

**BSc (Hons) Computing**

**COMP1640**

**ENTERPRISE WEB SOFTWARE DEVELOPMENT**

|  |  |
| --- | --- |
| **Four Stars** | **UOG ID** |
| **KAJENDRA NAATH RAJAN** | **001192656** |
| **HO KAI LIN** | **001192641** |
| **KEERTHIGAI DEVI THEVENTHIRAN** | **001192663** |
| **LEE MIN QI** | **001192652** |
| **BRIAN EE CHI CHIN** | **001192627** |

Table of Contents

[**1.0 Introduction** 3](#_Toc133014162)

[**1.1 Assumptions** 3](#_Toc133014163)

[**1.2 Software Requirements** 6](#_Toc133014164)

[**1.2.1 Windows & MacOS** 6](#_Toc133014165)

[**1.2.2 Xampp Server** 6](#_Toc133014166)

[**1.2.3 PHP (Language)** 6](#_Toc133014167)

[**1.2.4 MYSQL (phpMyAdmin)** 7](#_Toc133014168)

[**1.3 Installation Steps** 7](#_Toc133014169)

[**2.0 Artifacts** 8](#_Toc133014170)

[**2.1 Credentials & Links** 8](#_Toc133014171)

[**2.2 Group members & roles** 8](#_Toc133014172)

[**2.3 System Test Accounts Credentials** 8](#_Toc133014173)

[**2.4 Product Backlog** 9](#_Toc133014174)

[**2.5 Sprint 1 Backlog** 17](#_Toc133014175)

[**2.6 Sprint 2 Backlog** 21](#_Toc133014176)

[**2.7 Sprint Backlog 3** 29](#_Toc133014177)

[**2.8 Sprint Backlog 4** 30](#_Toc133014178)

[**2.9 Burndown Charts** 31](#_Toc133014179)

[**2.10 Meeting Minutes** 34](#_Toc133014180)

[**2.11 UML Case Diagram** 39](#_Toc133014181)

[**2.13 Entity Relationship Diagram (ERD)** 40](#_Toc133014182)

[**2.14 Data Dictionary** 40](#_Toc133014183)

[**2.15 Test Case results** 44](#_Toc133014184)

[**2.16 Test Plan** 54](#_Toc133014185)

[**2.17 Test Logs** 57](#_Toc133014186)

[**3.0 Screenshots and Explanation** 67](#_Toc133014187)

[**4.0 Conclusion** 76](#_Toc133014188)

# 1.0 Introduction

The primary scope of this coursework focuses on how agile methodology is applied throughout the completion of this web application. In this project, agile methodology plays a vital role by presenting the workloads that should be done in each sprint. As of for this project, we had 4 sprints in total and each sprint had its respective tasks. In order to keep track of all these tasks, we had a management tool called Trello to keep us updated regarding the upcoming tasks and the completed tasks of this project. Besides the methodology and workloads that has been performed, it is primarily important to explain what is this system all about. This system is built for universities to receive feedbacks from their staffs in order to improve their university services. This system contains the primary facilities like login and registration for admin, staffs, QA managers and coordinators. Each roles have its own responsibilities and can perform their respective tasks in the system. There are several assumptions to be made based on the functionalities that have been built in this system.

## **1.1 Assumptions**

* When a register process occurs, it is assumed the registered data of the user will be stored in the database and will allow user to login with the same credentials. When the username and password is entered, the login screen will then navigate the user to the forum page. This assumption is taken to complete the registration and login processes.
* When the login process is passed, it is assumed the system will show the forum screen to the user according to the user types. There are 4 primary users for this system. The admin, staff, QA manager and coordinator. It is assumed that each of these users will be seeing the same screens with different functions. This assumption is taken to complete various login process in the system.
* When the main forum screen is seen, it is assumed the system will show certain navigation buttons like top rated ideas, most viewed ideas, recent ideas and recent comments. Each of these buttons are assumed to show the ideas that are posted which belongs to the types of buttons when they are clicked. This assumption is taken to complete forum screen functions.
* When the forum screen is seen by all the users, it is assumed that the screen will the username and will present some buttons which are fully functionable. Buttons like admin area, QA management panel will navigate the user to a different screen. Whereas, the report button will present the overall summary of the posted ideas in terms of numbers and percentage. The sign out button is assumed to sign the user out of the system and would need the user to login again during the next system usage. This assumption is taken to complete the button functions in the forum screen.
* The idea display page is assumed to show the ideas with like and dislike buttons for each idea. This screen is assumed to be paginated and can be scrolled down to see more ideas if there are more than 10 or 20 ideas. This screen is assumed to show the title name and the description posted by the user along with the date and time for their reference. This assumption is taken to complete the like, dislike and pagination functions.
* This system is assumed to store each idea posted in the database and can be seen by the admin at any time. It is also assumed that the admin can edit or delete and ideas from the server and manage the data that is shown in the user dashboard. It is also assumed that the admin can login to the system to oversee the process. This assumption is taken to complete the database process performed by the admin.
* The select forum page is assumed to show any forums that have been inserted. User can click the enter forum button and it’s assumed that the system will bring the user to the next screen where ideas are posted and seen. This screen is also assumed to have all the navigation buttons which the user can click. This assumption is taken to complete the forum selection process. This assumption is taken to see the number of forums entered and complete the click function of each forum.
* The admin panel screen is assumed to show two primary button functions like manage users and manage forum. These buttons are assumed to bring the admin to the manage users screen to oversee the user’s performance and the forum screen as well. Consisting of several similar functions like other screen, this screen is assumed to show various buttons in the top navigation bar. This assumption is taken to see the two buttons’ functions and to complete the screen pop up once the button is clicked.
* The category display screen is assumed to show all subjects that are inserted in the database. It is assumed the admin or QA manager and coordinator can edit or delete the subject and it will present the status. This assumption is taken to see how the edit and delete functions works when it is being clicked.
* This screen is also assumed to show a button called add categories which allows users to add new category to the list. This function is assumed to ask the subject name, description title, closure date and final closure date from the user. This assumption is taken to see the add category button’s function when it is being clicked.
* The submit idea screen is assumed to show the posted ideas of other staffs and some buttons which allows user to post theirs as well. This screen is assumed to show the title of the topic that is chosen. This assumption is taken to see the if the screen shows the correct title when the button is clicked from the forum screen.
* This screen is assumed to have the submit idea button which allows users to input the idea title, description, category selection, file upload and terms and conditions agreement selection. This assumption is taken to complete the information input for the idea submission by the staff.
* This same screen is assumed to have the submit anonymously checkbox function which allows user to submit their ideas anonymously without showing their identity to other staffs. It is assumed the selection will present a title called anonymous when an idea is posted. This assumption is taken to complete the anonymous comment postings by the staff.
* The last screen of this system is assumed to show the overall summary of all ideas that have been posted by staffs, QA managers and coordinators. This screen is assumed to show the forum title and the number of ideas that were posted in each title post. It is also assumed the screen has six different button functions. The ideas per department button is assumed to show ideas posted in each department. The percentage button will present the overall percentage calculated from the ideas posted. The ideas without comment button are assumed to show posts without comments. The anonymous idea button is assumed to show anonymous users. the anonymous comments are assumed to show users who have posted anonymous comments in any posts.

## **1.2 Software Requirements**

### **1.2.1 Windows & MacOS**

In order to use this website, the end user can either use windows operating system or Mac operating system. Since this software can be installed in both platforms, it would be flexible for both windows and mac users to access this system. Besides the operating system requirements, the user should have a minimum knowledge on how to operate this system in either windows or mac. Besides that, it would be highly recommended to use a 64-bit processor with an Intel i5 at least to avoid laggings.

### **1.2.2 Xampp Server**

To run this system, several servers are needed in order to make the system function completely. Xampp server supports both MySQL and PHP and it is one of those local servers which runs smoothly in all personal computers. This is selected due to its reliability and the ease of usage by end clients and admins. The secondary reason to choose this server is for its simple usability and set up process. Xampp server is easy to set up and manage. Hence it makes developers and web makers to choose a great server like Xampp. This server is accepted in both Windows and MacOS.

### **1.2.3 PHP (Language)**

Considering a good language to build a system is very important as it would affect the entire system’s functionalities and progress. During the early stage of our system development, the first language that we considered was ASP.NET. This language was undertaken as we had 2 programmers who had wide knowledge on this particular language. Sooner during the mid-duration of our 2nd sprint, there were some unavoidable issues with the development progress. We then had to drop the language and adopt another one instead. After several meetings, as a scrum master I appointed Keerthigai as the lead programmer to build the system using PHP language. Since Keerthi has a very good knowledge and practical exposures on PHP, she was able to handle the system development process with the help of our secondary programmer and the rest of us. PHP was way more flexible to handle and it was a bit easy for us to pass around among us to do our necessary parts according to our working schedule. Besides that, we were also able to implement almost all the functions as per the coursework requirements despite having 1 or 2 bugs in it.

### **1.2.4 MYSQL (phpMyAdmin)**

Since the language and database server is directly associated with phpMyAdmin administration tool, it was a little easy for us to manage the database of this system. the MYSQL database server was used to add, edit and delete the data that is produced by the user in the web system. this process can only be done by the administrator. While tracking the usernames, emails, passwords produced by the user, the admin can easily oversee the process from the phpMyAdmin server. On the other hand, it is also possible to see any uploaded ideas to the forum. The admin will also be able to edit or delete them anytime. By using MYSQL as the database tool for this system, it was a little flexible to track and run the script before accessing the website.

## **1.3 Installation Steps**

To run this system in a windows or mac operating system, there are several components which are to be considered installing and starting it before installing. Major components that are needed includes Apache, MYSQL and PHP. In order to have a smooth performance, it is primarily required to have at least 512 RAM and 2GB disk space. Before running the website, start phpMyAdmin and choose Apache version. It is highly recommended to use the latest version to avoid laggings. Once done, select MYSQL and run it. Click enter to use MYSQL as the default data directory. Select PHP to install. Once done, run the database scripts that have been coded before running the web system on google. Once the script has started, make sure your internet connection is stable. The final configuration is to run the website on a selected engine like Google or Mozilla Firefox to test the web system if its working. Recommended engine would be Google. Once the system runs, the user can start to access the web as usual and proceed to register or login process.

# 2.0 Artifacts

## **2.1 Credentials & Links**

|  |  |  |
| --- | --- | --- |
| Accounts | Hyperlinks | Password | Sign in procedures |
| Gmail | [fourstarsuog4@gmail.com](mailto:fourstarsuog4@gmail.com) | *KAJENKAILIN@minkitty\_4stars* |
| Trello | <https://trello.com/w/fourstarsenterpisegroupcw> | *Google sign in* |
| GitHub | <https://github.com/teamfourstar> | *Google sign in* |
| Presentation | <https://www.youtube.com/watch?v=vwD3xWgFyQw> | *YouTube* |
| Web system | <http://virtualhost.local/loginreg.php> | *Access using XAMPP* |

## **2.2 Group members & roles**

|  |  |  |
| --- | --- | --- |
| UOG ID | Members | Role(s) |
| 001192656 | **KAJENDRA NAATH RAJAN** | Scrum master, UI designer & tester |
| 001192641 | **HO KAI LIN** | Programmer & tester |
| 001192663 | **KEERTHIGAI DEVI THEVENTHIRAN** | Programmer, tester & Database designer |
| 001192652 | **LEE MIN QI** | Database designer, meeting minutes handler & tester |
| 001192627 | **BRIAN EE CHI CHIN** | Diagrams handler & tester |

## **2.3 System Test Accounts Credentials**

|  |  |  |
| --- | --- | --- |
| Role | Username | Password |
| Administrator | faten | Faten123 |
| QA Manager | kajen | password |
| QA Coordinator | minqi | minqi1212 |
| Staff 1 | kailin | kailin@123 |
| Staff 2 | brian | brian@55 |
| Staff 3 | keerthi | NotPassword |

## **2.4 Product Backlog**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | User Story | Acceptance Criteria | Story Point | Priority | Sprint |
| Staff Role | | | | | |
| 1 | As a staff I want to be able login into the system so that I am able to submit ideas. | * Staff should be able to enter their username and password on the login page. * Staff should be able to click on a "Login" button to submit their credentials. * Staff should receive an error message if the entered username or password is incorrect. * Staff should be redirected to the system's main page after successful login. * Staff should be able to logout from the system after logging in. | 5 | MUST | 1 |
| 2 | As a staff I want to be able to sign up to the system so that I can have an account in order to access the system. | * Staff should be able to click on a "Sign Up" button on the system's login page. * Staff should be directed to a registration page where they can enter their personal information such as name, email, and password. * Staff should be able to click on a "Submit" button to register their account. * Staff should receive an error message if any required field is left blank or if the entered email is already registered. * Staff should be able to login with their newly created account after successful registration. | 5 | MUST | 1 |
| 3 | As a staff I want to be able to read the terms and conditions so that I can know what T&C is. | * Staff should be able to access the "Terms and Conditions" page from the system's main menu or footer. * Staff should be able to read and understand the terms and conditions clearly. * Staff should be able to scroll through the entire terms and conditions content. * Staff should be able to close the "Terms and Conditions" page and return to the previous page. | 4 | MUST | 1 |
| 4 | As a staff I want to be able to choose to upload a pdf document so that I am able to support my ideas. | * Staff should be able to choose a PDF file from their local device by clicking on an "Upload" button. * Staff should receive a confirmation message after a successful file upload. * Staff should be able to view the uploaded PDF file and its details (e.g., file name, size, date) on the system. * Staff should be able to replace or delete the uploaded PDF file if needed. * Staff should receive an error message if the uploaded file is not in PDF format or exceeds the maximum file size allowed by the system. | 5 | MUST | 1 |
| 5 | As a staff I want to be able to add tags to my ideas so that I am to categorize them. | * Staff should be able to add tags to their ideas while submitting or editing them. * Staff should be able to enter tags as keywords or phrases that describe the content or category of their ideas. * Staff should be able to add multiple tags to a single idea. * Staff should be able to view and manage the tags associated with their ideas in their account settings or idea management page. * Staff should be able to search and filter ideas based on tags to easily categorize and retrieve their ideas. | 5 | SHOULD | 2 |
| 6 | As a staff I want to be able to see the ideas of other people so that I can keep that as a reference for my submission. | * Staff should be able to access a page or section where they can view ideas submitted by other users. * Staff should be able to see the title, description, and tags (if available) of the ideas submitted by other users. | 3 | COULD | 2 |
| 7 | As a staff I want to be able to see the ideas of other people so that I can comment on their ideas. | * Staff should be able to add comments to ideas submitted by other users. * Staff should be able to view their own comments as well as comments made by other users on the ideas. * Staff should be able to edit or delete their own comments if needed. * Staff should receive a notification or email if someone comments on their idea or replies to their comments. | 3 | SHOULD | 2 |
| 8 | As a staff I want to be able to be invisible through my ideas so that I can be free in terms of sharing contents. | * Staff should be able to mark their ideas as "invisible" or "private" during submission or editing. * Staff's invisible ideas should not be visible to other users in the public view or search results. * Staff should be able to view and manage their invisible ideas in their account settings or idea management page. | 3 | WONT | 2 |
| 9 | As a staff I want to be able to receive confirmation email so that I can know I have successfully submitted my ideas. | * Staff should receive a confirmation email after successfully submitting their ideas. * The confirmation email should include details such as the idea title, submission date, and any other relevant information. | 3 | MUST | 2 |
| 10 | As a staff I want to be able to see the ideas of other people so that I can like or dislike their ideas. | * Staff should be able to like or dislike ideas submitted by other users. * Staff should be able to click on a "Like" or "Dislike" button associated with each idea to express their preference. * Staff should be able to view the total number of likes and dislikes received by each idea. * Staff should be able to see their own likes and dislikes on the ideas they have voted on. * Staff should not be able to vote multiple times on the same idea or manipulate the voting system. | 5 | SHOULD | 2 |
| 11 | As a staff I want to be able to see the ideas of other people so that I can comment their ideas. | * Staff should be able to access a page or section where they can view ideas submitted by other users. * Staff should be able to see the title, description, and tags (if available) of the ideas submitted by other users. | 3 | SHOULD | 3 |
| 12 | As a staff I want to be able to choose to be visible so that I can share my ideas to other people. | * Staff should be able to choose the visibility settings for their own ideas during submission or editing. * Staff should be able to set their ideas as "visible" or "public" to share them with other users. * Staff should be able to set their ideas as "invisible" or "private" to restrict access to them. * Staff should be able to change the visibility settings of their ideas from "visible" to "invisible" or vice versa if needed. | 3 | MUST | 3 |
| 13 | As a staff I want to be able to receive notification email so that I know someone has replied to my ideas. | * Staff should receive notification emails when someone replies to their ideas. * The notification email should include details such as the idea title, reply content, and submission date. * The notification email should be sent to the email address associated with the staff's account. | 3 | SHOULD | 4 |
| Department QA Coordinator | | | | | |
| 14 | As a Department QA Coordinator I want to be able to access the system so that I am able to manage the system. | * Department QA Coordinator should be able to access the system with a unique username and password. * Department QA Coordinator should have appropriate permissions and access levels to manage the system, including managing user accounts, reviewing and approving/rejecting idea submissions, and managing system settings. | 2 | MUST | 4 |
| 15 | As a Department QA Coordinator I want to be able to receive notification email so that I am able to get notified for every idea submission. | * Department QA Coordinator should receive notification emails for every idea submission made in the system. | 1 | SHOULD | 4 |
| QA Manager | | | | | |
| 16 | As a QA Manager I want to be able to access the system so that I am able to oversee all the process. | * QA Manager should be able to access the system with a unique username and password. * QA Manager should have appropriate permissions and access levels to oversee all processes in the system, including reviewing and managing ideas, tags, and data. | 2 | MUST | 4 |
| 17 | As a QA Manager I want to be able to add tags to my ideas so that I that categories will be increased. | * QA Manager should be able to download the ideas data from the system. | 1 | SHOULD | 4 |
| 18 | As a QA Manager I want to be able to delete the tag for ideas so that categories will be reduced once its not being used. | * QA Manager should be able to delete tags from ideas in the system. * The deletion of tags should be reflected in the system immediately and accurately. * Deleted tags should be removed from all associated ideas in the system, ensuring that categories are reduced once tags are no longer being used. | 3 | SHOULD | 4 |
| 19 | As a QA Manager I want to be able to download the ideas data so that I can keep track and manage them. | * QA Manager should be able to download the ideas data from the system. | 1 | SHOULD | 4 |
| 20 | As a QA Manager I want to be able to download the uploaded support document so that I am able to manage the data. | * QA Manager should be able to download the uploaded support documents attached to ideas in the system. | 1 | MUST | 4 |
| Admin | | | | | |
| 21 | As an Admin I want to be able to access the system so that I can manage the data. | * Admin should be able to access the system using appropriate authentication and authorization mechanisms. * Admin should have administrative privileges to manage the data in the system, including creating, updating, and deleting data as needed. * Admin should be able to access all the functionalities and features of the system necessary for data management, such as user management, data configuration, and system settings. * Admin should be able to perform data management tasks efficiently and effectively, with a user-friendly and intuitive interface. * The system should provide appropriate error handling and feedback if there are any issues with Admin's access or data management tasks. * Admin should have the necessary permissions and rights to access and manage all relevant data in the system, while ensuring data security and confidentiality. | 5 | MUST | 4 |

