NANYANG TECHNOLOGICAL UNIVERSITY
SINGAPORE

SC2006/CZ2007 SOFTWARE ENGINEERING

Semester 2 23/24

Lab 4: Implementation, Testing, and Preparation for Demo

Team Name: Seow's Team
Project Name: Kanbanize
Lab Group: SCSG

GROUP MEMBERS	LAB GROUP	SIGNATURE
Augustine Jesuraj Senchia Gladine	SCSG	
Samuel Tan	SCSG	
Seet Tze Shin, Cheyenne	SCSG	
Seow Ming Han Samuel	SCSG	
Willy Tang Jing Lin	SCSG	

Deliverables for Lab 4

- ☑ Working application prototype
- ☑ Source code
- ☑ Test Cases and Testing Results
- ☑ Demo script

Purpose, Outcome and Benefits

Kanbanize is a website that enables its users to perform seamless work management through the use of a Kanban board-style interface. Users can customize their workflows, tasks and permissions within their workspace. Our purpose is to visualize workflow and maximize efficiency for work management.

Intended Audience

Our target stakeholders are all types of teams and organizations for school projects and business project management. Our Kanban board is easy to use and entirely customizable to each user's or team's preferences.

Source Code

Refer to https://github.com/softwarelab3/2006-SCSG-Seow-s-Team for source code.

Table of Contents

Test Cases and Testing Results

Black Box Testing

1. Registration

Location	Login Interface → Register Interface

a. General cases

ID	Input	Oracle	Log
#1	Register with valid username, email address and password	System redirects to Project List Page.	System redirects to Project List Page.
#2	Register without populating username, email address, password and/or confirm password fields	System displays error message, “Fill up all the fields!” prompting user.	System displays error message, “Fill up all the fields!” prompting user.
#3	Register with invalid email address format	System displays error message, “Enter a valid email address!” prompting user to re-enter email address.	System displays error message, “Enter a valid email address!” prompting user to re-enter email address.
#4	Register with an existing email address	System displays error message “email address already taken!” prompting user to change their username.	System displays error message “email address already taken!” prompting user to change their username.
#5	Registers with an existing username	System displays error message “username already taken!” prompting user to change their username.	System displays error message “username already taken!” prompting user to change their username.
#6	Registers without meeting password criteria	System displays error message “ Password not strong enough! Password should include numbers, uppercase and	System displays error message “Password not strong enough! Password should include numbers, uppercase and

		lowercase letters and should have at least 8 characters.” prompting user to choose another password that follows the criteria.	lowercase letters and should have at least 8 characters.” prompting user to choose another password that follows the criteria.
#7	Registers with different inputs in “Password” and “Confirm Password” fields	System displays error message, “Passwords do not match!” prompting user to recheck.	System displays error message, “Passwords do not match!” prompting user to recheck.

b. Specific cases

ID	Username	Email	Password	Confirm Password	Expected Output	Test Result
#1	{Empty}	{Empty}	{Empty}	{Empty}	Error! Fill up all the fields!	Error! Fill up all the fields!
#2	{Empty}	username@gmail.com	P@ssw0rd	P@ssw0rd	Error! Fill up all the fields!	Error! Fill up all the fields!
#3	username	{Empty}	P@ssw0rd	P@ssw0rd	Error! Fill up all the fields!	Error! Fill up all the fields!
#4	username	username@gmail.com	{Empty}	P@ssw0rd	Error! Fill up all the fields!	Error! Fill up all the fields!
#5	username	username@gmail.com	P@ssw0rd	{Empty}	Error! Fill up all the fields!	Error! Fill up all the fields!
#6	username	user@gmail	P@ssw0rd	P@ssw0rd	Error! Enter a valid email address!	Error! Enter a valid email address!
#7	registeredUsername	username@gmail.com	P@ssw0rd	P@ssw0rd	Error username already taken!	Error username already taken!
#8	username	registeredEmail@gmail.com	P@ssw0rd	P@ssw0rd	Error! Email address already taken!	Error! Email address already taken!
#9	username	username@gmail.com	password	password	Error! Password not strong enough!	Error! Password not strong enough!

					Password should include numbers, uppercase and lowercase letters and should have at least 8 characters.	Password should include numbers, uppercase and lowercase letters and should have at least 8 characters.
#10	username	username@gmail.com	P@ssw0rd	P@ssw0rd	System redirects to Project List Page.	System redirects to Project List Page.
#11	username	username@gmail.com	P@ssw0rd	password	Error! Passwords do not match!	Error! Passwords do not match!

2. Login

Location	Login Interface

a. General cases

ID	Input	Oracle	Log
#1	Login with registered username and valid password	System redirects to Project List Page.	System redirects to Project List Page.
#2	Login with unregistered username	System displays error message “There was a problem with your login. Please check your username and password and try again.” prompting user to recheck their login credentials.	System displays error message “There was a problem with your login. Please check your username and password and try again.” prompting user to recheck their login credentials.
#3	Login with registered username but invalid password	System displays error message “There was a problem with your login. Please check your username and password and try again.” prompting user to recheck their login credentials.	System displays error message “There was a problem with your login. Please check your username and password and try again.” prompting user to recheck their login credentials.
#4	Login with invalid username and password	System displays error message “There was a problem with your login. Please check your username and password and try again.” prompting user to recheck their login credentials.	System displays error message “There was a problem with your login. Please check your username and password and try again.” prompting user to recheck their login credentials.
#5	Login with username or/and password field(s) unpopulated	System displays error message, “Fill up all the fields!” prompting user.	System displays error message, “Fill up all the fields!” prompting user.

b. Specific cases

ID	Username	Password	Expected Output	Test Result
#1	registered username	P@ssw0rd	System redirects to Project List Page.	System redirects to Project List Page.

#2	unregistered	P@ssw0rd	Error! There was a problem with your login. Please check your username and password and try again	Error! There was a problem with your login. Please check your username and password and try again
#3	registered username	password	Error! There was a problem with your login. Please check your username and password and try again	Error! There was a problem with your login. Please check your username and password and try again
#4	invalid username	password	Error! There was a problem with your login. Please check your username and password and try again	Error! There was a problem with your login. Please check your username and password and try again
#5	{Empty}	P@ssw0rd	Error! Fill up all the fields!	Error! Fill up all the fields!
#6	registered username	{Empty}	Error! Fill up all the fields!	Error! Fill up all the fields!
#7	{Empty}	{Empty}	Error! Fill up all the fields!	Error! Fill up all the fields!

3. Forget Password

Location	Register Interface → Forgot Password Interface → Login Interface

a. General cases

ID	**Input**	**Oracle**	**Log**
#1	Change password with valid password	System displays password reset message and prompts user to return to login page.	System displays password reset message and prompts user to return to login page.
#2	Change password without meeting password criteria	System displays error message “Password not strong enough! Password should include numbers, uppercase and lowercase letters and should have at least 8 characters.” prompting user to choose another password that follows the criteria.	System displays error message “Password not strong enough! Password should include numbers, uppercase and lowercase letters and should have at least 8 characters.” prompting user to choose another password that follows the criteria.

b. Specific cases

ID	**Password**	**Confirm Password**	**Expected Output**	**Test Result**
#1	P@ssw0rd	P@ssw0rd	Password Reset! Your password has been successfully reset. Please log in with your updated credentials.	Password Reset! Your password has been successfully reset. Please log in with your updated credentials.
#2	password	password	Error! Password not strong enough! Password should include numbers, uppercase and lowercase letters and should have at least 8 characters.	Error! Password not strong enough! Password should include numbers, uppercase and lowercase letters and should have at least 8 characters.
#3	P@ssw0rd	password	Error! Passwords do not match!	Error! Passwords do not match!

#4	{Empty}	{Empty}	Error! “Password not strong enough! Password should include numbers, uppercase and lowercase letters and should have at least 8 characters.”	Error! “Password not strong enough! Password should include numbers, uppercase and lowercase letters and should have at least 8 characters.”

4. Create New Project

Location	Project List Page

a. General cases

ID	Input	Oracle	Log
#1	Create a project with name	System creates a new Project and adds it to the list of projects.	Creates a new Project and adds it to the list of projects.
#2	Create a project with {Empty} name field	System displays error message, “Fill up all the fields!” prompting user.	System displays error message, “Fill up all the fields!” prompting user.

b. Specific cases

ID	Project Name	Expected Output	Test Result
#1	Project1	Project “Project1” gets added to the project list.	Project “Project1” gets added to the project list.
#2	{Empty}	Error! Input a title!	Error! Input a title!

5. Create New Task Card

Location	Kanban View Page

a. General cases

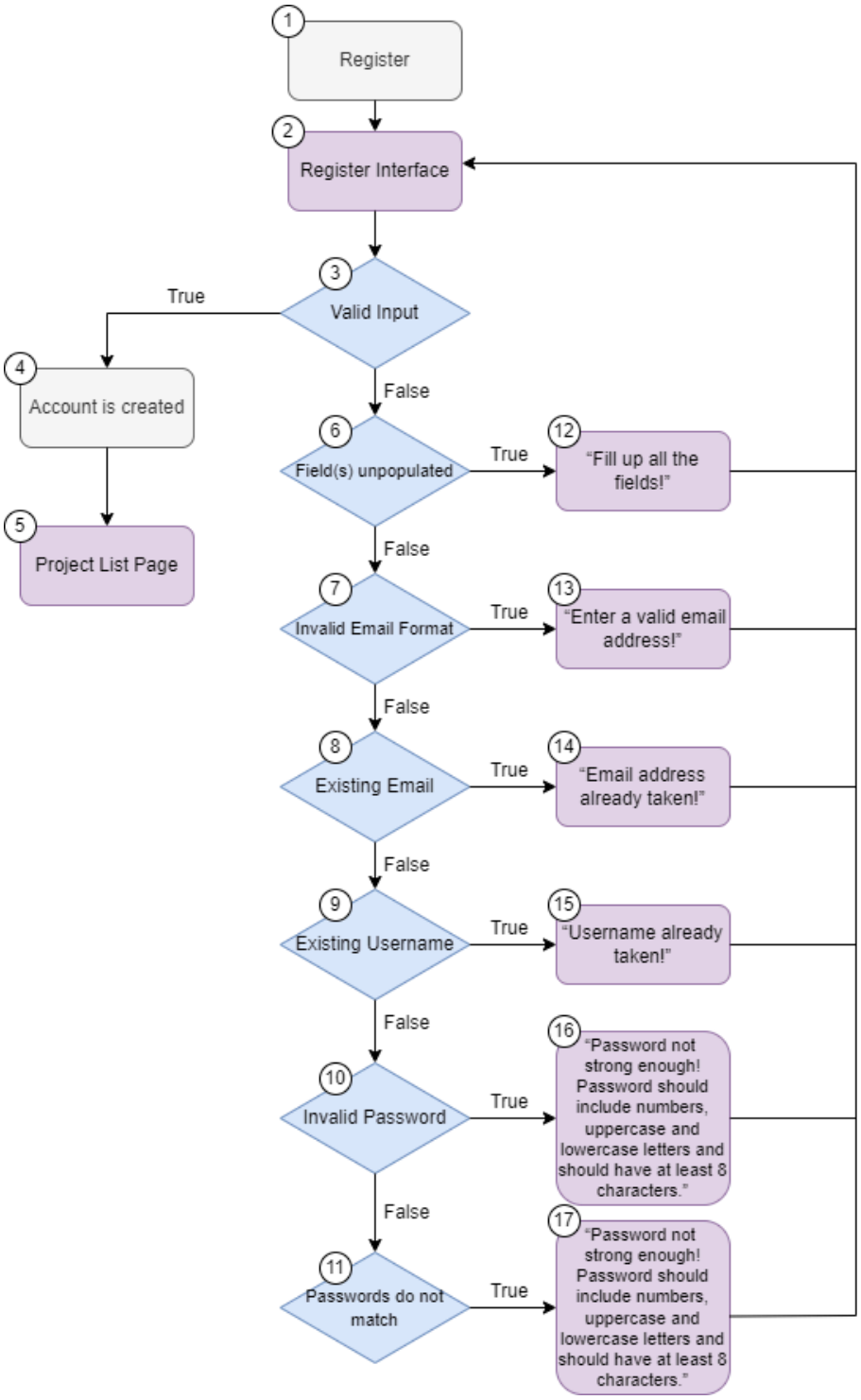
ID	Input	Oracle	Log
#1	Create a task in the preferred column	System creates a new task and adds the task to the preferred column.	System creates a new task and adds the task to the preferred column.
#2	Create a task with {Empty} title, description and date	System displays error message, “Input a title!” and prompts user.	System displays error message, “Input a title!” and prompts user.
#3	Create a task with title, description and {Empty} date	System displays error message, “Input a date between today and 2 years later!” and prompts user.	System displays error message, “Input a date between today and 2 years later!” and prompts user.

