours Allocated Per Member:

|  |  |
| --- | --- |
| Member | Hours |
| Brian Chan | 90 |
| Andy Chen | 90 |
| Xu Han | 80 |
| Hongda Zeng | 90 |
| **Total** | 350 |

# User Story

## 5: As a user, I want to upload documents

|  |  |  |  |
| --- | --- | --- | --- |
|  | Task Description | Estimated | Owner |
| 1 | Create database table to store document information | 5 | Hongda |
| 2 | Create API routes to upload documents | 10 | Andy |
| 3 | Front-end create document upload dialog | 20 | Jack |
| 4 | Front-end create document preview | 20 | Brian |

**Acceptance Criteria:**

Given a user logs into his/her account, when the user goes to a course page, then document upload will be enabled. Given a user, when he or she press upload document, then a window will appear with options to select documents from the computer. Given the user clicks the submit button, when on the upload document page, then the RESTful query is submitted to the NodeJS server. Given the NodeJS server, when a document is received, the new document will be entered into the database.

## 6: As a user, I want to access shared documents

|  |  |  |  |
| --- | --- | --- | --- |
|  | Task Description | Estimated | Owner |
| 1 | Create API route to get documents | 5 | Andy |
| 2 | Generate preview of documents | 15 | Andy |
| 3 | Create front-end page to show documents | 20 | Brian |

**Acceptance Criteria:**

Given a user is logged into his/her account or not, when the user goes to a course page, then document preview is available. Given a user, when he or she presses the document, then a window will appear with the full document preview. Given the NodeJS server, when a document is requested, the new document will be returned from the database.

## 

## 7: As a user I want to delete my uploaded documents

|  |  |  |  |
| --- | --- | --- | --- |
|  | Task Description | Estimated | Owner |
| 1 | Create API router to disable documents | 10 | Andy |
| 2 | Create button to disable a document | 5 | Jack |

**Acceptance Criteria:**

After a user logs into his/her account, the user will be able to see all of their uploaded document and will have the option to delete their uploaded document. Given the option to see all documents, when the user clicks to see all documents, then all the document will be listed. Given the user clicks delete document on a specific document, then the RESTful query is submitted to the NodeJS server. Given the NodeJS server, when a delete is received, then the document will be deleted in the database and not displayed on the website.

## 13: As a user, I want to be able to edit basic user information

|  |  |  |  |
| --- | --- | --- | --- |
|  | Task Description | Estimated | Owner |
| 1. | Create API route to modify user information | 10 | Andy |
| 2. | Create edit profile page with route | 15 | Jack |

**Acceptance Criteria:**

After a user logs into his/her account, the user will be able to see all of their information and be able to edit their information. Given the option to see user information, when the user clicks to see their information, then their information will be displayed on their profile. Given the option to edit user information, when the user clicks to edit their information, then they will be able to edit their information and submit to the NodeJS server. Given the NodeJS server, when user information is changed, then the user information will be changed in the database.

## 15: As an administrator, I want to delete undesired documents

|  |  |  |  |
| --- | --- | --- | --- |
|  | Task Description | Estimated | Owner |
| 1. | Create admin route to disable any document | 10 | Andy |
| 2. | Add all document to admin page | 10 | Jack |
| 3. | Add button to each document to enable or disable | 10 | Jack |

**Acceptance Criteria:**

Given a document, when the admin wants to see all documents, then the admin will have the ability to see all the uploaded documents. Given an admin that wants to disable documents, when the admin press the disable button, then the RESTful query is submitted to the NodeJS server. Given the NodeJS server, when a disable is received, then the document will be disabled in the database and not displayed on the website.

## 18: As a user, I want to add tags to a document

|  |  |  |  |
| --- | --- | --- | --- |
|  | Task Description | Estimated | Owner |
| 1. | Update database to support tags | 5 | Hongda |
| 2. | Create API route to add tags to document | 10 | Andy |
| 3. | Create dialog to add tags to documents | 10 | Jack |

**Acceptance Criteria:**

After a user logs into his/her account, the user will be able to see all of their uploaded document and will have the option to add/remove tags to an uploaded document. When the user clicks submit, the RESTful query is submitted to the NodeJS server. Given the NodeJS server, when a submit is received, the document’s tags will be updated in the database and displayed on the website.

## 

## 19: As a user, I want to add ratings to a document

|  |  |  |  |
| --- | --- | --- | --- |
|  | Task Description | Estimated | Owner |
| 1 | Update database to support rating | 5 | Hongda |
| 2 | Create API route to add likes and dislikes to a document | 10 | Andy |
| 3 | Create button to like and dislike documents | 20 | Brian |

**Acceptance Criteria:**

Given a document, when other users view a document, then there would be a rating options. Given a rating, when a rating is retrieved for a document, then it would show the average rating given by the other users. GIven a document, when a user rates a document, then the server will add the rating into the server.

## 20: As a user, I want to add comments to a document

|  |  |  |  |
| --- | --- | --- | --- |
|  | Task Description | Estimated | Owner |
| 1 | Update database to support comments | 10 | Hongda |
| 2 | Create API route to add comments to a document | 10 | Andy |
| 3 | Create dialog to add comments to a document | 20 | Brian |
| 4 | Create comment bar below documents | 10 | Jack |

**Acceptance Criteria:**

Given a document, when the user finish uploading, then the user will have the ability to add comments to below the documents. Given a user inputed comments, when the user press submit, the server will receive the comment. Given a user comment when the server receives the comment, then it stores it into the database. Given a documents, when other users access the document, then the server will retrieve comments from the database.

## 21: As a user, I want the logo to redirect to the home page

|  |  |  |  |
| --- | --- | --- | --- |
|  | Task Description | Estimated | Owner |
| 1 | Update logo to redirect to home page | 10 | Hongda |
| 2 | Verify all logo redirects to home page | 10 | Hongda |

**Acceptance Criteria:**

Given an user, when the user want to return to the home page, then the user can click the logo. Given an user who wants to return to the home page, when the use click the logo, then the homepage will load. Given an user who is navigating the website, when the user arrive at a page, then the user will see a logo.

## 22: As a user, I want login to change to profile when I logged in.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Task Description | Estimated | Owner |
| 1 | Replace login button with profile button | 10 | Hongda |
| 2 | Remove profile button on logout | 10 | Hongda |

**Acceptance Criteria:**

Given an user who logged in, when the login is completed, then the user will see a button linking to the profile. Given a user who is navigating the website, when the user arrive at any page, then a button will display in the header that can lead them to their profile page. Given an user who wants to logout, when the user is logged out, then the profile button will not display.

## 23: Replace static header/footer with dynamic header/footer

|  |  |  |  |
| --- | --- | --- | --- |
|  | Task Description | Estimated | Owner |
| 1 | Investigate how to create dynamic header/footer | 15 | Hongda |
| 2 | Implement dynamic header/footer | 10 | Hongda, Brian |

**Acceptance Criteria:**

Given an user who arrived at the site, when the user load the homepage, then the user will see a header and a footer. Given an user who is navigating the site, when the user arrives at a new page, then the user will see the same header and footer for all pages. Given a user who want to use functionalities on the header and footer, when the user click on a function, then the requested page will load.

# Remaining User Stories

User story #4: As a user, I want to upvote/downvote questions/answers.

User story #8: As a user, I want to review documents uploaded.

User story #9: As a user, I want to be able to contact the developer to report any issues.

User story #24: As a user, I want to be able to search documents in the search bar.

User story #25: As a user, I want to be able to by University in the search bar