## **2.5 Sprint 1 Backlog**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | User Story | Task ID | Task Breakdown | Story Point | Development |
| 1 | As a staff, I want to login into the system so that I am able to submit ideas. | Staff-01 | 1. Develop a login page with a username and password input fields for staff members to enter their credentials. 2. Implement authentication and authorization mechanisms to validate staff members' credentials and grant access to the system upon successful login. 3. Create a sign-up page with required fields for staff members to provide their information, such as username, email, and password. | 5 | * Design and implement the login page UI * Develop backend authentication logic * Implement frontend login functionality * Implement error handling for incorrect credentials * Implement redirection after successful login * Develop logout functionality * Write unit tests for login functionality * Conduct testing and debugging * Document implementation details and conduct code review * Deployment and integration |
| 2 | As a staff, I want to sign up to the system so that I can have an account in order to access the system. | Staff-02 | 1. Develop a user registration form that allows staff members to sign up for an account in the system. 2. Implement form validation to ensure that all required fields are filled out correctly, such as username, password, email, etc. 3. Include password validation rules, such as minimum length, complexity requirements, and password confirmation. 4. Implement email verification to ensure that the registered email address is valid and belongs to the staff member. 5. Store the registered user information securely in the system's database, including username, password (hashed), email, and any other required information. | 3 | * Design and implement the login page UI * Develop backend authentication logic * Implement frontend login functionality * Implement error handling for incorrect credentials * Implement redirection after successful login * Develop logout functionality * Write unit tests for login functionality * Conduct testing and debugging * Document implementation details and conduct code review * Deployment and integration |
| 3 | As a staff, I want to read the terms and conditions so that I am able to submit ideas into the system. | Staff-03 | 1. Develop a terms and conditions page that staff members can access from the system. 2. Design the terms and conditions page with clear and easy-to-read content. 3. Implement a user-friendly interface that allows staff members to scroll and read the entire terms and conditions content. 4. Include a confirmation button or checkbox for staff members to acknowledge that they have read and understood the terms and conditions. | 2 | * Design and implement the login page UI * Develop backend authentication logic * Implement frontend login functionality * Implement error handling for incorrect credentials * Implement redirection after successful login * Develop logout functionality * Write unit tests for login functionality * Conduct testing and debugging * Document implementation details and conduct code review * Deployment and integration |
| 4 | As a staff, I want to agree to the terms and conditions so that I am able to submit ideas into the system. | Staff-04 | Add a checkbox or similar mechanism on the idea submission form that requires staff members to agree to the terms and conditions before submitting their ideas.  Implement server-side validation to check if the terms and conditions have been agreed to before allowing the idea submission to be processed.  Include appropriate error handling and feedback mechanisms to notify staff members if they have not agreed to the terms and conditions. | 1 | * Design and implement the login page UI * Develop backend authentication logic * Implement frontend login functionality * Implement error handling for incorrect credentials * Implement redirection after successful login * Develop logout functionality * Write unit tests for login functionality * Conduct testing and debugging * Document implementation details and conduct code review * Deployment and integration |

## **2.6 Sprint 2 Backlog**

Sprint 02

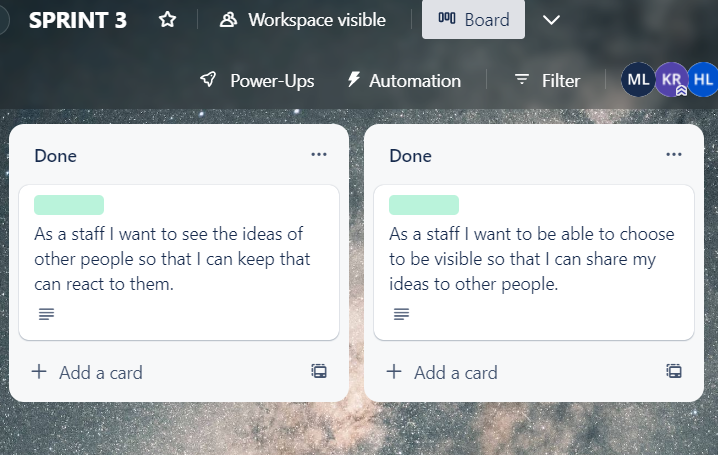
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | User Story | Task ID | Task Breakdown | Story Point | Development |
| 4 | As a staff I want to choose to upload a pdf document so that I am able to support my ideas. | Staff-05 | 1. Add a file upload feature to the idea submission form that allows staff members to select and upload a PDF document to support their ideas. 2. Implement server-side validation to check if the uploaded file is in PDF format and within the allowed file size limit. 3. Store the uploaded PDF document securely on the server, ensuring proper file naming conventions and directory structure. 4. Implement a mechanism to associate the uploaded PDF document with the corresponding idea submission, such as linking the file path or storing it in a database. 5. Add appropriate error handling and feedback mechanisms to notify staff members if the PDF document upload fails, and provide instructions on how to correct the issue. 6. Test the PDF document upload functionality thoroughly, including positive and negative scenarios, and debug any issues found. 7. Update the system documentation to include instructions for staff members on how to upload PDF documents to support their ideas. | 3 | * Design and implement UI for file upload functionality * Develop backend logic for handling file uploads * Implement frontend functionality for selecting and uploading PDF documents * Implement validation and error handling for file size, format, and other requirements * Store uploaded files securely on the server or a designated file storage system * Write unit tests for file upload functionality * Conduct testing and debugging * Document implementation details and conduct code review * Deployment and integration |
| 5 | As a staff I want to add tags to my ideas so that I am able to categorize them. | Staff-06 | 1. Add a tags field to the idea submission form that allows staff members to input tags to categorize their ideas. 2. Implement server-side validation to check for valid tag inputs, such as ensuring they are alphanumeric and within a certain character limit. 3. Store the submitted tags securely on the server, ensuring proper handling and storage practices. 4. Implement a mechanism to associate the submitted tags with the corresponding idea submission, such as storing them in a database or linking them to the submission ID. 5. Implement a feature to display the tags associated with each idea in the system, such as in the idea details page or in search/filter options. 6. Test the idea tagging functionality thoroughly, including positive and negative scenarios, and debug any issues found. 7. Update the system documentation to include instructions for staff members on how to add tags to their ideas for categorization. | 2 | * Design and implement UI for adding tags to ideas * Develop backend logic for managing tags and their association with ideas * Implement frontend functionality for adding, editing, and removing tags * Implement tag-based filtering or search functionality * Write unit tests for tag management functionality * Conduct testing and debugging * Document implementation details and conduct code review * Deployment and integration |
| 6 | As a staff I want to see the ideas of other people so that I can keep that as a reference for my submission. | Staff-07 | 1. Create a user interface (UI) component, such as a comment section, to allow staff members to enter and submit comments for specific ideas. 2. Implement server-side logic to store and retrieve comments in the database or data source, associating them with the corresponding ideas. 3. Implement validation checks to ensure that staff members can only comment on ideas they are authorized to access and that the comments meet any required criteria, such as length or content restrictions. 4. Implement notifications or alerts to notify the original submitter of the idea when a comment is added to their idea. 5. Test the idea commenting functionality thoroughly, including different scenarios, such as adding comments to different ideas and validating the comments stored in the database. 6. Update the system documentation to include instructions for staff members on how to view and comment on ideas submitted by other people. | 3 | * Design and implement UI for viewing ideas of other people * Develop backend logic for fetching and displaying ideas of other people * Implement frontend functionality for browsing and viewing ideas of other people * Implement sorting or filtering options for ideas * Write unit tests for idea viewing functionality * Conduct testing and debugging * Document implementation details and conduct code review * Deployment and integration |
| 7 | As a staff I want to see the ideas of othe people so that I can comment on their ideas. | Staff-08 | 1. Implement a feature that allows staff members to view and comment on ideas submitted by other people in the system. 2. Create a user interface (UI) component, such as a comment section, to allow staff members to enter and submit comments for specific ideas. 3. Implement server-side logic to store and retrieve comments in the database or data source, associating them with the corresponding ideas. 4. Implement validation checks to ensure that staff members can only comment on ideas they are authorized to access and that the comments meet any required criteria, such as length or content restrictions. 5. Implement notifications or alerts to notify the original submitter of the idea when a comment is added to their idea. 6. Test the idea commenting functionality thoroughly, including different scenarios, such as adding comments to different ideas and validating the comments stored in the database. 7. Update the system documentation to include instructions for staff members on how to view and comment on ideas submitted by other people. | 3 | * Design and implement UI for commenting on ideas * Develop backend logic for handling comments on ideas * Implement frontend functionality for adding, editing, and deleting comments * Implement notifications or alerts for new comments on subscribed ideas * Write unit tests for commenting functionality * Conduct testing and debugging * Document implementation details and conduct code review * Deployment and integration |
| 8 | As a staff I want to be invisible through my ideas so that I can be free in terms of sharing contents. | Staff-09 | 1. Analyse the current system architecture and identify areas where staff members' identity is exposed through their submitted ideas. 2. Design and implement a solution to anonymize staff members' ideas, such as removing personal information, usernames, or any identifying details from the ideas. 3. Modify the user interface (UI) components related to idea submission and display to ensure that staff members' identity is not exposed, such as replacing usernames with anonymous identifiers or generic labels. 4. Implement server-side logic to handle the anonymization process, including data processing and storage, while complying with relevant privacy and data protection regulations. 5. Implement validation checks to ensure that the anonymization process is applied consistently and accurately to all submitted ideas by staff members. 6. Update the system documentation to include instructions for staff members on how their ideas will be anonymized and the limitations of the anonymization process. 7. Conduct thorough testing to validate the anonymization process and ensure that staff members' identity is not exposed through their ideas in any part of the system. 8. Collaborate with the QA team to conduct additional testing, including security testing, to verify the effectiveness of the anonymization process. 9. Provide support and address any issues or questions from staff members regarding the anonymization process. | 5 | * Implement functionality for setting ideas as invisible or private * Develop backend logic for managing invisible ideas and their visibility settings * Implement frontend functionality for toggling the visibility of ideas * Implement authentication and authorization checks for accessing invisible ideas * Write unit tests for invisible idea functionality * Conduct testing and debugging * Document implementation details and conduct code review * Deployment and integration |
| 9 | As a staff I want to be able to receive confirmation email so that I know I have successfully submitted my ideas. | Staff-10 | 1. Design and implement a confirmation email template that will be sent to staff members after successfully submitting their ideas. 2. Implement server-side logic to trigger the sending of confirmation emails upon successful idea submission by staff members. 3. Update the database schema to include fields for storing the necessary information related to confirmation emails, such as recipient email address, timestamp, and status. 4. Modify the UI components related to idea submission to display a confirmation message after successful submission and provide information on the expected delivery time of the confirmation email. 5. Implement error handling and logging to capture and handle any errors or exceptions related to the sending of confirmation emails. 6. Collaborate with the QA team to conduct testing of the confirmation email functionality, including verifying that the confirmation emails are sent accurately and received by staff members. 7. Update the system documentation to include instructions for staff members on how to check for and use the confirmation email to confirm the successful submission of their ideas. 8. Provide support and address any issues or questions from staff members regarding the confirmation email process. | 2 | * Implement email sending functionality for confirmation notifications * Develop backend logic for sending confirmation emails after idea submission * Implement email template design for confirmation emails * Implement email delivery and error handling mechanisms * Write unit tests for email sending functionality * Conduct testing and debugging * Document implementation details and conduct code review * Deployment and integration |

2.7 Sprint Backlog 3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| User Story ID | User Story | Tasks ID | Task Breakdown | Developer(s) |
| 10 | As a staff I want to see the ideas of other people so that I can keep that can like or dislike to their ideas | 1 | Develop the function by developers within the time frame | Ho Kai Lim |
| 2 | Do testing by testers |
| 11 | As a staff I want to be able to choose to be visible so that I can share my ideas to other people. | 1 | Develop the function by developers within the time frame | Keerthigai Devi |
| 2 | Do testing by testers |

Table above is the table for sprint 3. In the table, consist of the user story for the sprint.

Below is the screenshot for sprint 3 at Trello.