b. Specific cases

ID	Task Title	Task Description	Task Due Date	Expected Output	Test Result
#1	Task 1	{Empty}	16/4/2024 (Current date)	Task gets added to preferred column.	Task gets added to preferred column.
#2	{Empty}	abc	16/4/2024	Error! Input a title!	Error! Input a title!
#3	Task 1	abc	16/4/2024	Task gets added to preferred column.	Task gets added to preferred column.
#4	Task 1	abc	10/4/2024 (date before current date)	Error! Input a date between today and 2 years later!	Error! Input a date between today and 2 years later!
#5	Task 1	abc	17/4/2026 (date beyond 2 years from current date)	Error! Input a date between today and 2 years later!	Error! Input a date between today and 2 years later!

White Box Testing

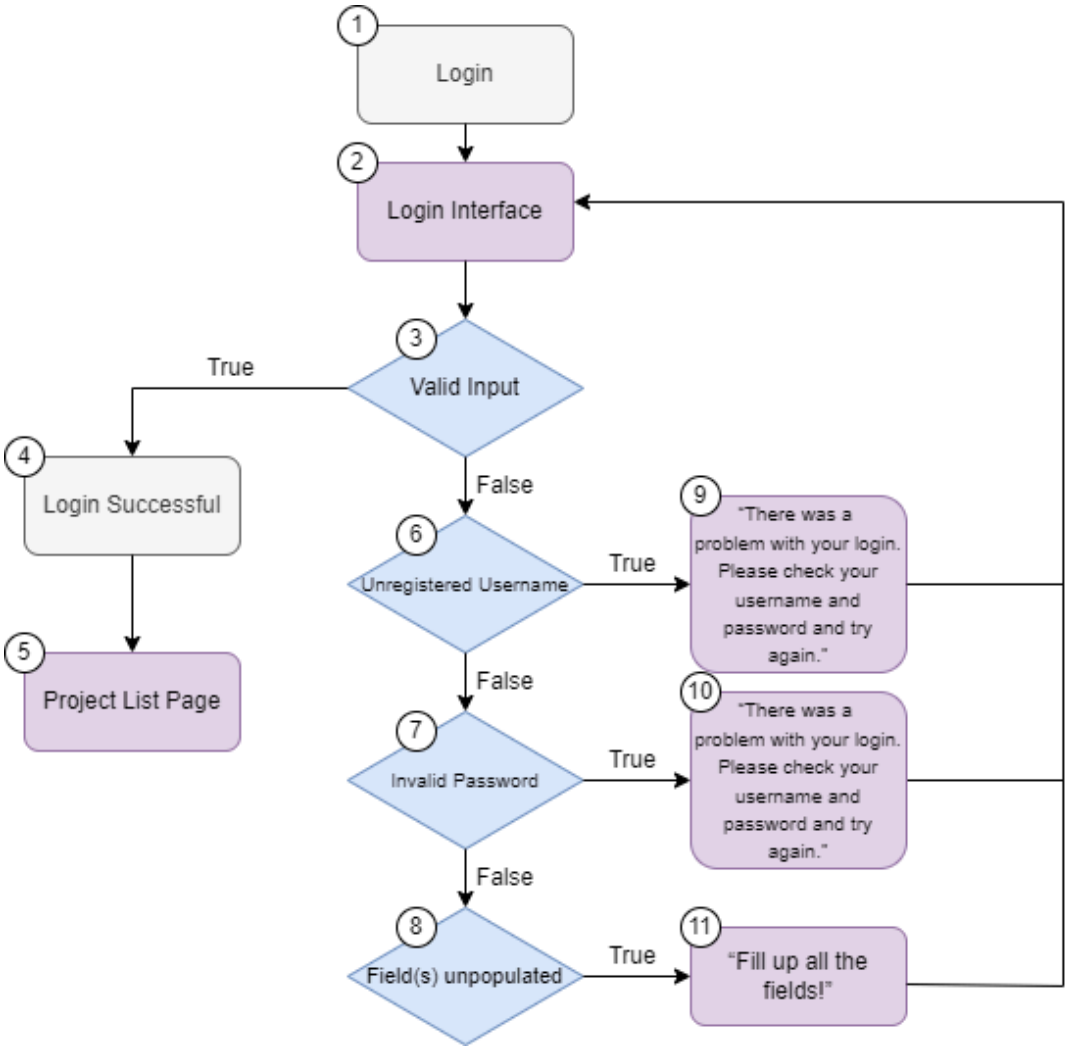
1. Registration



Cyclomatic complexity = 22 - 17 + 2 = 7

ID	Basis paths	Real execution paths
#1	1, 2, 3, 4, 5	1, 2, 3, 4, 5
#2	1, 2, 3, 6, 12, 2, 3, 4, 5	1, 2, 3, 6, 12, 2, 3, 4, 5
#3	1, 2, 3, 6, 7, 13, 2, 3, 4, 5	1, 2, 3, 6, 7, 13, 2, 3, 4, 5
#4	1, 2, 3, 6, 7, 8, 14, 2, 3, 4, 5	1, 2, 3, 6, 7, 8, 14, 2, 3, 4, 5
#5	1, 2, 3, 6, 7, 8, 9, 15, 2, 3, 4, 5	1, 2, 3, 6, 7, 8, 9, 15, 2, 3, 4, 5
#6	1, 2, 3, 6, 7, 8, 9, 10, 16, 2, 3, 4, 5	1, 2, 3, 6, 7, 8, 9, 10, 16, 2, 3, 4, 5
#7	1, 2, 3, 6, 7, 8, 9, 10, 11, 17, 2, 3, 4, 5	1, 2, 3, 6, 7, 8, 9, 10, 11, 17, 2, 3, 4, 5

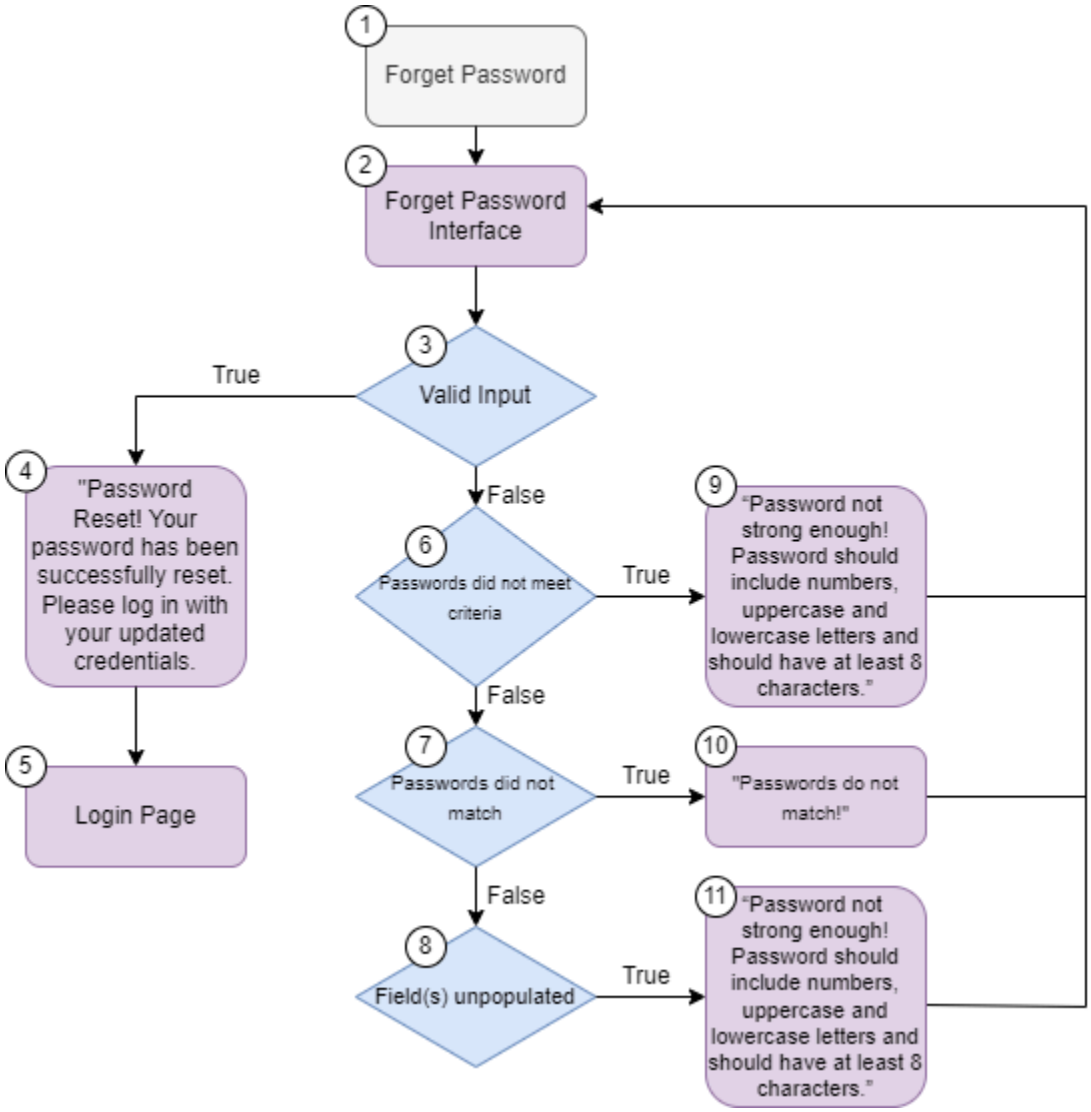
2. Login



Cyclomatic complexity = 13 - 11 + 2 = 4

ID	Basis paths	Real execution paths
#1	1, 2, 3, 4, 5	1, 2, 3, 4, 5
#2	1, 2, 3, 6, 9, 2, 3, 4, 5	1, 2, 3, 6, 9, 2, 3, 4, 5
#3	1, 2, 3, 6, 7, 10, 2, 3, 4, 5	1, 2, 3, 6, 7, 10, 2, 3, 4, 5
#4	1, 2, 3, 6, 7, 8, 11, 2, 3, 4, 5	1, 2, 3, 6, 7, 8, 11, 2, 3, 4, 5

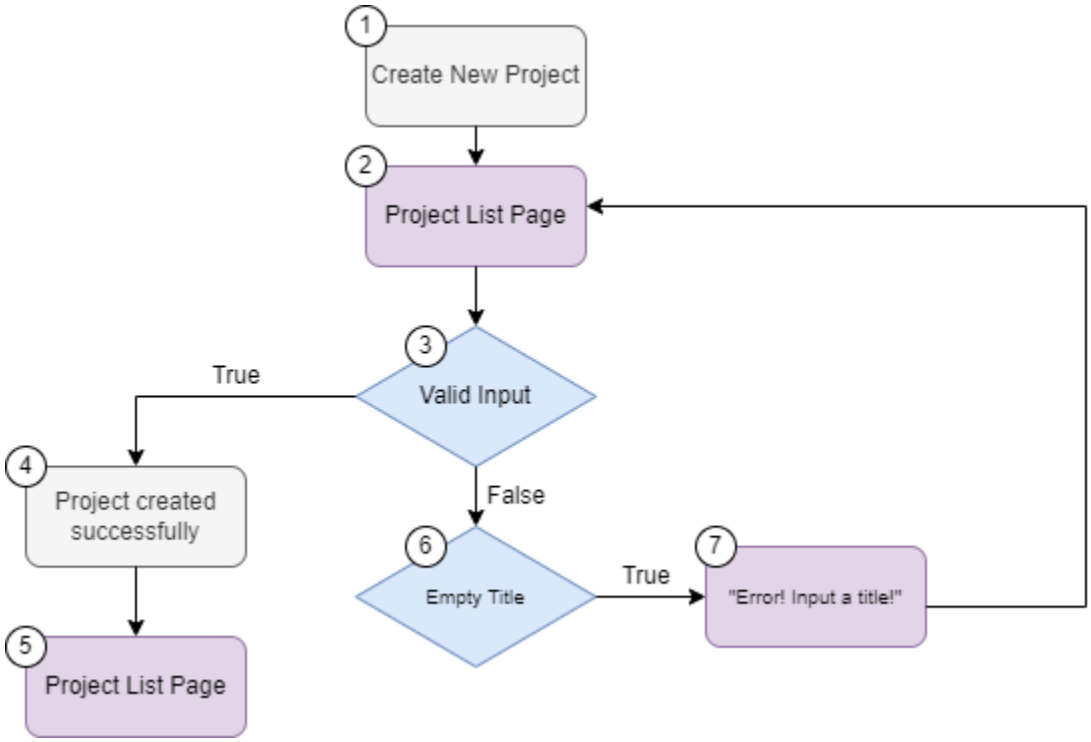
3. Forget Password



Cyclomatic complexity = 13 - 11 + 2 = 4

ID	Basis paths	Real execution paths
#1	1, 2, 3, 4, 5	1, 2, 3, 4, 5
#2	1, 2, 3, 6, 9, 2, 3, 4, 5	1, 2, 3, 6, 9, 2, 3, 4, 5
#3	1, 2, 3, 6, 7, 10, 2, 3, 4, 5	1, 2, 3, 6, 7, 10, 2, 3, 4, 5
#4	1, 2, 3, 6, 7, 8, 11, 2, 3, 4, 5	1, 2, 3, 6, 7, 8, 11, 2, 3, 4, 5

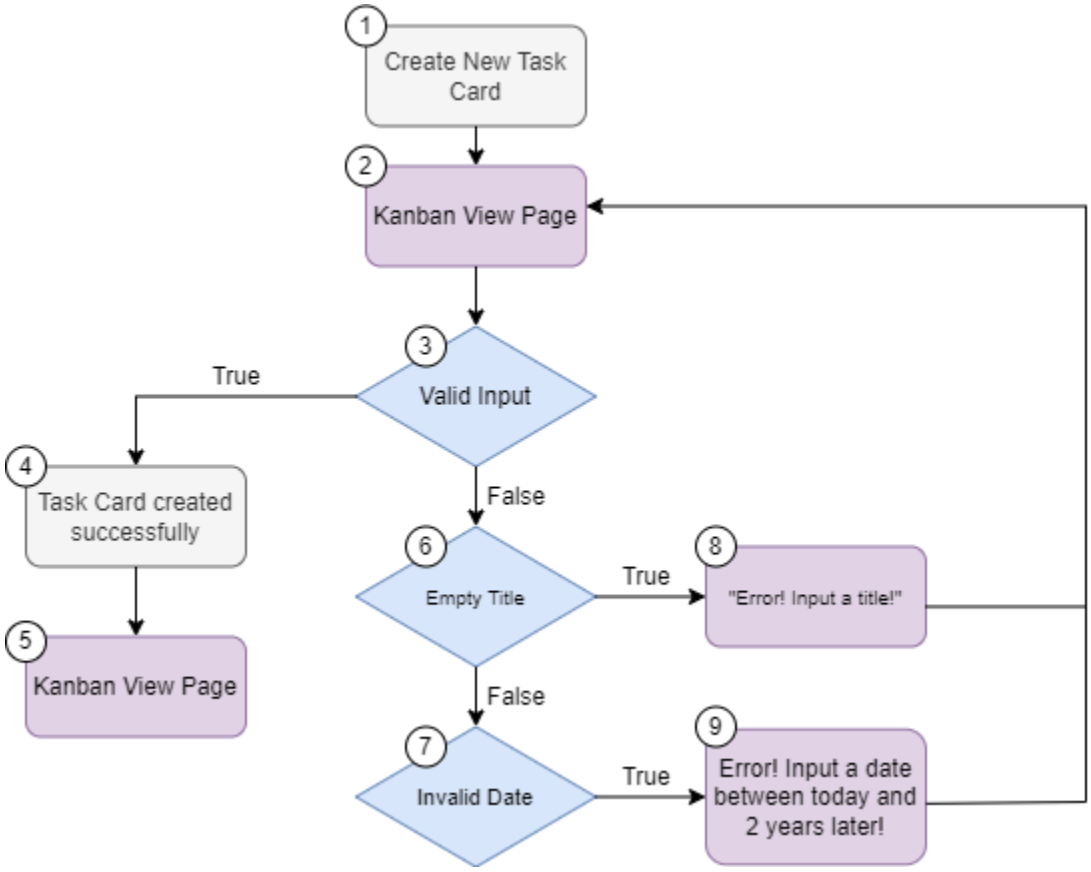
4. Create New Project



Cyclomatic complexity = 7 - 7 + 2 = 2

ID	Basis paths	Real execution paths
#1	1, 2, 3, 4, 5	1, 2, 3, 4, 5
#2	1, 2, 3, 6, 7, 2, 3, 4, 5	1, 2, 3, 6, 7, 2, 3, 4, 5

5. Create New Task Card



Cyclomatic complexity = 10 - 9 + 2 = 3

ID	Basis paths	Real execution paths
#1	1, 2, 3, 4, 5	1, 2, 3, 4, 5
#2	1, 2, 3, 6, 8, 2, 3, 4, 5	1, 2, 3, 6, 8, 2, 3, 4, 5
#3	1, 2, 3, 6, 7, 9, 2, 3, 4, 5	1, 2, 3, 6, 7, 9, 2, 3, 4, 5

Demo Script

Introduction

Slide 1 Tan: Good afternoon everyone, we are Team Kanbanize and our team consists of Samuel Seow, Willy, Samuel Tan, Cheyenne and Senchia. Today, we will be presenting our SC2006 project, Kanbanize.

Slide 2 Tan: These are what we will be running through today with regards to our project.

Slide 3 Tan: Before we begin, here is a brief overview of what Kanbanize is about. Kanbanize is a website that is designed to help teams and organizations perform seamless workflow management through the use of a Kanban board-style interface. Users can customize their tasks and prioritize workflows within their workspace. Our main purpose is to visualize workflow and maximize efficiency for work management.

Use Case Diagram and Main Functionalities

Slide 4 Tan: Firstly, let's refer to our Use Case Diagram

Slide 5 Tan: Our functionality consist of login, logout, registering of a new account and displaying of current ongoing projects and the milestones of each deliverable task on each project.

Board Creation

Upon creation of a valid account, users can begin to create Project Boards.

Tasks

Once inside a Project Board, users can create tasks, represented by sticky notes and add them to the one of the workflow columns - (To do, In progress, needs review and completed) .

When creating a task, users can set the task title, description and due date. Users can also flag/unflag and comment on tasks. Once a task is added, its details can be edited and it can be moved between the workflow columns as work progresses. Tasks can also be deleted.

Manage Users

Users can add other users to, or view current users of a Project Board. They can be assigned or unassigned specific tasks.

Boards Management

Users can create additional Project Boards, view or rename their existing boards and also delete boards

Forgot Password

Users can reset their password using their email if they ever forget it.

Innovative Features

Slide 6 Cheyenne: So what is it that really sets Kanbanize apart from all other similar applications? It is the following features that adds value to our application:

Slide 7 Cheyenne: The most innovative feature of Kanbanize its focus on cross-functional collaboration, which presents itself in many of its features.

Click

The project owner has the option to add any registered user in our database into the project as a collaborator, meaning they can add, edit, delete tasks; same as the project owner.

Click

Another special feature that promotes user collaboration includes the Task Assign feature, which allows any user to assign a task to any other user, encouraging user interaction within the website. The user is also allowed to unassign a task from someone to switch task allocations.

Click

Finally, each task card carries a comment feature which allows all users to input text separate from the task description, providing a means for users to give feedback on each assigned task.

System Architecture

Slide 8/9 Willy:Our system architecture consists of the user interfaces, such as registration/login, projects and board which serves as the system's boundaries. The control layer consists of managers like LoginManager, RegistrationManager, KanbanViewManager and our Database API manager/Forget Password API manager. These managers coordinate the interactions between the User Interface and the underlying entities such as user, board and tasks. Our frontend UI is handled by Angular (explain) and our backend is handled by MongoDB.