## 2.8 Sprint Backlog 4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| User Story ID | User Story | Tasks ID | Task Breakdown | Developer(s) |
| 1 | As a QA Manager I want to be able to access the system so that I can oversee all the process. | 1 | Develop the function by developers within the time frame | Keerthigai Devi |
| 2 | Do testing by testers |
| 2 | As a QA Manager I want to be able to add new category so that categories will be seen in the main forum page. | 1 | Develop the function by developers within the time frame | Keerthigai Devi |
| 2 | Do testing by testers |
| 3 | As a QA Manager I want to be able to delete any latest or old ideas so that categories will be reduced once it’s not being used. | 1 | Develop the function by developers within the time frame | Keerthigai Devi |
| 2 | Do testing by testers |
| 4 | As a QA Manager I want to be able to ban any staffs or coordinators from posting any ideas so that there won't be any inappropriate contents in the forum. | 1 | Develop the function by developers within the time frame | Keerthigai Devi |
| 2 | Do testing by testers |
| 5 | As a QA Manager I want to be able to upload any relevant files to my idea so that uploaded files can be seen in the posted idea. | 1 | Develop the function by developers within the time frame | Keerthigai Devi |
| 2 | Do testing by testers |
| 6 | As a QA Manager I want to be able to upload any supporting documents so that I am able to see the files in the idea section. | 1 | Develop the function by developers within the time frame | Keerthigai Devi |
| 2 | Do testing by testers |
| 7 | As a QA Coordinator I want to be able to access the system so that I am able to track all the latest ideas of staffs. | 1 | Develop the function by developers within the time frame | Keerthigai Devi |
| 2 | Do testing by testers |
| 8 | As a QA Coordinator I want to be able to upload any relevant files to my idea so that uploaded files can be seen in the posted idea. | 1 | Develop the function by developers within the time frame | Keerthigai Devi |
| 2 | Do testing by testers |
| 9 | As a QA Coordinator I want to be able to see the statistics of the entire portal so that I can keep track of the overall performance of the forum. | 1 | Develop the function by developers within the time frame | Keerthigai Devi |
| 2 | Do testing by testers |
| 10 | As a staff I want to be able choose to be visible so that my ideas can be seen by other people. | 1 | Develop the function by developers within the time frame | Ho Kai Lin |
| 2 | Do testing by testers |
| 11 | As a Staff I want to be able to upload any relevant files to my idea so that uploaded files can be seen in the posted idea. | 1 | Develop the function by developers within the time frame | Ho Kai Lin |
| 2 | Do testing by testers |
| 12 | As an Admin I want to be able to maintain all the system's data so that system will be up to date. | 1 | Develop the function by developers within the time frame | Ho Kai Lin |
| 2 | Do testing by testers |

Table above is the table for sprint 4. In the table, consist of the user story for the sprint 4. Sprint 4 have more than sprint 3 because of our team’s work had been rearrange at sprint 3.

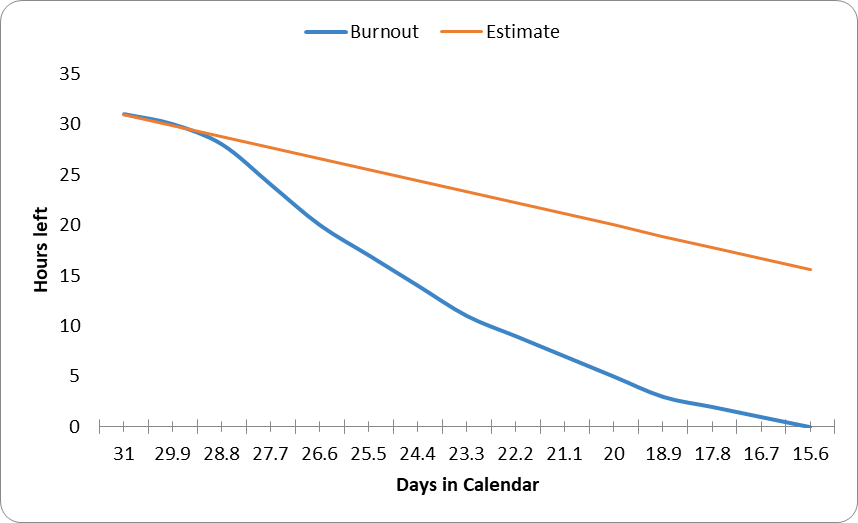
Below is the screenshot for sprint 4 at Trello.



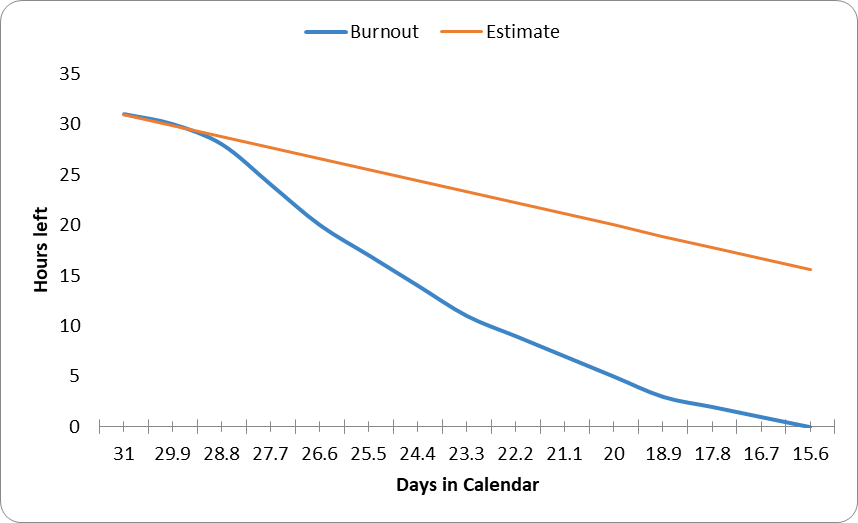
2.9 Burndown Charts

In this section there are four burndown charts will be presented. The four burndown chart is from four different sprint.

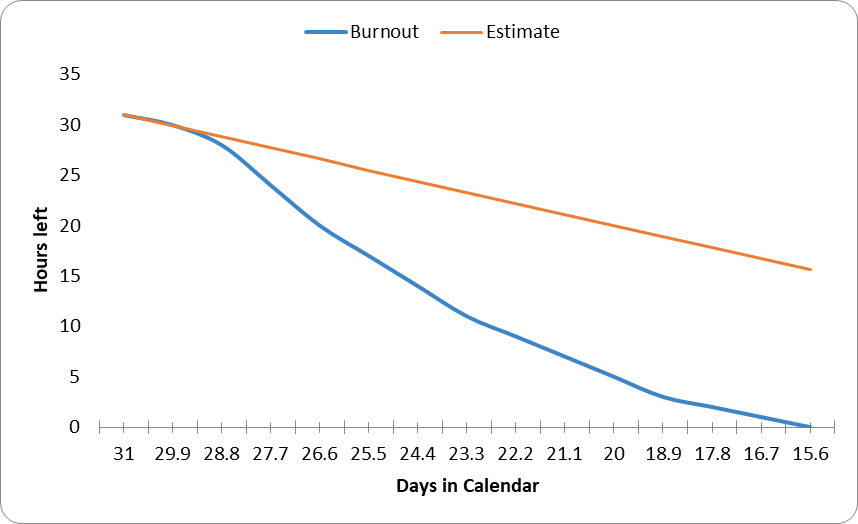
Sprint 1 burndown chart



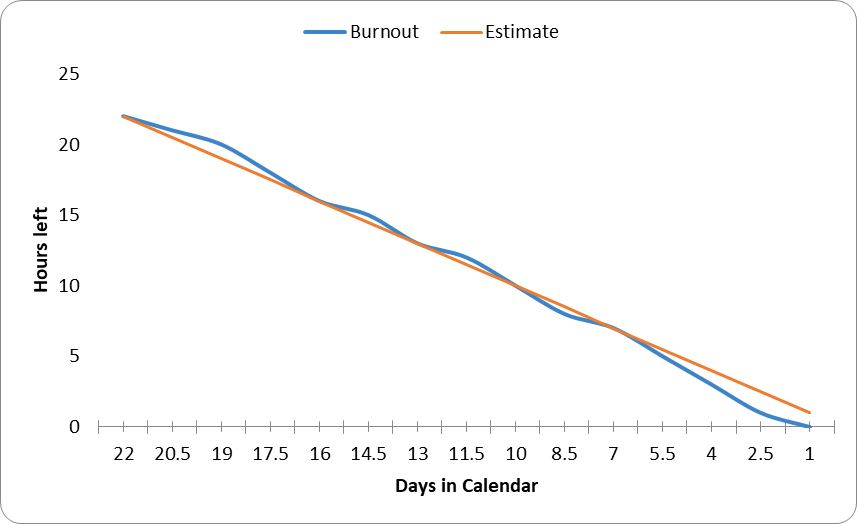
Sprint 2 burndown chart



Sprint 3 Burndown chart



Sprint 4 burndown chart

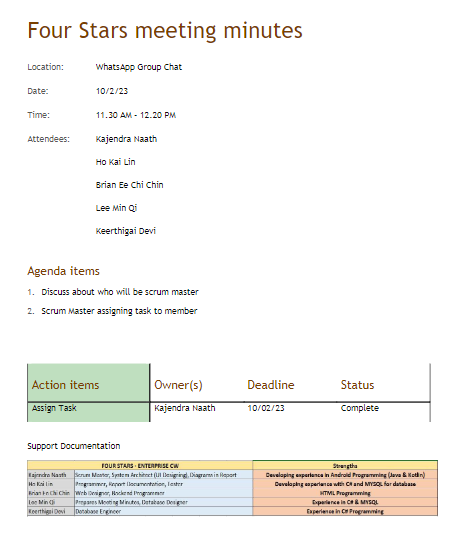


## 2.10 Meeting Minutes

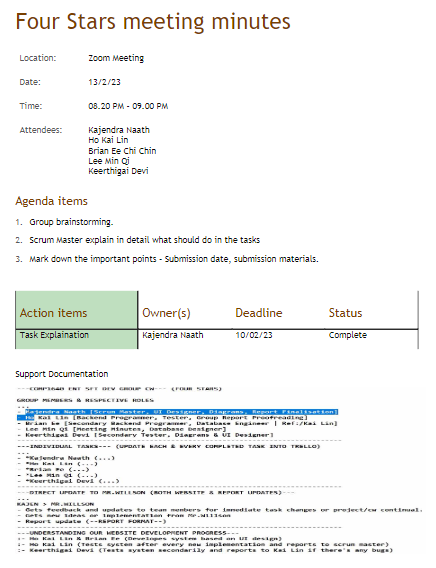
Working in a team, meeting is a very important session. This is because meeting able to gather everyone idea and ensure the team progression is at the right track.

For our team, we had managed to held 13 meeting throughout the whole coursework. Below are the screenshot of the meeting minutes.

* 10/02/2023



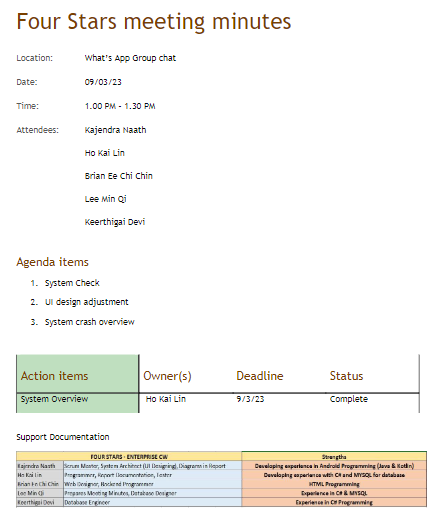
* 13/02/2023



* 02/03/2023



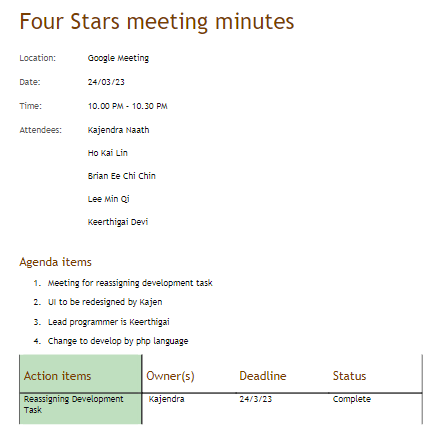
* 09/03/2023



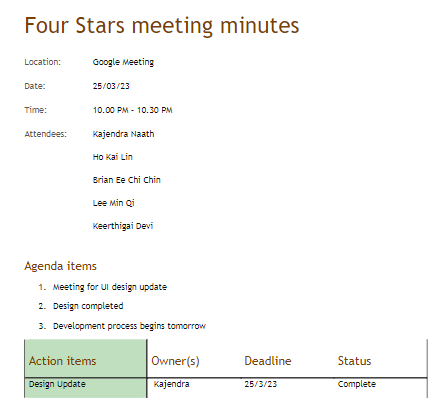
* 23/03/2023



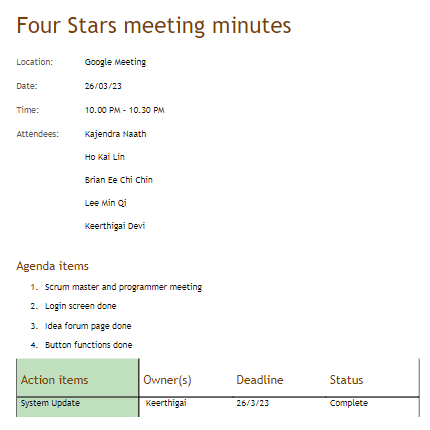
* 24/03/2023



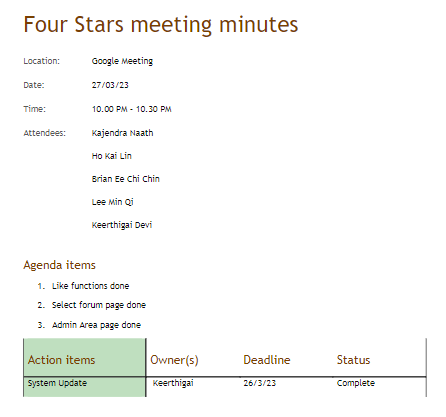
* 25/03/2023



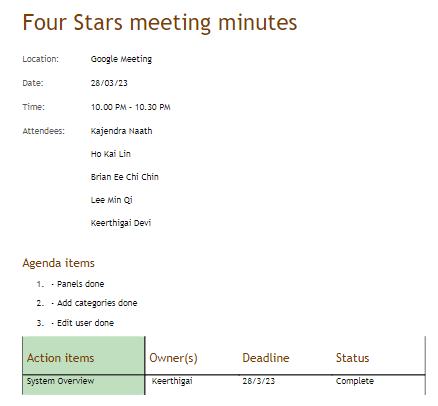
* 26/03/2023



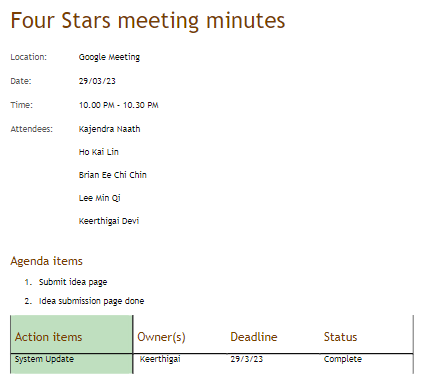
* 27/03/2023



* 28/03/2023



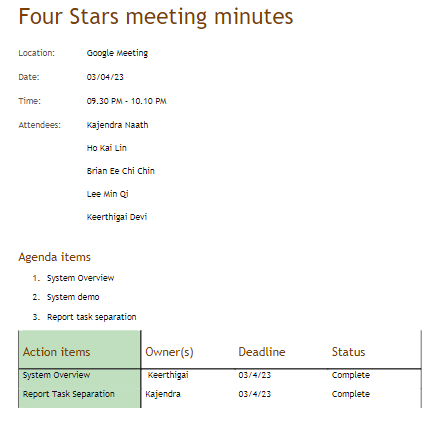
* 29/03/2023



* 30/03/2023



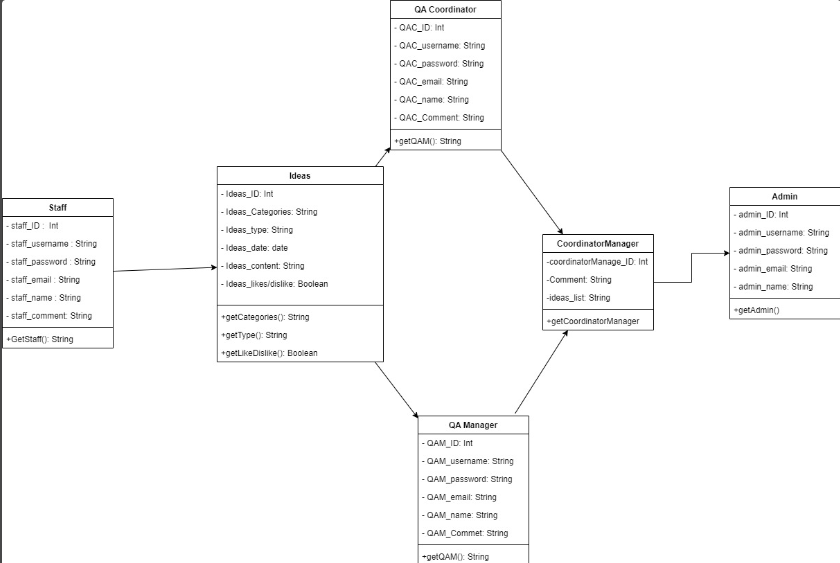
* 03/04/2023



2.11 UML Case Diagram

In this section, a case diagram will be presented. The UML diagram is a diagram that design for the system that our team developed. The UML diagram explain generally how the database in the system works and what data will be implemented into the database.