- MEAN Stack (https://www.mongodb.com/mean-stack#:~:text=The%20MEAN%20stack%20is%20a,layers%20of%20the%20technology%20stack)

Design Patterns

Model-View-ViewModel (MVVM) Pattern

Slide 10 Willy: In the design phase, we wanted to create a modular and maintainable system that handles the functionality of our website.

Slide 11 Willy: Therefore, we decided to use the MVVM design pattern to manage the various components. The Model takes charge of all aspects of the user created data from Boards and Tasks. The view allows us to visualize the Kanban board, tasks workflow and individual task card details. The ViewModel, is represented by components. These component classes contain

the application logic and interact with services to retrieve and manipulate data. This binding between view and model allows seamless data flow between viewmodel and view.

click

We also used the Service Layer Pattern, which ensures that there is real-time synchronization whenever an update is made to the tasks(edited/dragged). We created Angular services to interact with the MongoDB database using HttpClient. These services handle CRUD(create,read,update,delete) operations and manage data synchronization between the frontend and backend for our tasks, boards and users .

click

Lastly, we used dependency injection to achieve loose coupling between the various external services such as authentication, json webtokens, smtp requests and mongoose. We injected services into Angular components using Angular's dependency injection mechanism. This ensures our program is modular, extensible and we would have a codebase that is easier to maintain.

Traceability

Slide 12 Seow : To demonstrate the traceability of our project, we will be going through the registration and login features.

Slide 13 Seow: This is the use case for the registration of an account. The flow of the events are as stated here: The user is brought to the login page first, and clicks sign up if they do not have

an account. The sign up interface is displayed to the user, where they will fill up their details. The user clicks on Register, which creates their account and brings them to the project list page.

Exceptions include username or email already existing in the database, password not fulfilling the special requirement, and “password” and “confirm password” not matching.

Slide 14 Seow : Next, the use case for login. The flow of events are; the user is brought to the login page first, where they click login. Login interface is displayed, where users input their unique username and password in the respective fields and clicks on Login. After the Kanbanize database verifies the user’s login credentials, the user is then brought to the project list page. There is no alternative way to login.

Exception occurs when the username or password does not match the one in the system. The user will be returned to the login page to re-enter all details again.

Slide 15 Seow: Next will be the class diagram for this use case.

So in our diagram these are all our model view components, including registration and login. Login button calls auth service and waits for a reply, which then calls webreqservice, which then makes a HTTP request to the API. So this request will be received by the backend server to authenticate the user. Successful login will create a session for the user.

Slide 16 Seow: Moving onto the sequence diagram, this is the sequence of messages exchanged between the User, the interface, and Database. Our sequence diagram follows the use case flow of events, including loops where user inputs incorrect details and is prompted to redo until a successful registration or login.

Slide 17-19 Senchia: The testing for this function is relatively simple, using equivalence class testing techniques to generate these test cases since the inputs are discrete values. As you can see, the general cases are testing the aforementioned input requirements, namely whether the password meets requirements, matches, all fields are filled with the correct format, and email is already linked to an account. The specific cases test the validity of each user input field, first email and password, and then a combination of both.

Slide 20 Seow: Here is the control flow graph for user registration, with all the decision nodes numbered. We have 22 edges and 17 nodes, therefore a cyclomatic complexity value of 7. Which means 7 basis paths to test.

Slide 21 Seow: Here is the control flow graph for user login, with all the decision nodes numbered. We have 13 edges and 11 nodes, therefore a cyclomatic complexity value of 4. Which means 4 basis paths to test.

Good Software Engineering Practices

Slide 22 Cheyenne: Having good software engineering practice is vital when developing any software.

Slide 23 Cheyenne: When developing this system, our team utilized version control systems such as Github to manage changes to the codebase. This allows for collaboration, tracking changes, and reverting to previous versions if necessary.

click

Another credible SE practice is good documentation and communication throughout the course of the project. This ensures the traceability of our progress and that information is

well-documented and effectively communicated among team members and stakeholders. Clear and comprehensive documentation helps in maintaining a shared understanding of the software development process and facilitates collaboration.

Future Upgrades

Slide 24 Senchia: Finally, for the future development plans of our website, we aim to improve the functionality and design of our website.

Slide 25 Senchia: Firstly, we want to Implement advanced filtering and search capabilities that allow users to quickly find and focus on specific tasks. The tasks can be filtered based on due date, priority status and name.

click

Secondly, we want to ensure accessibility on mobile devices by implementing progressive web app features for a seamless mobile experience.

click

Lastly, users can create and customize task card fields such as attaching files, checkboxes, subtasks, labels, tags, location, customizable priority levels and activity logs for each task. Users can also change and personalize the colours of their Kanban boards.

Slide 26: Live Demo

Demo Script/Flow

Now we will be going into the live demonstration of our project.

Registration Do:Willy, Narrate: Seow

Hi I'm John, I am doing a software engineering project and I want to use a project management tool to help visualize workflow. I came across this website called Kanbanize that I can use.

1. First, the user will register for an account.
    a. To demonstrate error checking, we will go through a few exception cases first.
        i. User will enter a username that already exists in the database
            1. Error message pops up
        ii. User enters an email that already exists in the database
            1. Error message pops up
        iii. User will input a password that doesn't fulfill the password requirements (8 characters minimum; one upper and lower case and one number)
            1. User will be prompted to re-enter a password.
        iv. User will input password such that password and confirm password are different
            1. User will be prompted to re-enter a password
    b. Now, User will enter all details correctly
    c. User is brought to project list page

Logout Do:Willy, Narrate: Seow

2. Now we will demonstrate our logout feature.
    a. User clicks on their username in the Header bar.
    b. A drop-down menu is displayed with "Log out"
    c. User clicks on "Log out".
    d. User is redirected to the Login Page.

Login and Forget Password Do: Willy, Narrate: Seow

3. Now we will properly log into the account we created just now, using the updated password.
    a. Lets say John forgets his password and enters incorrect password
        i. Error message pops up
4. Since John forgot his password, he will reset it
    a. User clicks on "Forgot Password? Click here".

b. User enters his email address in the textbox.
c. System sends an email to the user's email address.
d. The email includes a hyperlink to the Reset Password Page.
e. User clicks on "Reset Password" and is redirected to the Reset Password Page.
f. User will be prompted to enter a new Password, and once more to confirm the new password.
g. User clicks "Confirm".
h. User is redirected back to the Login Page.
i. John will login to his account using his new password
    i. User inputs their unique username and password in text boxes and clicks on the Login button.
    ii. Kanbanize system verifies users login credentials.
    iii. User is redirected to the Project List Page.

Create Project Do:Seow, Narrate: Senchia

5. Now we will create a new project.
    a. User is at the Project List Page.
    b. User clicks on "+ New Project".
    c. User is prompted to enter project name.
    d. A new project is created and the user returns to the Project List Page.

Switch Project Do: Seow, Narrate: Senchia

6. Now we will demonstrate how to switch between projects
    a. First create a second project to switch between projects
        i. User will end up at project list page after second project is completed
    b. User is at the Project List Page and sees the 2 projects created.
    c. User clicks on the desired project to switch to.
    d. User is redirected to that project's Kanban View Page.
    e. Alternatively, if the user is already in the project's Kanban View Page
        i. User clicks "← Back to Project List"
        ii. User is redirected to the Project List Page.
        iii. User switches to other project

Delete Project Do:Seow, Narrate: Senchia

7. Now we will delete the project we are currently on.
    a. User clicks on the “Settings” icon at the top right of the page.
    b. A dropdown menu is shown with “Delete Board” at the bottom.
    c. User clicks on “Delete Board”.
    d. User is redirected to a page to confirm deletion of the board.
    e. User clicks “Save” and deletes the project.
    f. User is then redirected back to the Project List Page.

Add task Do: Senchia Narrate: Cheyenne

8. Now we will add tasks to the remaining project
    a. Select the desired project.
    b. Click on “+” on the top right corner of the particular column.
    c. We will go through all exception cases first.
        i. User does not put in a Task Title:
            1. Pop-up displays: “Input a title!”
        ii. User did not input a valid date:
            1. Pop-up displays: “Input a date between today and 2 years later!”
    d. Now, we properly Fill in “Title”, “Description”, “Due Date” and “Progress” status.
    e. Task is created once User clicks “Create”.
    f. User is brought back to Kanban View page.

Flag task Do: Senchia Narrate: Cheyenne

9. Now we will demonstrate our flag task feature
    a. User clicks on the “Flag” icon on a task card.
    b. “Flag” icon is now in red.

Edit task details Do: Senchia Narrate: Cheyenne

10. Now we will edit task details
    a. User clicks on the “Pen” icon on a task card.

b. User is redirected to a page to edit Task Details.
c. We will go through all exception cases first.
    i. User removed the previous Task Title:
    ii. Pop-up displays: “Input a title!”
    iii. User did not input a valid date:
    iv. Pop-up displays: “Input a date between today and 2 years later!”
d. Now we properly edit the Task Title, Task Description and Task Due Date.
e. User saves changes by clicking “Save”.
f. User is brought back to Kanban view page

Edit task status Do: Senchia Narrate: Cheyenne

11. Now we will edit the task status, meaning to change from to-do to completed
    a. User clicks and holds on a Task Card.
    b. User drags the Card to another Column.
    c. User releases the click.
    d. Refresh to show that its saved

Add user to project Do: Cheyenne Narrate:Tan

12. Now we will add other users in the database to our project
    a.
    b. User clicks on the “Settings” icon at the top right of the page.
    c. A dropdown menu is shown with “Add User” at the top.
    d. User clicks on “Add User”.
    e. User is redirected to a page to input a user to add.
    f. User is prompted to enter the username of the user to be added.
    g. We will go through all exception cases first.
        i. User enters a username not in database
            1. Pop-up displays: “Username not found!
    h. Now, User correctly enters the username of the user to be added
        i. User clicks “Add” and adds a new user to the project.
    i. User returns to board.

View users Do: Cheyenne Narrate: Tan

13. Now we will view all users in the current project
    a. User clicks on the “Settings” icon at the top right of the page.
    b. A dropdown menu is shown with “View Users” in the middle.
    c. User clicks on “View Users”.
    d. User is redirected to a page to view users in the current project.
    e. User can see the usernames of all the members of the project.
    f. User returns to board.

Assign task to user Do: Cheyenne Narrate:Tan

14. Now we will assign a task to the user.
    a. User clicks on the “person with a tick” icon on a task card.
    b. User is redirected to Assign User Page where the user can choose which user to assign the task to.
    c. System displays a drop down menu called “Select User”.
    d. User clicks the menu and selects the user to assign to the task.
    e. User clicks “Assign” and is shown a popup that says “User Assigned!”
    f. User returns to board

Unassign task from user Do: Cheyenne Narrate:Tan

15. Now we will unassign a task from a user
    a. User clicks on the “person with a tick” icon on a task card.
    b. User is redirected to Assign User Page where the user can choose to unassign a user from the task.
    c. System displays a drop down menu called “Select User”.
    d. User clicks the menu and selects “Unassign” at the bottom of the menu.
    e. User clicks “Assign” and is shown a popup that says “User Unassigned!”
    f. User returns to board

Delete task Do: Cheyenne Narrate:Tan

16. Now we will delete a task
    a. User clicks on a task card.
    b. User clicks on the “Trash can” icon on the bottom right-hand of the Task Card window.

c. User is redirected to a page to confirm deletion.
d. User clicks on “Save” and the task is deleted.
e. User is automatically redirected to the board.

With that, we have covered the main features of Kanbanize and this concludes the end of our demonstration video, thank you.

Use-Case Model

Functional Requirements

1. The system must have a **Login Page** with a Login interface.
    1.1. The system must display one text field for Username.
    1.2. The system must display one text field for Password.
    1.3. The system must display a “Login” button.
        1.3.1. When the user clicks “Login”, the system must validate that all required fields are not empty.
        1.3.2. When the user clicks “Login”, the system must validate whether the username is valid.
            1.3.2.1. If it is invalid, the system must prompt the user to “Check your username and try again.”
        1.3.3. When the user clicks “Login”, the system must validate whether the corresponding password is valid.
            1.3.3.1. If it is invalid, the system must prompt the user to “Check your password and try again.”
        1.3.4. When the user enters a username in the database and the correct password, the system must redirect the user to the Project List Page.
    1.4. The system must allow users to reset their account passwords through a “Forgot Password” button.

2. The system must have a **Forget Password Page** with a Forget Password interface.
    2.1. The system must display one text field for the user to enter their registered email address.
        2.1.1. When the user enters an email that is not in the database, the system must prompt the user “No users with email found.”
        2.1.2. When the user enters an email that is in the database, the system must display “Email sent!”
    2.2. The system must send an email to the inputted email address with instructions on how to change their password.
        2.2.1. When the user clicks “Reset Password” in the email, they must be redirected to the Reset Password Page.

2.2.2. The system must display a text field for the user to enter their new password.

2.2.2.1. The system must require the user to type in the new password twice to ensure the accuracy of the password.

2.2.2.2. The system must follow the same password validation practices as the Registration Page.

3. The system must have a Sign Up interface within the Login Page.

3.1. The system must display one text field for Username.

3.2. The system must display one text field for Email Address.

3.3. The system must display one text field for Password.

3.4. The system must display one text field for Confirm Password.

3.5. The system must display a “Register” button.

3.5.1. When the user clicks “Register”, the system must validate that all required fields are not empty.

3.5.1.1. If any field is empty, the system must prompt the user to “Fill up all fields.”

3.5.2. When the user clicks “Register”, the system must validate that the username has not been used before.

3.5.2.1. If the username has been used before, the system must prompt the user “Username already taken.”

3.5.3. When the user clicks “Register”, the system must validate that the email address entered is valid.

3.5.3.1. If the email address is invalid, the system must prompt the user to “Enter a valid email address.”

3.5.4. When the user clicks “Register”, the system must validate that the password entered must meet the following requirements:

3.5.4.1. A minimum of 8 characters in length.

3.5.4.2. Contains at least one character from three of the following categories: Uppercase letter (A-Z) Lowercase letter (a-z) Digit (0-9).

3.5.4.3. If the password does not satisfy the requirements, the system must prompt the user “Password not strong enough.” and include the default password validation practices.

4. The system must have a header bar.

4.1. The system must display the Username of a currently logged in user.

4.1.1. When the user clicks on their username, the system must display “Log out”.

4.1.1.1. The system must allow the user to log out of their account.

5. The system must have a **Project List Page**.

5.1. The system must display a button to create new projects.

5.1.1. The system must display one text field for Project Name.

5.2. The system must display the current projects the user is a part of, including projects that the user has been added to.

5.3. The system must redirect the user to the Kanban View Page of the project that the user selected.

6. The system must have a **Kanban View Page** for each of their projects.

6.1. The Kanban View Page of the system must contain the main functionalities of the Kanban Board that the user can use.

6.1.1. The system must allow the user to manage team members of the project.

6.1.1.1. The system must allow the user to add new users to the project.

6.1.1.2. The system must allow the user to view users of the project.

6.1.2. The system must allow the user to delete the current board.

6.2. The system must display 4 distinct columns to represent different stages of the workflow. (TO DO, IN PROGRESS, NEEDS REVIEW, COMPLETED)

6.3. The system must be able to display task cards.

6.3.1. Users must be able to create, edit or delete existing task cards.

6.3.2. Users must be able to add/edit the title and description of a task.

6.3.3. Users must be able to set/edit a deadline for a task.

6.3.4. Users must be able to flag a task as high priority.

6.3.5. Users must be able to add comments for a task card.

6.3.6. Users must be able to assign members of the project to a task card.

6.3.7. Users must be able to drag and drop tasks between different columns to update workflow progress.

Non-Functional Requirements

Performance

- When the user uses task-based functionalities, changes must be reflected within at least 1 minute.

Reliability

- The system should not crash more than thrice daily.

Security

- The system should prevent unauthorized personnel from accessing user data.
- The system should only allow registered users to view their account information.

Scalability

- The website must be able to support 10 active users while meeting all real-time requirements.
- The website must be able to support at least 20 tasks on a board while meeting all real-time requirements.

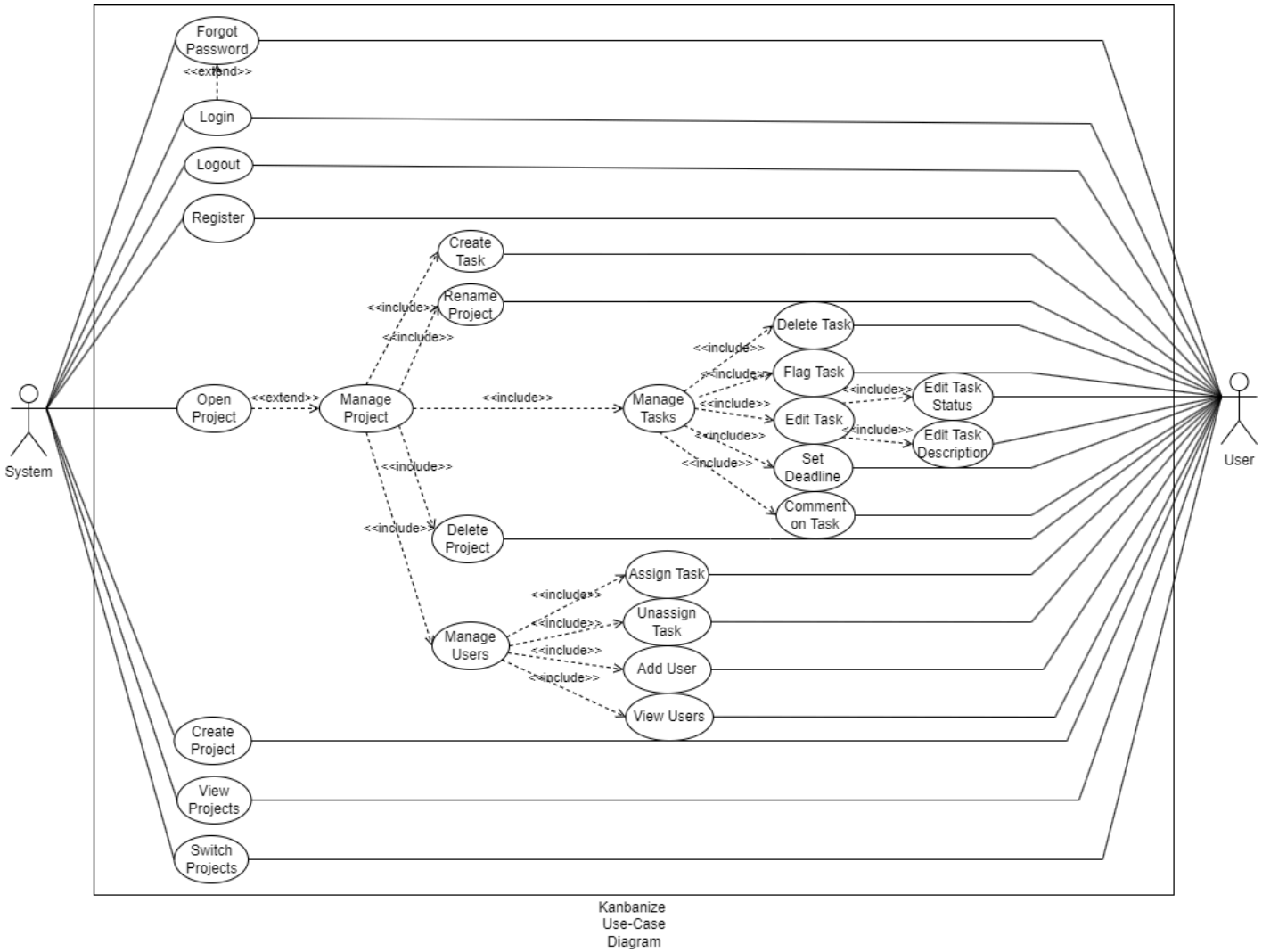
Usability

- Features on the website must be responsive within 3 seconds of being clicked.

Data Dictionary

Term	Definition
Kanban board	A project management tool which is the main feature of our website. Kanban boards visually depict work at various stages of a process using task cards to represent work items and columns to represent each stage of the process.
Column	Vertical division of a Kanban board that allows users to visualize different stages of workflow.
Task Card	Digital cards to represent a task in a project.
Login Page	The page where users can either register, login or access the Forget Password page. Found on this page are: Login Interface and Registration Interface.
Forget Password Page	The page where users can enter their email should they forget their password. Found in this page is: Forget Password Interface.
Reset Password Page	The page where users can enter their new password. Only retrieved when the user requests a password change and an email is sent to the user's inbox.
Project List Page	The page where users are directed to after a successful login. Users can view their projects or create a new project here.
Kanban View Page	The main page of a project. The main functionalities of the Kanban board will be performed here.
Header bar	A black bar loaded at the top of the page with the website's name. Users that have logged in are able to see their username and log out.