Below is the UML Case Diagram.



## 2.13 Entity Relationship Diagram (ERD)

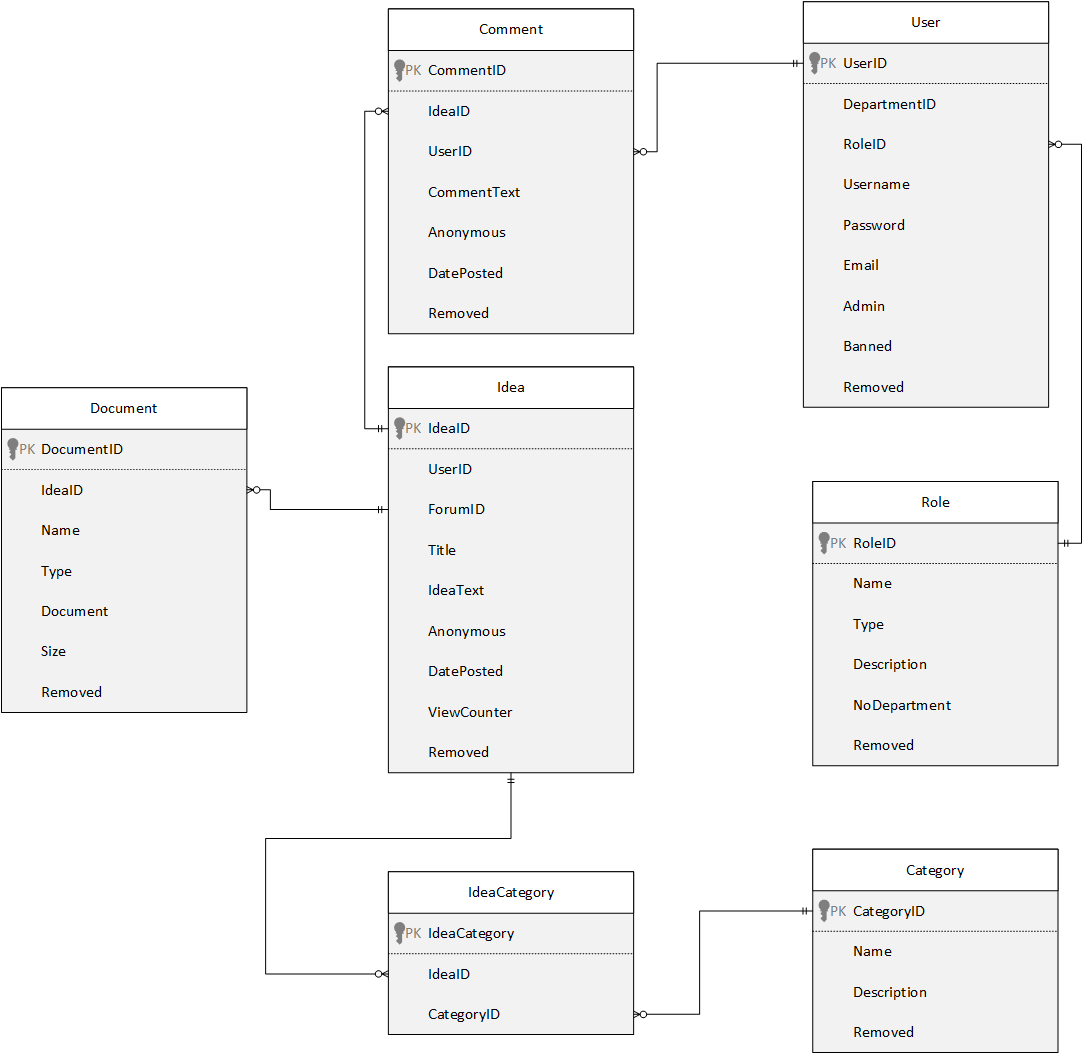


Figure 1 Entity relationship diagram

## 2.14 Data Dictionary

Comment

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Comment | | | | | |
| Field Name | Data Type | Field Length | Constraint | Description | Example |
| CommentID | Int | Max | Primary key (PK) of Comment table, not null | ID of a comment | 6 |
| IdeaID | Int | max | Foreign Key (FK) of Idea table | ID of an idea | 9 |
| UserID | int | max | Not null | ID of a user who made the comment | 1 |
| CommentText | varchar | max | Not null | Details of the comment | huhu |
| Anonymous | Tinyint | 1 | Not null | Displays username if 1, else name is hidden if 0 | 0 |
| DatePosted | datetime | 20 | Not null | Date posted when comment is made | 10/10/2022 00:00:00 |
| Removed | Tinyint | 1 | Null | Displays 1 comment will be hidden from users | 0 |

Table 1 Comment data dictionary

User

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| User | | | | | |
| Field Name | Data Type | Field Length | Constraint | Description | Example |
| UserID | Int | max | Primary key (PK) of User table, not null | User ID for users | 1 |
| DepartmentID | Int | 5 | Null | Department ID of users | 1 |
| RoleID | Int | 10 | Foreign Key (FK) of Role table, not null | Role ID for users | 1 |
| Username | Varchar | 500 | Not null | Username for users | kajen |
| Password | Varchar | 50 | Not null | Password for users | potato |
| Email | Varchar | 50 | Not null | Email for users | kajen@mail.com |
| Admin | boolean | 1 | Not null | Display 1, If user is admin | 1 |
| Banned | boolean | 1 | Not null | Display 1 if user account is banned | 0 |
| Removed | boolean | 1 | Not null | Display 1 if user account is removed | 0 |

Table 2 User data dictionary

Document

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Document | | | | | |
| Field Name | Data Type | Field Length | Constraint | Description | Example |
| DocumentID | Int | Max | Primary key (PK), not null | Document ID for documents | 1 |
| IdeaID | Int | Max | Foreign Key (FK) of Idea table, not null | Idea ID of the idea | 1 |
| Name | varchar | Max | Not null | Name of the document | Logo.jpg |
| Type | varchar | Max | Not null | File type of the document submitted | Image/jpeg |
| Document | Nvarchar | max | Not null | File submitted by users | [BLOB – 12.2 KiB] |
| Size | int | 1000 | Not null | Size of the file | 12489 |
| Removed | Tinyint | 1 | null | 1 if the document is removed | 0 |

Table 3 Document data dictionary

Category

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Category | | | | | |
| Field Name | Data Type | Field Length | Constraint | Description | Example |
| CategoryID | Int | 11 | Primary key (PK) of Category table, not null | ID of a category | 2 |
| Name | Varchar | 20 | Not null | Name of a category | Reunion |
| Description | Varchar | 255 | Null | Details of a category | Medical problems |
| Removed | tinyint | 1 | Not null | 1 if the category is currently removed | 0 |

Table 4 Category data dictionary

Idea

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Idea | | | | | |
| Field Name | Data Type | Field Length | Constraint | Description | Example |
| IdeaID | Int | Max | Primary key (PK) of Idea table, not null | ID of an idea | 1 |
| UserID | Int | Max | Foreign key (FK) of user table, not null | ID of a user | 2 |
| ForumID | Int | Max | Foreign key (FK) of forum table, not null | ID of a forum | 4 |
| Title | Varchar | Max | Not null | Title name of the idea | Server |
| IdeaText | Varchar | Max | Not null | Details of the idea | Equipment problem |
| Anonymous | Tinyint | 1 | Not null | Displays username if 1, else name is hidden if 0 | 1 |
| DatePosted | datetime | 20 | Not null | The date the idea is posted | 10/10/2222 00:00:00 |
| ViewCounter | Int | max | Not null | Total views of an idea | 100 |
| Removed | tinyint | 1 | Not null | 1 if idea is currently removed | 0 |

Table 5 Idea data dictionary

Role

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Role | | | | | |
| Field Name | Data Type | Field Length | Constraint | Description | Example |
| RoleID | Int | Max | Primary key (PK) of role table, not null | ID of a role | 1 |
| Name | Text | 50 | Not null | Name of a role | Academic Staff |
| Type | Text | 50 | Not null | Type of role | Manager |
| Description | Varchar | Max | Null | Details of a role | null |
| NoDepartment | Int | 1 | Not null | 1 if user is not in any department | 1 |
| Removed | Tinyint | 1 | Not null | 1 if role is currently removed | 1 |

Table 6 Role data dictionary

IdeaCategory

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| IdeaCategory | | | | | |
| Field Name | Data Type | Field Length | Constraint | Description | Example |
| IdeaCategory | int | Max | Primary key (PK), not null | ID of Idea Category table | 1 |
| IdeaID | Int | Max | Foreign Key (FK) of Idea table, not null | Idea ID of idea table | 1 |
| CategoryID | int | Max | Foreign Key (FK) of Category table, not null | Category ID of category table | 1 |

Table 7 IdeaCategory data dictionary

## 2.15 Test Case results

Staff

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Staff | | | | |
| Test Case ID | Category | Condition | Expected Results | Actual Results |
| 1 | Register/login | Register a new account | Register success message prompt | Register success message prompt |
| 2 | Register/login | Register with an existing account | User already exist message will be prompt | User already exist message will be prompt |
| 3 | Register/login | Login with no email and no password | Email/Password is incorrect message will be prompt | Email/Password is incorrect message will be prompt |
| 4 | Register/login | Login with wrong email, no password | Email/Password is incorrect message will be prompt | Email/Password is incorrect message will be prompt |
| 5 | Register/login | Login with no email, wrong password | Email/Password is incorrect message will be prompt | Email/Password is incorrect message will be prompt |
| 6 | Register/login | Login with correct email, wrong password | Email/Password is incorrect message will be prompt | Email/Password is incorrect message will be prompt |
| 7 | Register/login | Login with wrong email, correct password | Email/Password is incorrect message will be prompt | Email/Password is incorrect message will be prompt |
| 8 | Register/login | Login with correct email, correct password | Go to select an idea page | Go to select an idea page |
| 9 | Register/login | Existing account is banned | Cannot login into the system | Cannot login into the system |
| 10 | Ideas | Can view other user’s idea | Idea can be viewed | Idea can be viewed |
| 11 | Ideas | Adds a new idea with no title, no description, no category, terms and condition not checked, no file upload, anonymous not checked | Message prompt, user must type a title, type a description, select a category, agree to the terms & conditions | Message prompt, user must type a title, type a description, select a category, agree to the terms & conditions |
| 12 | Ideas | Adds a new idea with title, no description, without a category, terms and condition not checked, no file upload, anonymous not checked | Message prompt, user must type description, select a category, agree to the terms & conditions | Message prompt, user must type description, select a category, agree to the terms & conditions |
| 13 | Ideas | Adds a new idea with no title, with description, without a category, terms & conditions not checked, no file upload, anonymous not checked | Message prompts, user must type a title, select a category, agree to the terms & conditions | Message prompts, user must type a title, select a category, agree to the terms & conditions |
| 14 | Ideas | Adds a new idea with title, without description, with a category, terms & conditions not checked, no file upload, anonymous not checked | Message prompts, user must type a title, must type description, agree to the terms & conditions | Message prompts, user must type a title, select a category, agree to the terms & conditions |
| 15 | Ideas | Adds a new idea with title, without description, without a category, terms & conditions checked, no file upload, anonymous not checked | Message prompts, user must type a title, must type description, select at least one category | Message prompts, user must type a title, must type description, select at least one category |
| 16 | Ideas | Adds a new idea with title, with description, without a category, terms & conditions not checked, no file upload, anonymous not checked | Message prompts, user must select at least one category, check terms & conditions | Message prompts, user must select at least one category, check terms & conditions |
| 17 | Ideas | Adds a new idea with title, with description, with a category, terms & conditions not checked, no file upload, anonymous not checked | Message prompts, user check terms & conditions | Message prompts, user check terms & conditions |
| 18 | Ideas | Adds a new idea with title, with description, with a category, terms & conditions checked, no file upload, anonymous not checked | Idea is posted successfully and name is displayed at idea forum | Idea is posted successfully and name is displayed at idea forum |
| 19 | Ideas | Adds a new idea with title, with description, with a category, terms & conditions checked, with a file upload | Idea is posted successfully and name is hidden at idea forum | Idea is posted successfully and name is hidden at idea forum |
| 20 | Ideas | When like button is clicked | Likes will add 1 if currently displayed 0 | Likes will add 1 if currently displayed 0 |
| 21 | Ideas | If another user likes an idea, the likes counter goes up by 1 | Likes currently 1, it will be 2 if it is liked | Likes currently 1, it will be 2 if it is liked |
| 22 | Ideas | When dislike button is clicked | Dislikes will add 1 if currently displayed 0 | Dislikes will add 1 if currently displayed 0 |
| 23 | Ideas | If another user dislikes an idea, the dislikes counter goes up by 1 | Dislikes currently 1, it will be 2 if it is disliked | Dislikes currently 1, it will be 2 if it is disliked |
| 23 | Ideas | Comment on another user’s idea | Comment is displayed under the idea | Comment is displayed under the idea |
| 25 | Ideas | Existing user is banned, likes/dislikes will be affected | Likes/dislike remains, ideas and comments of the banned user is hidden, prompt message this user is deleted | Likes/dislike remains, ideas and comments of the banned user is hidden, prompt message this user is deleted |
| 26 | Ideas | Existing user is unbanned, likes/dislikes will be affected | Likes/dislike remains, ideas and comments of the unbanned user will be displayed again in the idea forum | Likes/dislike remains, ideas and comments of the unbanned user will be displayed again in the idea forum |
| 27 | Navigation | Go to reports page | Go to reports page | Go to reports page |
| 28 | Statistics | View ideas per department | Can see statistics of overall ideas submit by each department | Can see statistics of overall ideas submit by each department |
| 29 | Statistics | Percentage per department | Can see overall percentage of ideas submit by each department | Can see overall percentage of ideas submit by each department |
| 30 | Statistics | Contributors per department | Can see the number of overall ideas submit by each department | Can see the number of overall ideas submit by each department |
| 31 | Exception reports | Ideas without comments | Can see total ideas submitted without comments | Can see total ideas submitted without comments |
| 32 | Exception reports | Anonymous ideas | Can see total ideas submitted by anonymous users | Can see total ideas submitted by anonymous users |
| 33 | Exception reports | Anonymous comments | Can see total number of comments given by anonymous users | Can see total number of comments given by anonymous users |
| 34 | Logout | Logout button clicked | Go back to login/registration page | Go back to login/registration page |