Use-Case Diagram



Use-Case Descriptions

1. Registration

Use Case ID:	1		
Use Case Name:	Registration		
Created By:	Samuel Seow	Last Updated By:	Samuel Tan
Date Created:	08/02/2024	Date Last Updated:	16/04/2024

Actor:	User, System
Description:	Registration of a new user account
Preconditions:	● User has an email account. ● User does not have an account registered with this email. ● User has a unique username. ● User has a password with a minimum of 8 characters that contains 1 of each: uppercase, lowercase and number. ● Connection to mobile data/WiFi.
Postconditions:	● Users are able to login to access Kanbanize services.
Priority:	High
Frequency of Use:	Once
Flow of Events:	1. User is able to view the Login Page. 2. User clicks on “Sign Up”. 3. Sign Up interface is displayed to the user. 4. User will be prompted to enter a “Username”, “Email”, “Password” and “Confirm Password”. 5. User clicks on the “Register” button. 6. User’s account is created and logged into his account, and brought to the Project List Page.
Alternative Flows:	NIL
Exceptions:	1. User’s Username is not unique: a. Pop-up displays: “Error. Username already taken!” 2. User’s Email Address already exists in the database: a. Pop-up displays: “Error. Email address already taken!” 3. User’s Password does not fulfill the requirements: a. Pop-up displays: “Error. Password not strong enough! Password should include numbers, uppercase and lowercase letters and should have at least 8 characters.” 4. User’s Password and Confirm Password are not the same: a. Pop-up displays: “Error. Passwords do not match!”
Includes:	NIL
Special Requirements:	● User has a password that contains 1 of each: uppercase, lowercase, number and minimum length of 8 characters.
Assumptions:	● User is connected to the internet.
Notes and Issues:	NIL

2. Login

Use Case ID:	2		
Use Case Name:	Login		
Created By:	Senchia	Last Updated By:	Samuel Tan
Date Created:	08/02/2024	Date Last Updated:	16/04/2024

Actor:	User, System
Description:	User logins to an existing account.
Preconditions:	● User has already registered for an account. ● User information is stored in the database. ● Connection to mobile data/WiFi.
Postconditions:	● User is redirected to the Project List Page.
Priority:	High
Frequency of Use:	Once until logged out
Flow of Events:	1. User is on the Login Page. 2. User clicks on “Login”. 3. Login interface is displayed to User. 4. User inputs their unique username and password in text boxes and clicks on the Login button. 5. Kanbanize system verifies users login credentials. 6. User is redirected to the Project List Page.
Alternative Flows:	NIL
Exceptions:	1. Incorrect username or password: a. Pop-up displays: “ Error. There was a problem with your login. Please check your username and password and try again.” b. System returns to step 1.
Includes:	NIL
Special Requirements:	NIL
Assumptions:	● User has already successfully registered for an account. ● User is connected to the internet.
Notes and Issues:	NIL

3. Logout

Use Case ID:	3		
Use Case Name:	Logout		
Created By:	Samuel Seow	Last Updated By:	Samuel Seow
Date Created:	19/03/2024	Date Last Updated:	15/04/2024

Actor:	User, System
Description:	User logs out of the system.
Preconditions:	● User is logged in.
Postconditions:	● User is redirected to the Login Page.
Priority:	Low
Frequency of Use:	Once until logged in
Flow of Events:	1. User clicks on their username in the Header bar. 2. A drop-down menu is displayed with “Log out” 3. User clicks on “Log out”. 4. User is redirected to the Login Page.
Alternative Flows:	● User clears browser cookies for Kanbanize and is logged out.
Exceptions:	NIL
Includes:	NIL
Special Requirements:	NIL
Assumptions:	● User has an account and is logged in. ● User is connected to the internet.
Notes and Issues:	NIL

4. Forget Password

Use Case ID:	4		
Use Case Name:	Forget Password		
Created By:	Samuel Seow	Last Updated By:	Samuel Tan
Date Created:	19/03/2024	Date Last Updated:	15/04/2024

Actor:	User, Database, System
Description:	User resets their password with “Forget Password”
Preconditions:	● User is on the Login Page and is not logged in. ● User has an existing account. ● User remembers the Email Address used to register for the account.
Postconditions:	● User is prompted to return to the Login Page.
Priority:	High
Frequency of Use:	Low, only when user forgets password
Flow of Events:	1. User clicks on “Login”. 2. Login interface is displayed to User. 3. User clicks on “Forgot Password? Click here”. 4. User enters his email address in the textbox. 5. System sends an email to the user’s email address. 6. The email includes a hyperlink to the Reset Password Page. 7. User clicks on “Reset Password” and is redirected to the Reset Password Page. 8. User will be prompted to enter a new Password, and once more to confirm the new password. 9. User clicks “Confirm”. 10. User is redirected back to the Login Page.
Alternative Flows:	NIL
Exceptions:	1. User’s Password does not fulfill the requirements: a. Pop-up displays: “Error. Password not strong enough! Password should include numbers, uppercase and lowercase letters and should have at least 8 characters.” 2. User’s Password and Confirm Password are not the same: a. Pop-up displays: “Error. Passwords do not match!” 3. User enters the incorrect email address: a. Pop-up displays: “Error. No users with email found!”
Includes:	NIL
Special Requirements:	● User has a password that contains 1 of each: uppercase, lowercase, number and minimum length of 8 characters.
Assumptions:	● User is connected to the internet. ● User has already successfully registered for an account.
Notes and Issues:	NIL

5. Create Project

Use Case ID:	5		
Use Case Name:	Create Project		
Created By:	Samuel Seow	Last Updated By:	Samuel Seow
Date Created:	08/02/2024	Date Last Updated:	15/04/2024

Actor:	User, Database, System
Description:	User creates a project.
Preconditions:	● User must have a Kanbanize account. ● User must be logged in. ● User is at the Project List Page.
Postconditions:	● A new project is created.
Priority:	High
Frequency of Use:	Medium
Flow of Events:	1. User is at the Project List Page. 2. User clicks on "+ New Project". 3. User is prompted to enter project name. 4. A new project is created and the user returns to the Project List Page.
Alternative Flows:	NIL
Exceptions:	1. User did not input a Project Name. a. Pop-up displays: "Error. Input a title!"
Includes:	NIL
Special Requirements:	NIL
Assumptions:	● User has an account and is logged in. ● User is connected to the internet.
Notes and Issues:	NIL

6. Switch Project

Use Case ID:	6		
Use Case Name:	Switch Project		
Created By:	Samuel Seow	Last Updated By:	Samuel Seow
Date Created:	19/03/2024	Date Last Updated:	15/04/2024

Actor:	User, System
Description:	User wants to switch between projects.
Preconditions:	● User must have a Kanbanize account. ● User must be logged in.
Postconditions:	● User is redirected to the desired project's Kanban View Page.
Priority:	High
Frequency of Use:	Medium
Flow of Events:	1. User is at the Project List Page and sees multiple projects which are either under their name or they have been added to. 2. User clicks on the desired project to switch to. 3. User is redirected to that project's Kanban View Page.
Alternative Flows:	1. User is at the Kanban View Page in another project. 2. User clicks on "← Back to Project List" 3. User is redirected to the Project List Page. 4. User is at the Project List Page and sees multiple projects which are under their name. 5. User clicks on the desired project to switch to. 6. User is redirected to that project's Kanban View Page.
Exceptions:	NIL
Includes:	NIL
Special Requirements:	NIL
Assumptions:	● User has an account and is logged in. ● User is connected to the internet. ● User has more than 1 project.
Notes and Issues:	NIL

7. Delete Project

Use Case ID:	7		
Use Case Name:	Delete Project		
Created By:	Samuel Seow	Last Updated By:	Samuel Seow
Date Created:	15/04/2024	Date Last Updated:	15/04/2024

Actor:	User, System
Description:	User deletes the current project board.
Preconditions:	● User must be logged in. ● User must have a registered account. ● User is at the Kanban View Page of the project to be deleted.
Postconditions:	● User is redirected to the Project List Page.
Priority:	High
Frequency of Use:	Medium
Flow of Events:	1. User clicks on the “Settings” icon at the top right of the page. 2. A dropdown menu is shown with “Delete Board” at the bottom. 3. User clicks on “Delete Board”. 4. User is redirected to a page to confirm deletion of the board. 5. User clicks “Delete” and deletes the project. 6. User is then redirected back to the Project List Page.
Alternative Flows:	NIL
Exceptions:	NIL
Includes:	NIL
Special Requirements:	NIL
Assumptions:	● User has an account and is logged in. ● User is connected to the internet.
Notes and Issues:	NIL

8. Add Task

Use Case ID:	8		
Use Case Name:	Add Task		
Created By:	Senchia Gladine	Last Updated By:	Samuel Tan
Date Created:	08/02/2024	Date Last Updated:	16/04/2024

Actor:	User, System
Description:	User adds a task in a selected project.
Preconditions:	● User must be logged in. ● User must have a registered account. ● User is at the Kanban View Page.
Postconditions:	● New task card shows up at the chosen column.
Priority:	High
Frequency of Use:	High
Flow of Events:	1. User click on “+” on the top right corner of the particular column. 2. User is redirected to new task creation page. 3. User fills in “Title”, “Description” and “Due Date”. 4. Task is created once User clicks “Create”.
Alternative Flows:	NIL
Exceptions:	1. User did not input a Task Title: a. Pop-up displays: “Input a title!” 2. User inputs a title more than 50 characters long: a. Pop-up displays: “Error. Limit your input to 50 characters!” 3. User inputs a description more than 250 characters long: a. Pop-up displays: “Error. Limit your input to 250 characters!” 4. User did not input a valid date: a. Pop-up displays: “Input a date between today and 2 years later!”
Includes:	● Title ● Description ● Due Date
Special Requirements:	NIL
Assumptions:	● User has an account and is logged in. ● User is connected to the internet.
Notes and Issues:	NIL

9. Flag Task

Use Case ID:	9		
Use Case Name:	Flag Task		
Created By:	Samuel Seow	Last Updated By:	Samuel Seow
Date Created:	15/04/2024	Date Last Updated:	15/04/2024

Actor:	User, System
Description:	User flags a task in a selected project.
Preconditions:	● User must be logged in. ● User must have a registered account. ● User is at the Kanban View Page.
Postconditions:	● Task is now flagged with a red flag icon.
Priority:	Low
Frequency of Use:	High
Flow of Events:	7. User clicks on the “Flag” icon on a task card. 8. “Flag” icon is now red.
Alternative Flows:	NIL
Exceptions:	NIL
Includes:	NIL
Special Requirements:	NIL
Assumptions:	● User has an account and is logged in. ● User is connected to the internet.
Notes and Issues:	NIL

10. Edit Task Details

Use Case ID:	10		
Use Case Name:	Edit Task Details		
Created By:	Samuel Seow	Last Updated By:	Samuel Seow
Date Created:	19/03/2024	Date Last Updated:	15/04/2024

Actor:	User, System
Description:	User updates the details of the task in a project.
Preconditions:	● User must be logged in. ● User must have a registered account. ● User must be a member of the project. ● User is at the Kanban View Page.
Postconditions:	● Task card is updated with new details.
Priority:	High
Frequency of Use:	High
Flow of Events:	1. User clicks on the “Pen” icon on a task card. 2. User is redirected to edit task details page. 3. User is able to edit the Task Title, Task Description and Task Due Date. 4. User saves changes by clicking “Save”.
Alternative Flows:	NIL
Exceptions:	1. User removed the previous Task Title: a. Pop-up displays: “Input a title!” 2. User inputs a title more than 50 characters long: a. Pop-up displays: “Error. Limit your input to 50 characters!” 3. User inputs a description more than 250 characters long: a. Pop-up displays: “Error. Limit your input to 250 characters!” 4. User did not input a valid date: a. Pop-up displays: “Input a date between today and 2 years later!”
Includes:	● Title ● Description ● Due Date
Special Requirements:	NIL
Assumptions:	● User has an account and is logged in. ● User is connected to the internet. ● User has already created a task.
Notes and Issues:	NIL

11. Edit Task Status

Use Case ID:	11		
Use Case Name:	Edit Task Status		
Created By:	Willy Tang	Last Updated By:	Samuel Seow
Date Created:	08/02/2024	Date Last Updated:	01/04/2024

Actor:	User, System
Description:	User updates the status of the task in a project.
Preconditions:	● User must be logged in. ● User must have a registered account. ● User must be a member of the project. ● User is at the Kanban View Page.
Postconditions:	● The status of the task is updated.
Priority:	High
Frequency of Use:	High
Flow of Events:	1. User clicks and holds on a Task Card. 2. User drags the Card to another Column. 3. User releases the click.
Alternative Flows:	NIL
Exceptions:	NIL
Includes:	NIL
Special Requirements:	NIL
Assumptions:	● User has an account and is logged in. ● User is connected to the internet. ● User has already created a task prior.
Notes and Issues:	● User must drop the task into 1 of the 4 columns: TO DO, IN PROGRESS, NEEDS REVIEW, COMPLETED.