Table 8 Staff test results

Admin

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Admin | | | | |
| Test Case ID | Category | Condition | Expected Results | Actual Results |
| 1 | Register/login | Register a new admin | Cannot register as admin message prompt | Cannot register as admin message prompt |
| 2 | Register/login | Register with existing admin | User already exists message prompt | User already exists message prompt |
| 3 | Register/login | Login with no email, no password | Email or password is incorrect message prompt | Email or password is incorrect message prompt |
| 4 | Register/login | Login with wrong email, no password | Email or password is incorrect message prompt | Email or password is incorrect message prompt |
| 5 | Register/login | Login with no email, wrong password | Email or password is incorrect message prompt | Email or password is incorrect message prompt |
| 6 | Register/login | Login with wrong email, wrong password | Email or password is incorrect message prompt | Email or password is incorrect message prompt |
| 7 | Register/login | Login with correct email, wrong password | Email or password is incorrect message prompt | Email or password is incorrect message prompt |
| 8 | Register/login | Login with wrong email, correct password | Email or password is incorrect message prompt | Email or password is incorrect message prompt |
| 9 | Register/login | Login with correct email and password | Go to admin page | Go to admin page |
| 10 | User management | View users | Can view all user details | Can view all user details |
| 11 | User management | Edit user’s role | User’s role updated | User’s role updated |
| 12 | User management | Banned user | Banned user cannot access system and ideas/comments made by the banned user is hidden from other users | Banned user cannot access system and ideas/comments made by the banned user is hidden from other users |
| 13 | User management | Unbanned user | Unbanned user can regain access of the system and ideas/comments made by the unbanned user is visible to other users | Unbanned user can regain access of the system and ideas/comments made by the unbanned user is visible to other users |
| 14 | Navigation | Go to reports page | Go to reports page | Go to reports page |
| 15 | Statistics | View ideas per department | Can see statistics of overall ideas submit by each department | Can see statistics of overall ideas submit by each department |
| 16 | Statistics | Percentage per department | Can see overall percentage of ideas submit by each department | Can see overall percentage of ideas submit by each department |
| 17 | Statistics | Contributors per department | Can see the number of overall ideas submit by each department | Can see the number of overall ideas submit by each department |
| 18 | Exception reports | Ideas without comments | Can see total ideas submitted without comments | Can see total ideas submitted without comments |
| 19 | Exception reports | Anonymous ideas | Can see total ideas submitted by anonymous users | Can see total ideas submitted by anonymous users |
| 20 | Exception reports | Anonymous comments | Can see total number of comments given by anonymous users | Can see total number of comments given by anonymous users |
| 21 | Logout | Logout button clicked | Go back to login/registration page | Go back to login/registration page |

Table 9 Admin test case results

QA coordinator

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| QA coordinator | | | | |
| Test Case ID | Category | Condition | Expected Results | Actual Results |
| 1 | Register/Login | Register a new QA coordinator account | Register success message prompt | Register success message prompt |
| 2 | Register/Login | Register with an existing QA coordinator account | User already exists message prompt | User already exists message prompt |
| 3 | Register/Login | Login with no email, no password | Email or password is incorrect message prompt | Email or password is incorrect message prompt |
| 4 | Register/Login | Login with wrong email, no password | Email or password is incorrect message prompt | Email or password is incorrect message prompt |
| 5 | Register/Login | Login with no email, wrong password | Email or password is incorrect message prompt | Email or password is incorrect message prompt |
| 6 | Register/Login | Login with wrong email, wrong password | Email or password is incorrect message prompt | Email or password is incorrect message prompt |
| 7 | Register/Login | Login with wrong email, correct password | Email or password is incorrect message prompt | Email or password is incorrect message prompt |
| 8 | Register/Login | Login with correct email, wrong password | Email or password is incorrect message prompt | Email or password is incorrect message prompt |
| 9 | Register/Login | Login with correct email and correct password | Go to select an idea page | Go to select an idea page |
| 10 | Edit roles | If a user is banned | The banned account cannot access the system and all related data such as ideas/comments is hidden from other users | The banned account cannot access the system and all related data such as ideas/comments is hidden from other users |
| 11 | Edit roles | If a user is unbanned | The unbanned account can regain access to the system and all related data such as ideas/comments is now visible to other users | The unbanned account can regain access to the system and all related data such as ideas/comments is now visible to other users |
| 12 | Navigation | Go to reports page | Go to reports page | Go to reports page |
| 13 | Statistics | View ideas per department | Can see statistics of overall ideas submit by each department | Can see statistics of overall ideas submit by each department |
| 14 | Statistics | Percentage per department | Can see overall percentage of ideas submit by each department | Can see overall percentage of ideas submit by each department |
| 15 | Statistics | Contributors per department | Can see the number of overall ideas submit by each department | Can see the number of overall ideas submit by each department |
| 16 | Exception reports | Ideas without comments | Can see total ideas submitted without comments | Can see total ideas submitted without comments |
| 17 | Exception reports | Anonymous ideas | Can see total ideas submitted by anonymous users | Can see total ideas submitted by anonymous users |
| 18 | Exception reports | Anonymous comments | Can see total number of comments given by anonymous users | Can see total number of comments given by anonymous users |
| 19 | Logout | Logout button clicked | Go back to login/registration page | Go back to login/registration page |
| Test Case ID | Category | Condition | Expected Results | Actual Results |
| 1 | Register/Login | Register a new QA coordinator account | Register success message prompt | Register success message prompt |
| 2 | Register/Login | Register with an existing QA coordinator account | User already exists message prompt | User already exists message prompt |
| 3 | Register/Login | Login with no email, no password | Email or password is incorrect message prompt | Email or password is incorrect message prompt |

Table 10 QA coordinator test results

QA manager

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| QA manager | | | | |
| Test Case ID | Category | Condition | Expected Results | Actual Results |
| 1 | Register/Login | Register as new QA manager account | Register successful message prompt | Register successful message prompt |
| 2 | Register/Login | Register with existing QA manager account | Account already exists message prompt | Account already exists message prompt |
| 3 | Register/Login | Login with no email, no password | Email or password is incorrect message prompt | Email or password is incorrect message prompt |
| 4 | Register/Login | Login with wrong email, no password | Email or password is incorrect message prompt | Email or password is incorrect message prompt |
| 5 | Register/Login | Login with no email, wrong password | Email or password is incorrect message prompt | Email or password is incorrect message prompt |
| 6 | Register/Login | Login with wrong email, wrong password | Email or password is incorrect message prompt | Email or password is incorrect message prompt |
| 7 | Register/Login | Login with wrong email, correct password | Email or password is incorrect message prompt | Email or password is incorrect message prompt |
| 8 | Register/Login | Login with correct email, wrong password | Email or password is incorrect message prompt | Email or password is incorrect message prompt |
| 9 | Register/Login | Login with correct email and correct password | Go to select an idea page | Go to select an idea page |
| 10 | Category | Add new category | New category added into the table | New category added into the table |
| 11 | Category | Delete category | Displays in the system, but users cannot see the category that is deleted, will prompt message deleted | Displays in the system, but users cannot see the category that is deleted, will prompt message deleted |
| 12 | Category | Edit closure date | Closure date is updated | Not working |
| 13 | Category | Edit final closure date | Final closure date is updated | Not working |
| 14 | Comments | File download | Can download file uploaded by user | Download file not working |
| 15 | Navigation | Go to reports page | Go to reports page | Go to reports page |
| 16 | Statistics | View ideas per department | Can see statistics of overall ideas submit by each department | Can see statistics of overall ideas submit by each department |
| 17 | Statistics | Percentage per department | Can see overall percentage of ideas submit by each department | Can see overall percentage of ideas submit by each department |
| 18 | Statistics | Contributors per department | Can see the number of overall ideas submit by each department | Can see the number of overall ideas submit by each department |
| 19 | Exception reports | Ideas without comments | Can see total ideas submitted without comments | Can see total ideas submitted without comments |
| 20 | Exception reports | Anonymous ideas | Can see total ideas submitted by anonymous users | Can see total ideas submitted by anonymous users |
| 21 | Exception reports | Anonymous comments | Can see total number of comments given by anonymous users | Can see total number of comments given by anonymous users |
| 22 | Logout | Logout button clicked | Go back to login/registration page | Go back to login/registration page |

Table 11 QA manager test results

## 2.16 Test Plan

Below will be the test plan on how the team made a blueprint to perform software testing of the idea submit web app.

Analyse the product

For this web app, admin, quality assurance manager, quality assurance coordinator and staffs will be using the app.

To begin, admin can use the system to view people who are users of the system, add/edit/update/delete users, monitor the statistics of the system, such as the overall ideas submitted by users and view the percentage of specify ideas being uploaded. For Quality Assurance (QA) Coordinator, this type of users is able to access the system as well as manage the system. For instance, coordinator will be able to receive notifications upon ideas submitted by staffs’ users. For Quality Assurance (QA) Manager, this type of users can access system and able to oversee the process. Some of the actions consists of adding tags to ideas, delete tags if it is not used, and download file uploaded by staffs upon submitting ideas. For the last type of users, staff users can register a new account, login into to the system, add ideas with/without files, remain anonymous upon submitting an idea. Upon submitting an idea, users can select a specific category to categorized an idea.

The web app functions by allowing staffs to submit ideas of a particular category to the system. Upon submitting an idea, staff users must agree to the terms & conditions. Not only so, users can choose to remain anonymous by checking the checkbox whereby their name will be hidden upon posting the idea. If the staff users’ desires, PDF and other file types can also be included as an additional document to support the idea. In addition to adding new ideas, staffs can also view other staff users’ idea posted in the forum. This allows users to read, perform likes/dislikes and comment on the ideas. Not only so, all types of users (admin, coordinator, manager, staff) can view the statistic reports of overall user activities of the system.

After submission, the QA Coordinator will receive the notification message when login into the system. As the QA coordinator, this type of users can ban or unban other users. Therefore, if a user is banned, the ideas/comments made by that user will be hidden from other users and the banned user cannot login into the system. On the other hand, if that user is unbanned, the ideas/comments made by that user will be visible again to other users and can login into the system once more.

As for the QA manager, this type of users functions almost the same like staff user types. This means, QA manager can view other users’ contribution of ideas and view the statistics report of user activities. However, one function that the staff user type does not have is the file download function, whereby managers can download the file added by users as supporting reference of an idea.

The last type of user is the admin type. Admin user type has the most authority of the system. To start off, admin need to log in like other users. From then on, admin can view all the users who currently are using the system. Beside viewing, admin can edit/delete users from the system. Not only that, admin can also change the roles of users in the system. An example is this is if a staff user role is change to coordinator, the account with the changed role will now have the authority like that of a coordinator when login into the system.

As for the software/hardware requirements, software can be any web browsers such google chrome and Opera in order to view the web app. As for hardware for computers, the web app can be access as long as there is a working internet connection.

Design the Test Strategy

Since this system is a web app, various tools are used during development and testing. With analysis and discussion with the team, the software used will be XAMPP for database, PHP and HTML as the main programming language to develop the system and google chrome will be used to test any potential errors.

As part of the requirement of the subject, the agile methodology will be used throughout the development of the system. With agile, the progress of the system will be divided and developed based on sprints and its requirements determined by the team.

In order to effectively test the system, the team’s programmer will update team members in every meeting to ensure every member have an idea of how the system works. This allows the team tests functions of the system with lesser confusions. Not only so, the scrum master also ensures every member is motivated throughout the development to maintain the morale as well as the productivity of the team.

As for testers, each member of the group will test the functions of the system. By selecting a number of functions to work on as a sprint, functions will be tested on sprint basis as part of the requirement of the subject that uses agile methodology. Each testing phase will occur on each sprint to make sure a complete system of error free is developed (if not all of the must have requirements).

Define the Test Objectives

Before the system can be officially release to the client, the functionalities of the system are all listed down, registration/login, statistic reports, idea submissions, like/dislikes, comments, file upload/downloads, role-based functions, user management will all be taken into consideration before building test case scenarios and sample data to test the defects.

Since all the functions has been listed, testing will be done by splitting them into main and sub components. For example, in accounts function, user must be able to register, login, and logout of a system. Therefore, for every user type, all three sub components of accounts function will be tested for each user type.

Staff

Account (register, login, logout)

Ideas (view ideas, add idea, select category, add file, anonymous, likes/dislikes, comment)

Statistics (view overall statistics)

QA Manager

Account (register, login, logout)

Category (Create, update, delete)

Ideas (view ideas, likes/dislikes, comment, file download)

Statistics (view overall statistics)

QA Coordinator

Account (register, login, logout, edit user roles)

Ideas (view ideas, likes/dislikes, comment)

Statistics (view overall statistics)

Admin

Account (register, login, logout, edit user roles)

Category (Create, update, delete)

Ideas (view ideas, likes/dislikes, comment, file download)

Statistics (view overall statistics)

Define Test Criteria

For this project, suspension testing used throughout the testing phase, whereby testing is done in stages and processes for each function. If more than 40% of errors are found in the process, testing will be put to a halt. Then, programmers will begin fixing the errors based on the current tested results. Once errors are fixed, the testing will once again begin and process is repeated if errors are found again.

Resource Planning

For resources, human, equipment and materials will be considered before tests are initiated as having sufficient resources to test helps in maintaining the overall costs.

For humans, here it refers to all human users that will interact the system during the testing phase. To start the list, programmer and testers will be discussing what is needed before the testing procedure. This includes preparing sample data to use, test case tables that will be use later to filled the results, estimate possible results from the tests and others. As for system resources, some of the hardware that will be used are two computers (after testing on the first one, install the system to the other PC), a working network that allows both computers to connect to the internet.

Plan Test Environment

In this phase, team members will discuss with each other to determine the best way to test the system. For instances, the hardware and software needed, what environment is suitable to test the system and so on.

Schedule & Estimation

|  |  |  |
| --- | --- | --- |
| Task | Members | Estimated effort (days) |
| Create test plan | Ho Kai Lin, KEERTHIGAI DEVI | 8 |
| Testing | All members | 4 |
| Report | All members | 2 |
| Total |  | 14 |

Table 12 Estimated time to perform test in all sprints

Test estimation schedule

From the above table, test plans will be formed around 8 days, testing will be done with 4 days and report will need at least 2 days throughout the process of the project. Since this project utilise the agile methodology and consists of 4 sprints, testing will be done in each sprint.

Determine Test Deliverables

Before testing for each function of the system, the team programmers will brief the test cases that will be conducted on the testing phase. This means that meetings will be held to discuss each members role to test specify parts of the system. Each testing will be label with ID, category, condition, expected results and actual results to record the results obtained.

During testing, all members will be given sample data to test certain functions of the system. By using a sample data given by the programmers, expected results and actual results will be within estimated predictions.

After first round of testing, all member will compare and discuss the results obtained from the system. On the other hand, if the data tested yield many major errors, team members will re-test that particular function to verify the errors. After the second round of testing, the results will be pass to the programmer to fix the errors and the process will repeat again

## 2.17 Test Logs

Sprint 1 (17/2 to 24/2)

Test Case ID: 1

Tester: KEERTHIGAI DEVI

Test Date: 17/2/2023

Test Result: Passed

Defects: None

Test Case ID: 2

Tester: KAJENDRA NAATH RAJAN

Test Date: 17/2/2023

Test Result: Failed

Defects: Message did not prompt when email/password is missing during register

Test Case ID: 3

Tester: HO KAI LIN

Test Date: 17/2/2023

Test Result: Passed

Defects: None

Test Case ID: 4

Tester: LEE MIN QI

Test Date: 17/2/2023

Test Result: Failed

Defects: Message did not prompt when name is missing for registration

Test Case ID: 5

Tester: BRIAN EE CHI CHIN

Test Date: 17/2/2023

Test Result: Passed

Defects: None

Test Case ID: 6

Tester: KEERTHIGAI DEVI

Test Date: 17/2/2023

Test Result: Failed

Defects: Message did not prompt when user’s email is missing during registration

Test Case ID: 7

Tester: KAJENDRA NAATH RAJAN

Test Date: 17/2/2023

Test Result: Passed

Defects: None

Test Case ID: 8

Tester: HO KAI LIN

Test Date: 18/2/2023

Test Result: Failed

Defects: Message did not prompt when user did not select a role during registration

Test Case ID: 9

Tester: LEE MIN QI

Test Date: 18/2/2023

Test Result: Passed

Defects: None

Test Case ID: 10

Tester: BRIAN EE CHI CHIN

Test Date: 18/2/2023

Test Result: Passed

Defects: None

Test Case ID: 11

Tester: KEERTHIGAI DEVI

Test Date: 17/2/2023

Test Result: Failed

Defects: Message did not prompt when user’s email is missing during registration

Test Case ID: 12

Tester: KAJENDRA NAATH RAJAN

Test Date: 17/2/2023

Test Result: Passed

Defects: None

Test Case ID: 13

Tester: HO KAI LIN

Test Date: 18/2/2023

Test Result: Failed

Defects: Message did not prompt when user did not select a role during registration

Test Case ID: 14

Tester: LEE MIN QI

Test Date: 18/2/2023

Test Result: Passed

Defects: None

Test Case ID: 15

Tester: BRIAN EE CHI CHIN

Test Date: 21/2/2023

Test Result: Passed

Defects: None

Test Case ID: 16

Tester: KEERTHIGAI DEVI

Test Date: 22/2/2023

Test Result: Failed

Defects: User login into system despite wrong email

Test Case ID: 17

Tester: KAJENDRA NAATH RAJAN

Test Date: 23/2/2023

Test Result: Passed

Defects: None

Test Case ID: 18

Tester: HO KAI LIN

Test Date: 23/2/2023

Test Result: Failed

Defects: User login into system despite wrong password

Test Case ID: 19

Tester: LEE MIN QI

Test Date: 24/2/2023

Test Result: Passed

Defects: None

Test Case ID: 20

Tester: BRIAN EE CHI CHIN

Test Date: 24/2/2023

Test Result: Passed

Defects: None

Sprint 2 (3/3/23 to 9/3)

Test Case ID: 21

Tester: KEERTHIGAI DEVI

Test Date: 3/3/2023

Test Result: Failed

Defects: Navigation missing for staff

Test Case ID: 22

Tester: KAJENDRA NAATH RAJAN

Test Date: 3/3/2023

Test Result: Passed

Defects: None

Test Case ID: 23

Tester: HO KAI LIN

Test Date: 3/32023

Test Result: Passed

Defects: None

Test Case ID: 24

Tester: LEE MIN QI

Test Date: 3/3/2023

Test Result: Failed

Defects: Add ideas button has some errors

Test Case ID: 25

Tester: BRIAN EE CHI CHIN

Test Date: 3/3/2023

Test Result: Passed

Defects: None

Test Case ID: 26

Tester: KEERTHIGAI DEVI

Test Date: 4/3/2023

Test Result: Failed

Defects: Idea did not display in the forum

Test Case ID: 27

Tester: KAJENDRA NAATH RAJAN

Test Date: 4/3/2023

Test Result: Passed

Defects: None

Test Case ID: 28

Tester: HO KAI LIN

Test Date: 4/3/2023

Test Result: Failed

Defects: Cannot upload file

Test Case ID: 29

Tester: LEE MIN QI

Test Date: 4/3/2023

Test Result: Passed

Defects: None

Test Case ID: 30

Tester: BRIAN EE CHI CHIN

Test Date: 5/3/2023

Test Result: Failed

Defects: User’s name still displays despite anonymous is selected

Test Case ID: 31

Tester: KEERTHIGAI DEVI

Test Date: 6/3/2023

Test Result: Passed

Defects: None

Test Case ID: 32

Tester: KAJENDRA NAATH RAJAN

Test Date: 6/3/2023

Test Result: Failed

Defects: Another user’s like did not add to the count

Test Case ID: 33

Tester: HO KAI LIN

Test Date: 6/3/2023

Test Result: Passed

Defects: None

Test Case ID: 34

Tester: LEE MIN QI

Test Date: 7/3/2023

Test Result: Failed

Defects: Another user’s dislike did not add to the count

Test Case ID: 35

Tester: BRIAN EE CHI CHIN

Test Date: 7/3/2023

Test Result: Passed

Defects: None

Test Case ID: 36

Tester: KEERTHIGAI DEVI

Test Date: 8/3/2023

Test Result: Failed

Defects: Another user’s comment did not display under the idea of another us

Test Case ID: 37

Tester: KAJENDRA NAATH RAJAN

Test Date: 8/3/2023

Test Result: Passed

Defects: None

Test Case ID: 38

Tester: HO KAI LIN

Test Date: 8/3/2023

Test Result: Failed

Defects: File download button did not appear after the idea submit

Test Case ID: 39

Tester: LEE MIN QI

Test Date: 9/3/2023

Test Result: Passed

Defects: None

Test Case ID: 40

Tester: BRIAN EE CHI CHIN

Test Date: 9/3/2023

Test Result: Passed

Defects: None

Sprint 3 (17/3/23 to 23/3)

Test Case ID: 41

Tester: KEERTHIGAI DEVI

Test Date: 17/3/2023

Test Result: Failed

Defects: Navigation exclusive to admin is missing

Test Case ID: 42

Tester: KAJENDRA NAATH RAJAN

Test Date: 17/3/2023

Test Result: Failed

Defects: Coordinator cannot edit roles

Test Case ID: 43

Tester: HO KAI LIN

Test Date: 17/3/2023

Test Result: Failed

Defects: Admin functionalities can be access as coordinator

Test Case ID: 44

Tester: LEE MIN QI

Test Date: 17/3/2023

Test Result: Passed

Defects: None

Test Case ID: 45

Tester: BRIAN EE CHI CHIN

Test Date: 17/3/2023

Test Result: Passed

Defects: None

Test Case ID: 46

Tester: KEERTHIGAI DEVI

Test Date: 17/10/2023

Test Result: Failed

Defects: Management could not add category

Test Case ID: 47

Tester: KAJENDRA NAATH RAJAN

Test Date: 17/10/2023

Test Result: Failed

Defects: Management could not delete category

Test Case ID: 48

Tester: HO KAI LIN

Test Date: 18/3/2023

Test Result: Passed

Defects: None

Test Case ID: 49

Tester: LEE MIN QI

Test Date: 18/3/2023

Test Result: Failed

Defects: Admin unable to edit user roles

Test Case ID: 50

Tester: BRIAN EE CHI CHIN

Test Date: 18/3/2023

Test Result: Passed

Defects: None

Test Case ID: 51

Tester: KEERTHIGAI DEVI

Test Date: 18/3/2023

Test Result: Passed

Defects: None

Test Case ID: 52

Tester: KAJENDRA NAATH RAJAN

Test Date: 18/3/2023

Test Result: Failed

Defects: Users banned by admin could still access the system

Test Case ID: 53

Tester: HO KAI LIN

Test Date: 18/3/2023

Test Result: Failed

Defects: Users banned by coordinator could still access the system

Test Case ID: 54

Tester: LEE MIN QI

Test Date: 20/3/2023

Test Result: Passed

Defects: None

Test Case ID: 55

Tester: BRIAN EE CHI CHIN

Test Date: 20/3/2023

Test Result: Passed

Defects: None

Test Case ID: 56

Tester: KEERTHIGAI DEVI

Test Date: 21/3/2023

Test Result: Failed

Defects: Manager could not download file uploaded by user in the system

Test Case ID: 57

Tester: KAJENDRA NAATH RAJAN

Test Date: 21/3/2023

Test Result: Failed

Defects: Admin could not navigate to statistic reports page

Test Case ID: 58

Tester: HO KAI LIN

Test Date: 22/3/2023

Test Result: Passed

Defects: None

Test Case ID: 59

Tester: LEE MIN QI

Test Date: 22/3/2023

Test Result: Passed

Defects: None

Test Case ID: 60

Tester: BRIAN EE CHI CHIN

Test Date: 22/3/2023

Test Result: Passed

Defects: None

Sprint 4 (31/3/23 to 6/4)

Test Case ID: 61

Tester: KEERTHIGAI DEVI

Test Date: 31/3/2023

Test Result: Passed

Defects: None

Test Case ID: 62

Tester: KAJENDRA NAATH RAJAN

Test Date: 31/3/2023

Test Result: Failed

Defects: Statistic report did not update count despite new ideas posted by staff

Test Case ID: 63

Tester: HO KAI LIN

Test Date: 31/3/2023

Test Result: Passed

Defects: None

Test Case ID: 64

Tester: LEE MIN QI

Test Date: 31/3/2023

Test Result: Failed

Defects: Coordinator could not access anonymous statistic report page

Test Case ID: 65

Tester: BRIAN EE CHI CHIN

Test Date: 31/3/2023

Test Result: Passed

Defects: None

Test Case ID: 66

Tester: KEERTHIGAI DEVI

Test Date: 1/4/2023

Test Result: Passed

Defects: None

Test Case ID: 67

Tester: KAJENDRA NAATH RAJAN

Test Date: 2/4/2023

Test Result: Failed

Defects: User role who are change to manager can banned/unbanned other users

Test Case ID: 68

Tester: HO KAI LIN

Test Date: 2/4/2023

Test Result: Passed

Defects: None

Test Case ID: 69

Tester: LEE MIN QI

Test Date: 2/4/2023

Test Result: Failed

Defects: Category can still be selected despite closure date has pass

Test Case ID: 70

Tester: BRIAN EE CHI CHIN

Test Date: 2/4/2023

Test Result: Passed

Defects: None

Test Case ID: 71

Tester: KEERTHIGAI DEVI

Test Date: 3/4/2023

Test Result: Passed

Defects: None

Test Case ID: 72

Tester: KAJENDRA NAATH RAJAN

Test Date: 3/4/2023

Test Result: Failed

Defects: Ideas did not appear despite user is unbanned by admin

Test Case ID: 73

Tester: HO KAI LIN

Test Date: 3/4/2023

Test Result: Failed

Defects: Comments did not appear despite user is unbanned by coordinator

Test Case ID: 74

Tester: LEE MIN QI

Test Date: 3/4/2023

Test Result: Failed

Defects: Idea still exists despite user banned by admin/coordinator

Test Case ID: 75

Tester: BRIAN EE CHI CHIN

Test Date: 3/4/2023

Test Result: Failed

Defects: Comments still exists despite user banned by admin

Test Case ID: 76

Tester: KEERTHIGAI DEVI

Test Date: 5/4/2023

Test Result: Passed

Defects: None

Test Case ID: 77

Tester: KAJENDRA NAATH RAJAN

Test Date: 5/4/2023

Test Result: Passed

Defects: None

Test Case ID: 78

Tester: HO KAI LIN

Test Date: 5/4/2023

Test Result: Passed

Defects: None

Test Case ID: 79

Tester: LEE MIN QI

Test Date: 6/4/2023

Test Result: Passed

Defects: None

Test Case ID: 80

Tester: BRIAN EE CHI CHIN

Test Date: 6/4/2023

Test Result: Passed

Defects: None

# 3.0 Screenshots and Explanation

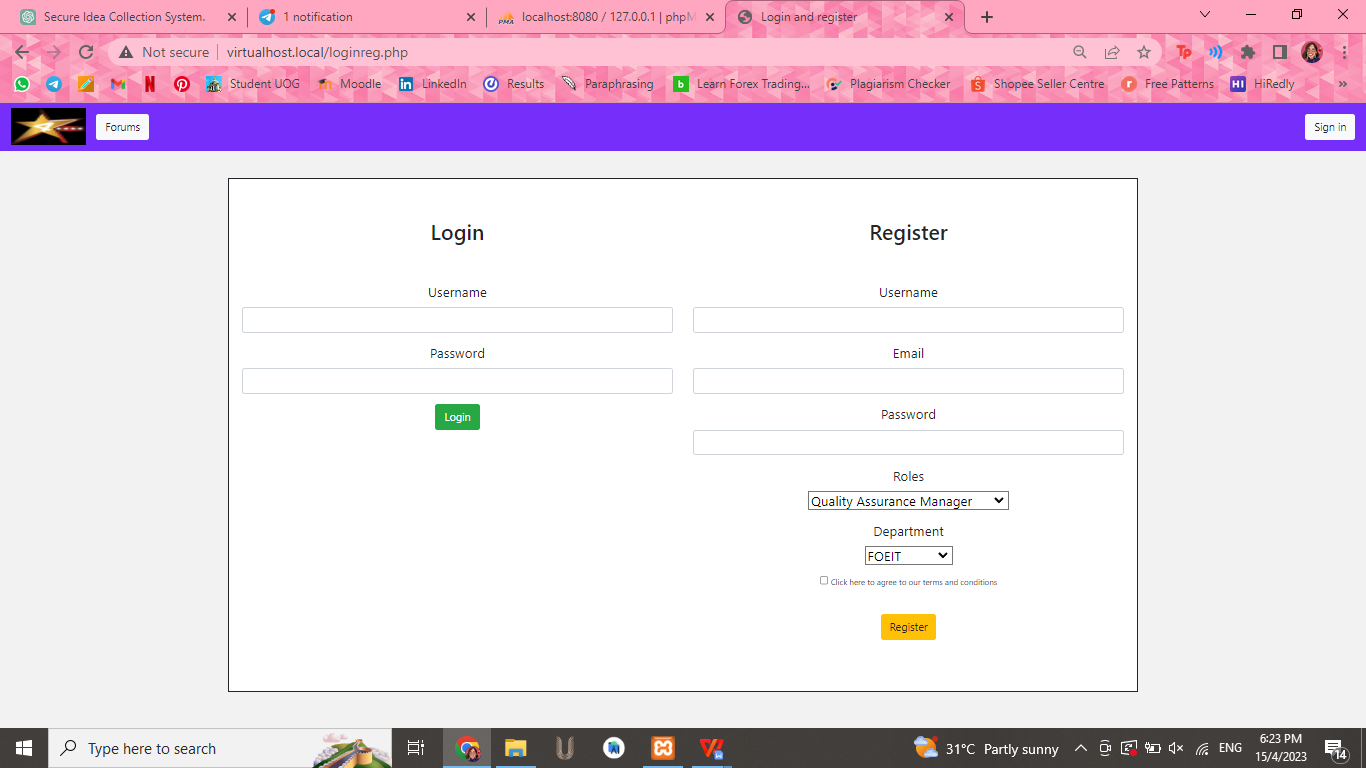


Figure 3.1 Login and registration page

The login page includes fields for users to enter their login credentials, such as their username and password. The registration page allows admins to create new users an account and obtain login credentials. Users are typically required to provide their personal information, such as their username, email address, and job title, department and choose a unique and password. Overall, the login and registration page play a critical role in ensuring the security and integrity of the system, while providing a user-friendly and efficient way for staff members to access the system and contribute their ideas for improvement.