12. Assign Task to User

Use Case ID:	12		
Use Case Name:	Assign Task to User		
Created By:	Samuel Seow	Last Updated By:	Samuel Seow
Date Created:	15/04/2024	Date Last Updated:	15/04/2024

Actor:	User, System
Description:	User assigns a task in a selected project to a member.
Preconditions:	● User must be logged in. ● User must have a registered account. ● User is at the Kanban View Page. ● User to be assigned must be part of the project.
Postconditions:	● Task is now assigned to a user.
Priority:	Low
Frequency of Use:	High
Flow of Events:	1. User clicks on the “person with a tick” icon on a task card. 2. User is redirected to Assign User to Task Page where the user can choose which user to assign the task to. 3. System displays a drop down menu called “Select User”. 4. User clicks the menu and selects the user to assign to the task. 5. User clicks “Assign” and is shown a popup that says “User Assigned!”
Alternative Flows:	NIL
Exceptions:	NIL
Includes:	NIL
Special Requirements:	NIL
Assumptions:	● User has an account and is logged in. ● User is connected to the internet.
Notes and Issues:	NIL

13. Unassign Task from User

Use Case ID:	13		
Use Case Name:	Unassign Task from User		
Created By:	Cheyenne Seet	Last Updated By:	Cheyenne Seet
Date Created:	16/04/2024	Date Last Updated:	16/04/2024

Actor:	User, System
Description:	User unassigns a task in a selected project from a member.
Preconditions:	● User must be logged in. ● User must have a registered account. ● User is at the Kanban View Page. ● User to be unassigned must already be assigned to the Task.
Postconditions:	● Task is now unassigned from a member.
Priority:	Low
Frequency of Use:	High
Flow of Events:	1. User clicks on the “person with a tick” icon on a task card. 2. User is redirected to Assign User to Task Page where the user can choose to unassign a user from the task. 3. System displays a drop down menu called “Select User”. 4. User clicks the menu and selects “Unassign” at the bottom of the menu. 5. User clicks “Assign” and is shown a popup that says “User Unassigned!”
Alternative Flows:	NIL
Exceptions:	NIL
Includes:	NIL
Special Requirements:	NIL
Assumptions:	● User has an account and is logged in. ● User is connected to the internet.
Notes and Issues:	NIL

14. Delete Task

Use Case ID:	14		
Use Case Name:	Delete Task		
Created By:	Samuel Seow	Last Updated By:	Samuel Seow
Date Created:	19/03/2024	Date Last Updated:	01/04/2024

Actor:	User, System
Description:	User deletes a task from the project.
Preconditions:	● User must be logged in. ● User must have a registered account. ● User must be a member of the project. ● User is at the Kanban View Page.
Postconditions:	● The task is removed from the system.
Priority:	Medium
Frequency of Use:	Medium
Flow of Events:	1. User clicks on a task card. 2. User clicks on the “Trash can” icon on the bottom right-hand of the Task Card window. 3. User is redirected to a page to confirm deletion. 4. User clicks on “Delete” and deletes the task.
Alternative Flows:	NIL
Exceptions:	NIL
Includes:	NIL
Special Requirements:	NIL
Assumptions:	● User has an account and is logged in. ● User is connected to the internet. ● User has already created a task prior that can be deleted.
Notes and Issues:	NIL

15. Add User to Project

Use Case ID:	15		
Use Case Name:	Add User to Project		
Created By:	Samuel Seow	Last Updated By:	Samuel Seow
Date Created:	15/04/2024	Date Last Updated:	15/04/2024

Actor:	User, System
Description:	User wants to add a user to a selected project.
Preconditions:	● User must be logged in. ● User must have a registered account. ● User is at the Kanban View Page.
Postconditions:	●
Priority:	Low
Frequency of Use:	Medium
Flow of Events:	1. User clicks on the “Settings” icon at the top right of the page. 2. A dropdown menu is shown with “Add User” at the top. 3. User clicks on “Add User”. 4. User is redirected to Add User page to input a user to add. 5. User is prompted to enter the username of the user to be added. 6. User clicks “Add” and adds a new user to the project.
Alternative Flows:	NIL
Exceptions:	1. User enters a username not found in the database a. Pop-up displays: “Username not found!”
Includes:	NIL
Special Requirements:	NIL
Assumptions:	● User has an account and is logged in. ● User is connected to the internet.
Notes and Issues:	NIL

16. View Users in Project

Use Case ID:	16		
Use Case Name:	View Users in Project		
Created By:	Samuel Seow	Last Updated By:	Samuel Seow
Date Created:	15/04/2024	Date Last Updated:	15/04/2024

Actor:	User, System
Description:	User wants to view all users in a project.
Preconditions:	● User must be logged in. ● User must have a registered account. ● User is at the Kanban View Page.
Postconditions:	●
Priority:	Low
Frequency of Use:	Medium
Flow of Events:	1. User clicks on the “Settings” icon at the top right of the page. 2. A dropdown menu is shown with “View Users” in the middle. 3. User clicks on “View Users”. 4. User is redirected to View Users page to view users in the current project. 5. User can see the usernames of all the members of the project.
Alternative Flows:	NIL
Exceptions:	NIL
Includes:	NIL
Special Requirements:	NIL
Assumptions:	● User has an account and is logged in. ● User is connected to the internet.
Notes and Issues:	NIL

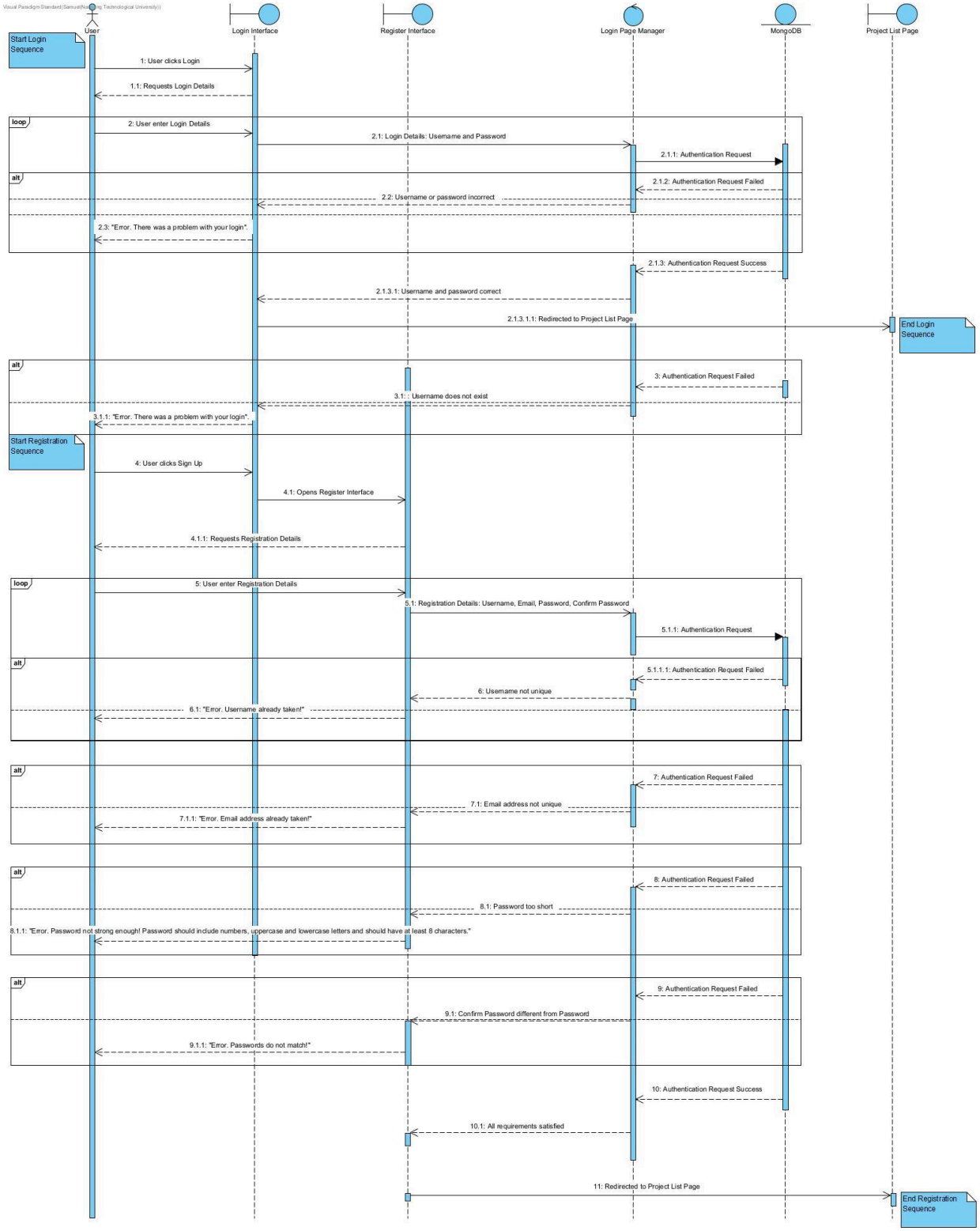
17. Rename Project

Use Case ID:	17		
Use Case Name:	Rename Project		
Created By:	Samuel Seow	Last Updated By:	Samuel Seow
Date Created:	17/04/2024	Date Last Updated:	17/04/2024