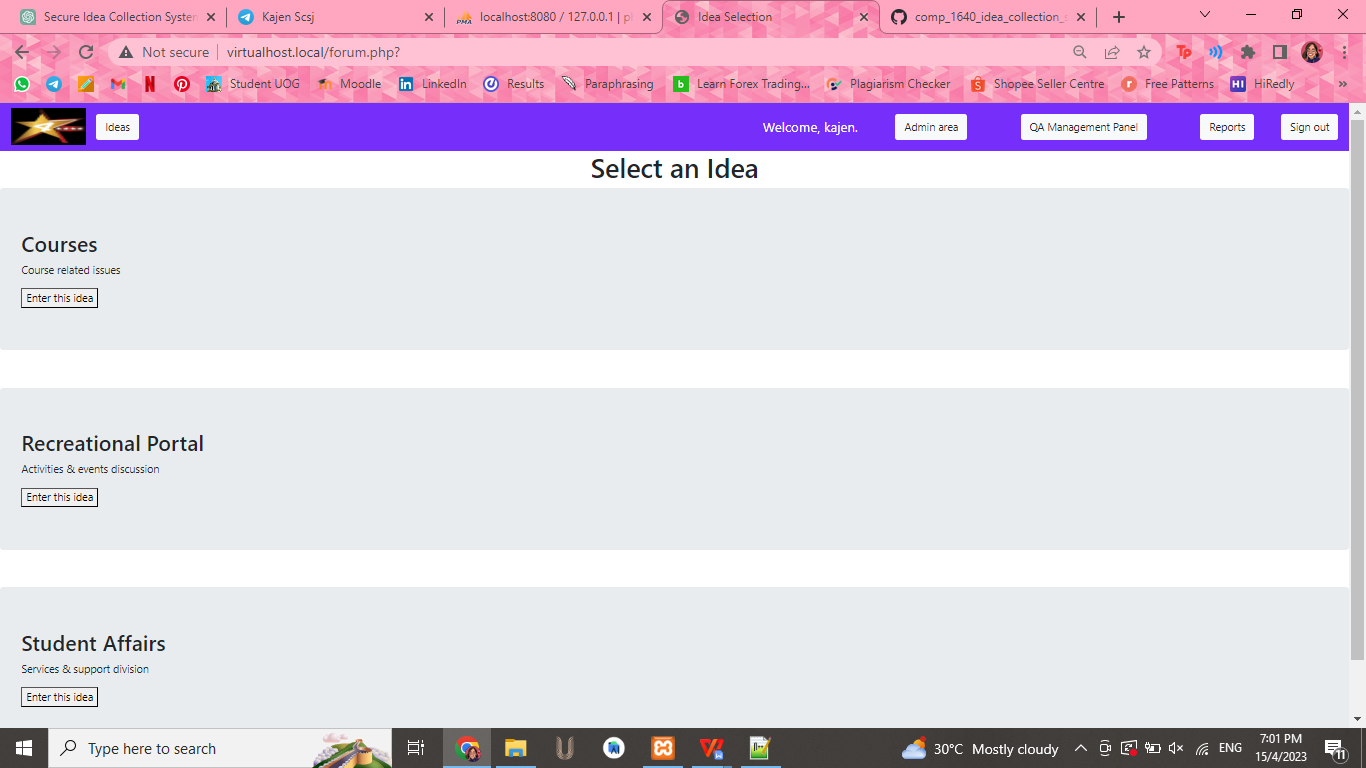


Figure 3.2 Select an Idea page

The "Select an Idea" page of a secure web-enabled role-based system for collecting ideas for improvement from staff in a large university is a central component of the system that enables users to browse and select from a list of ideas. From the "Select an Idea" page, authorized users can select an idea to view its details, including a more in-depth description of the idea, any supporting documents or files, and any comments or feedback from other users. Users can also vote on an idea to express their support and indicate its popularity among other staff members. In a secure web-enabled role-based system, the "Select an Idea" page is designed to ensure that users can easily find and select relevant ideas while maintaining the system's security and data protection capabilities.

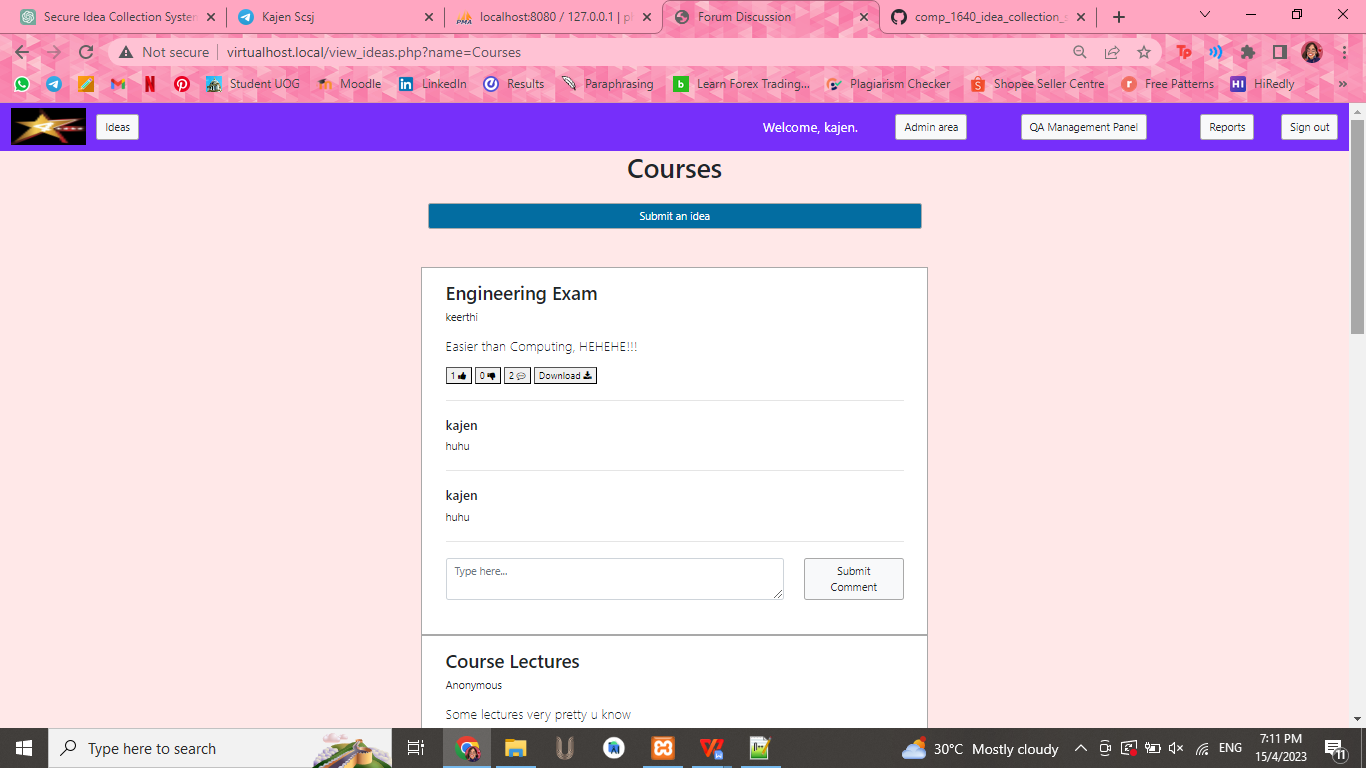


Figure 3.3 View ideas page

The view ideas page of a secure web-enabled role-based system for collecting ideas for improvement from staff in a large University is an essential component of the system that enables authorized users to view, comment, like and dislike to submitted ideas. The page is designed to provide a user-friendly interface that displays all ideas submitted by staff members. The page can be accessed by users who have the appropriate role-based access privileges, such as administrators or idea reviewers. To ensure the security of the system, the view ideas page should only display ideas that are accessible to the user based on their role-based access privileges. In addition to viewing submitted ideas, the view ideas page should also provide users with tools to manage and evaluate the ideas. For example, users with appropriate privileges may be able to provide feedback to submitter, or like and dislike to relevant departments or individuals for further review.

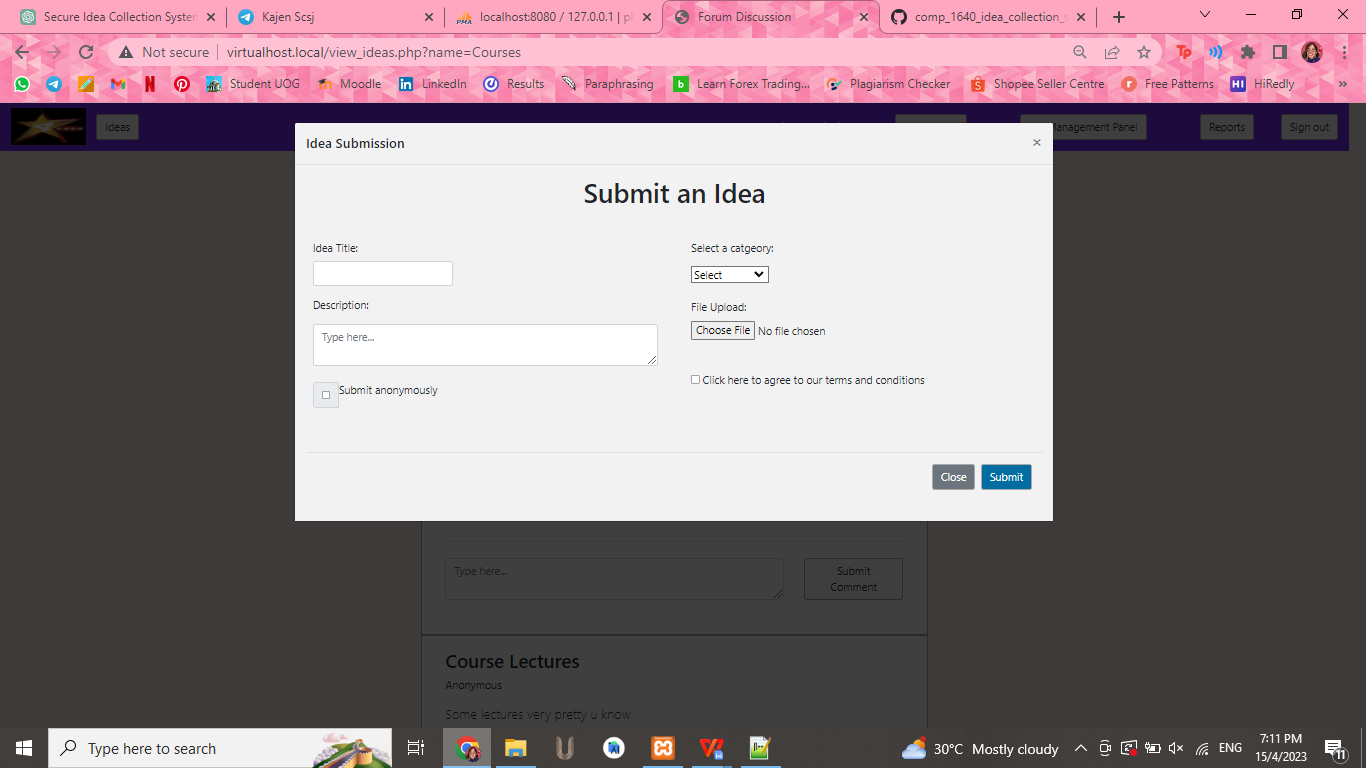


Figure 3.4: Submit an idea page

The submit idea feature is designed to be easy to use and accessible to all staff members who have access to the system. It typically includes a form where staff members can provide details about their idea, including the title, description, and any supporting documents or files. The form may also include fields for identifying the department or unit that the idea pertains to, the potential benefits or impact of the idea, and any challenges or risks that may need to be addressed. To ensure the security of the system, the submit idea feature should only be accessible to authorized users who have been granted the appropriate role-based access privileges. This means that only staff members who are authorized to submit ideas can access the feature, and they can only submit ideas as anonymously if they wish. Before submitting their ideas, the submitter must agree to the terms and conditions.

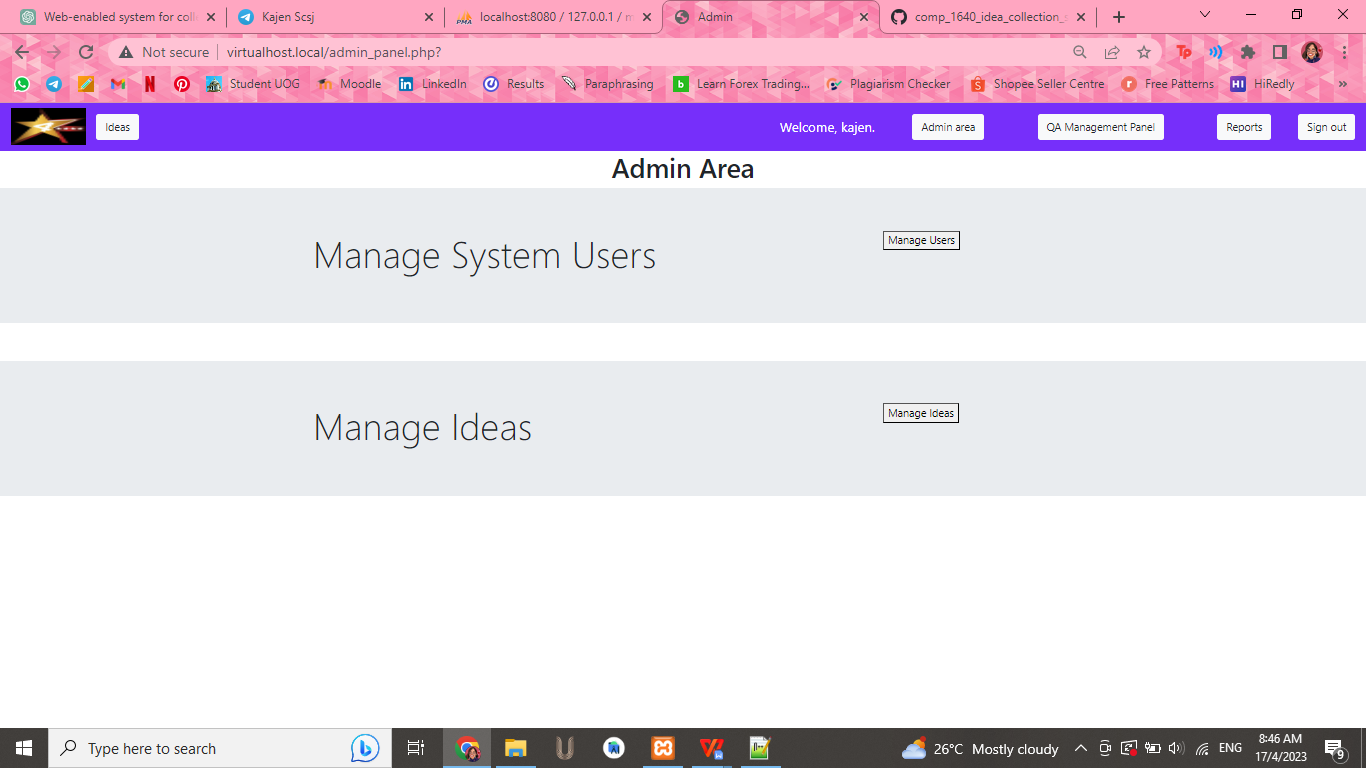


Figure 3.5 Admin Page

The admin area of a secure web-enabled role-based system for collecting ideas for improvement from staff in a large University is a critical component of the system that enables administrators to manage user accounts and ideas. The admin area typically includes two main features: manage users and manage ideas. These features enable administrators to perform tasks such as modifying user permissions, approving or rejecting ideas, and assigning ideas to relevant departments or individuals for further review. To ensure the security of the system, the admin area should only be accessible to authorized administrators who have been granted the appropriate role-based access privileges. This means that only users with administrator privileges can access the admin area and perform tasks such as managing user accounts and ideas.

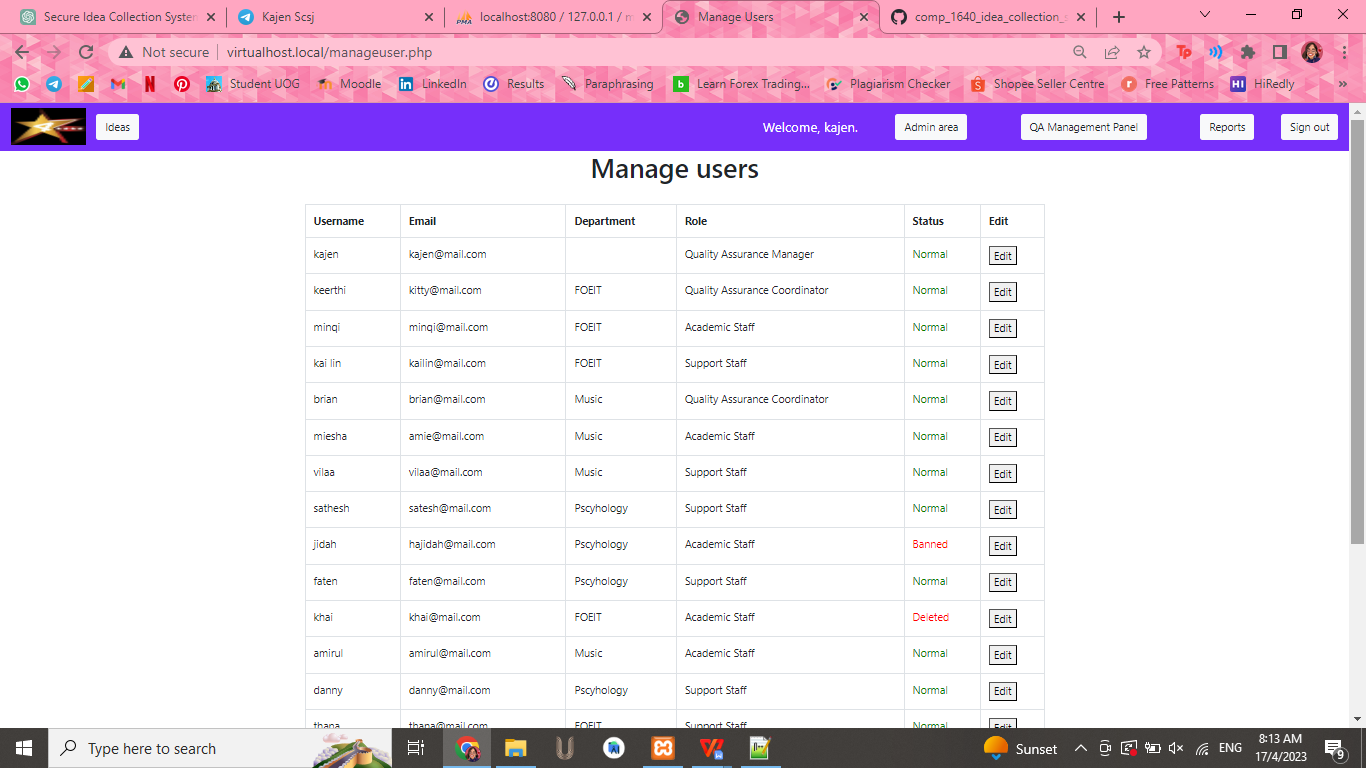


Figure 3.6: Manage User Page

The manage users feature allows administrators to edit, and remove user accounts in the system. This includes updating user profiles and revoking user access when necessary. The manage users feature may also allow administrators to assign role-based access privileges to users, which determine what features and functionality users can access within the system. The manage users feature allows administrators to edit, and remove user accounts in the system. The manage users feature may also allow administrators to assign role-based access privileges to users, which determine what features and functionality users can access within the system.

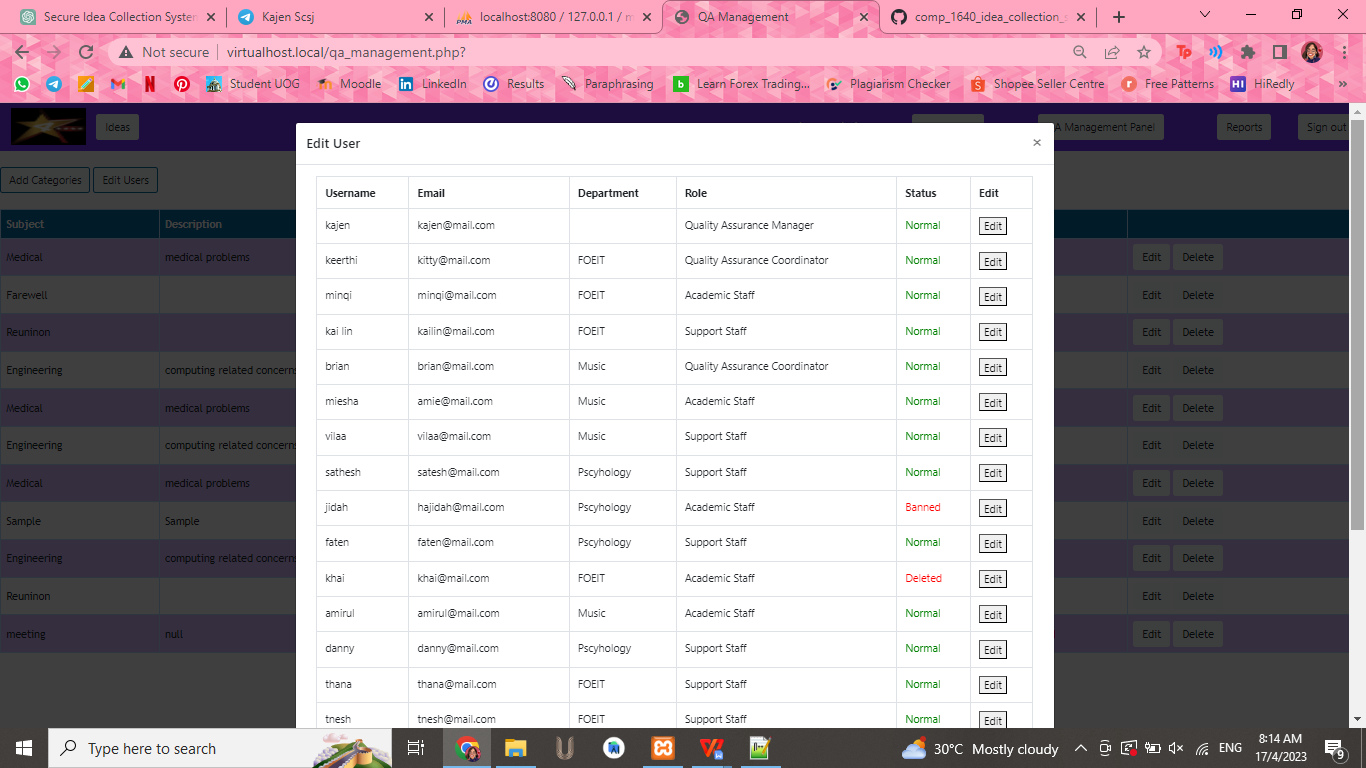


Figure 3.7: Edit User Page

The edit user page in a secure web-enabled role-based system for collecting ideas for improvement from staff in a large University is a critical component of the system that enables Quality Assurance (QA) coordinators to manage user accounts. In particular, the edit user page provides QA coordinators with the ability to ban or unban users from the system. Banning a user typically means revoking their access to the system and preventing them from submitting new ideas or comments. Banning a user can be necessary in situations where the user has violated the system's code of conduct, submitted inappropriate content, or engaged in other activities that are deemed to be in violation of the system's policies. Unbanning a user means restoring their access to the system after they have been banned. Unbanning a user can be necessary in situations where the user has rectified the issue that led to their ban or if the ban was applied in error.

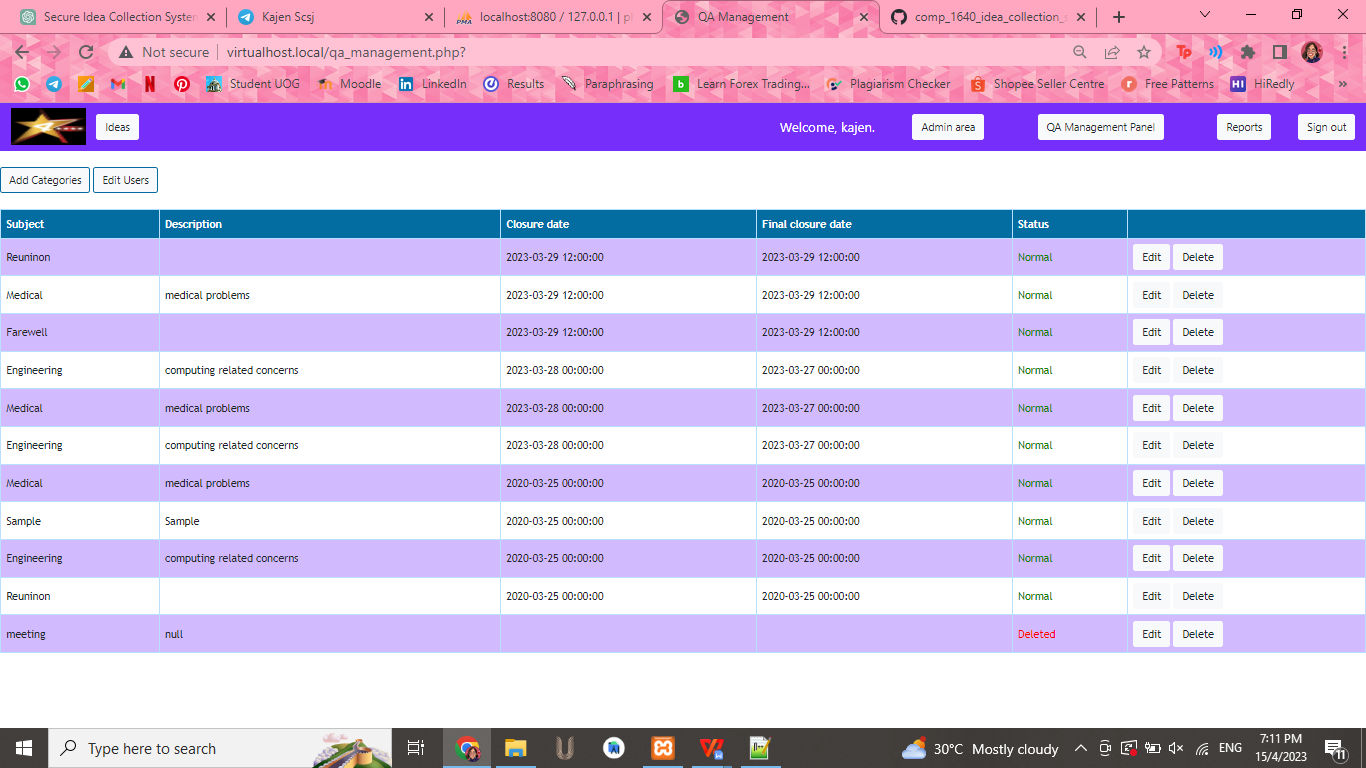


Figure 3.8: Add Categories Page

The add categories page in a secure web-enabled role-based system for collecting ideas for improvement from staff in a large University is a critical component of the system that enables Quality Assurance (QA) coordinators to manage categories for idea submission. In particular, the add categories page provides QA coordinators with the ability to add new categories, set closure dates for existing categories, and manage the closure of categories. Categories are an important organizational tool in the idea submission process, allowing ideas to be grouped by topic, department, or other relevant criteria. The add categories page typically includes options for QA coordinators to create new categories and specify the closure date for each category. The closure date is the date after which ideas can no longer be submitted to a particular category. To ensure the security of the system, the add categories page should only be accessible to authorized users who have been granted the appropriate role-based access privileges. This means that only QA coordinators or administrators with specific permissions can access the add categories page and perform tasks such as adding new categories or managing category closures.

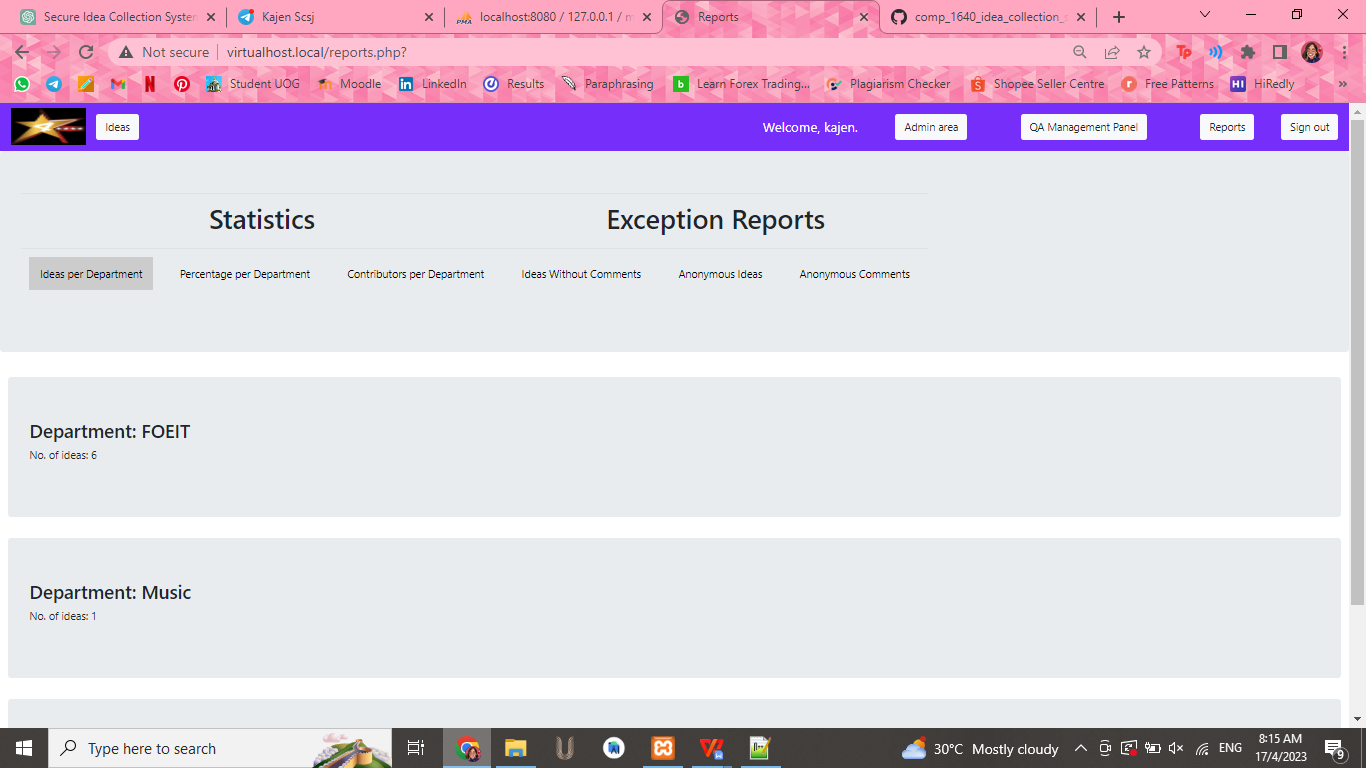


Figure 3.9: Reports Page

The report page of a secure web-enabled role-based system for collecting ideas for improvement from staff in a large University is a critical component of the system that enables staffs, admin, QA manager and QA coordinator to view overall statistical information about comments and ideas submitted to the system. The report page that provides an overview of the number of ideas submitted per department, percentage per department, contribution per department, ideas without comments and the number of comments and anonymous comments received. The report page may also provide information on the types of ideas submitted, the departments that submitted them, and the overall of implemented ideas.

# 4.0 Conclusion

In conclusion, the secure web-enabled role-based system we have proposed for collecting ideas for improvement from staff in a large university has the potential to significantly enhance the effectiveness and efficiency of the university's operations. By leveraging modern web technologies and employing role-based access control mechanisms, the system provides a user-friendly and secure platform for staff members to submit their ideas, while ensuring that only authorized personnel can access and manage the collected data. With the ability to track and monitor user activities, the system also enables administrators to identify potential issues or threats and take appropriate measures to address them. Overall, the proposed system represents a significant step forward in streamlining the process of idea collection and management at large universities. By providing a secure, web-enabled platform for staff members to submit their ideas, the system empowers the university to tap into the full potential of its staff and drive continuous improvement across all areas of operation.