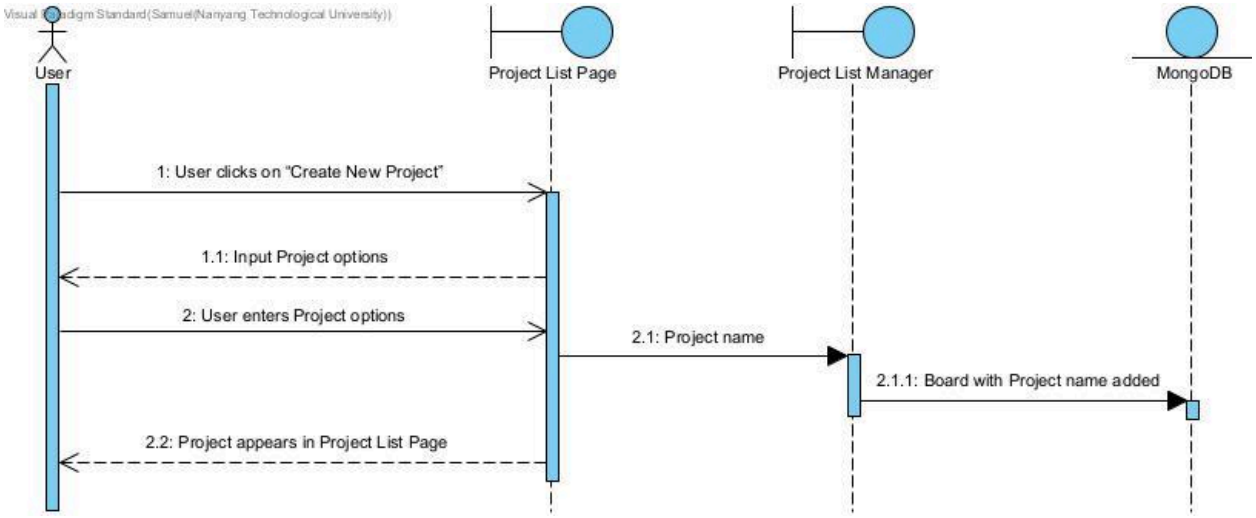
Actor:	User, System
Description:	User renames the current project board.
Preconditions:	● User must be logged in. ● User must have a registered account. ● User is at the Kanban View Page of the project to be renamed.
Postconditions:	NIL
Priority:	Low
Frequency of Use:	Low
Flow of Events:	9. User hovers over board name at the top left hand corner of the Kanban View Page. 10. A “pen” icon is displayed. 11. User clicks on “pen” icon to edit the board title. 12. User renames the project. 13. User saves changes by clicking the “save” icon. 14. New project title is reflected.
Alternative Flows:	NIL
Exceptions:	NIL
Includes:	NIL
Special Requirements:	NIL
Assumptions:	● User has an account and is logged in. ● User is connected to the internet.
Notes and Issues:	NIL

Sequence Diagrams

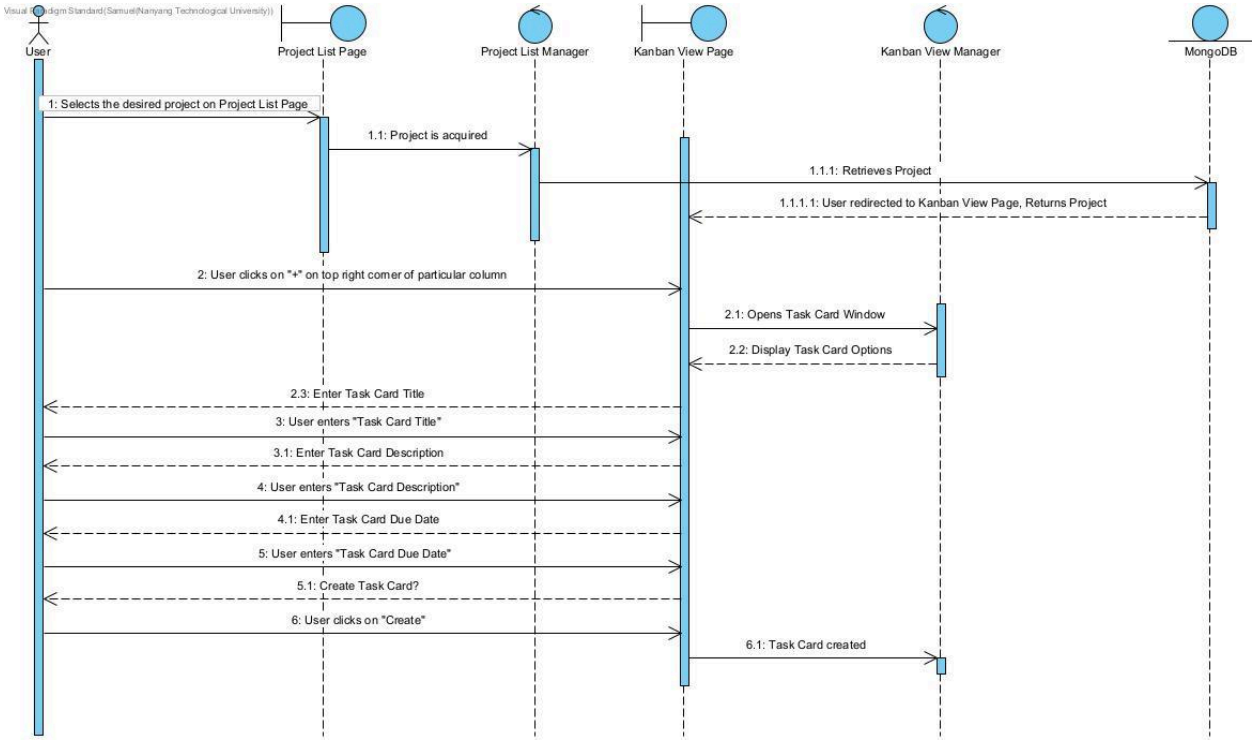
1. Registration and Login



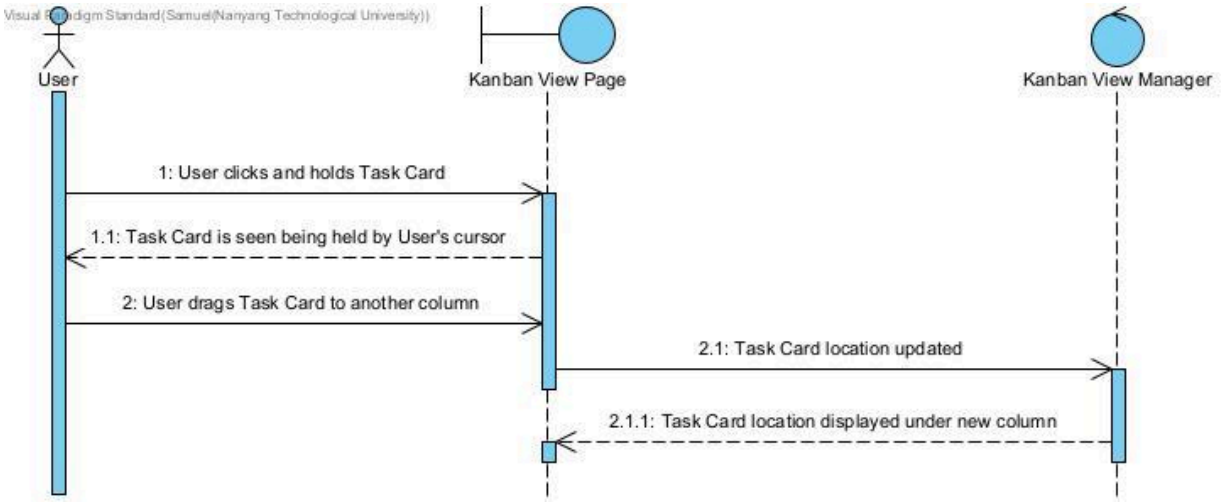
2. Create Project



3. Add Task

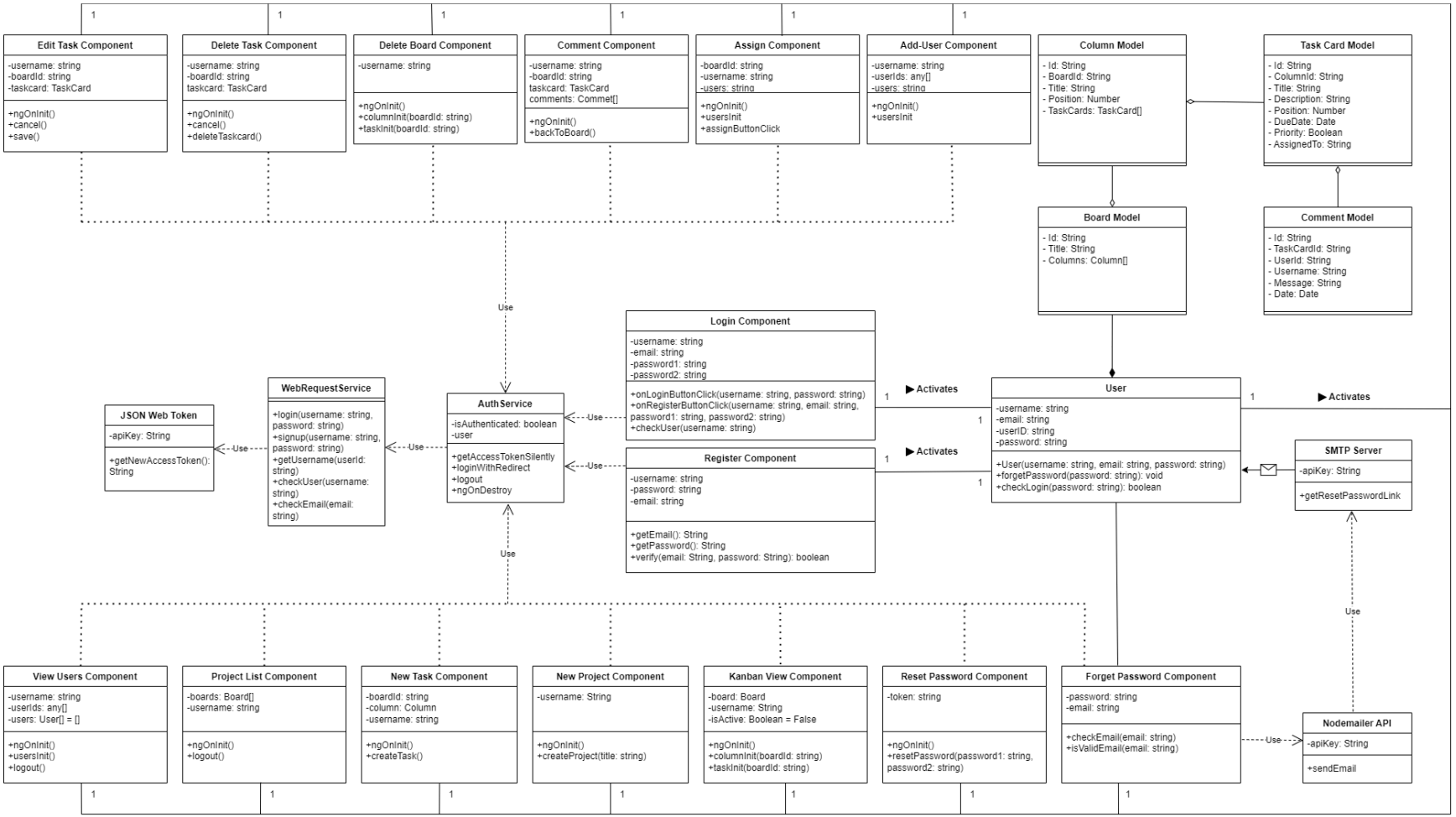


4. Edit Task (Status)

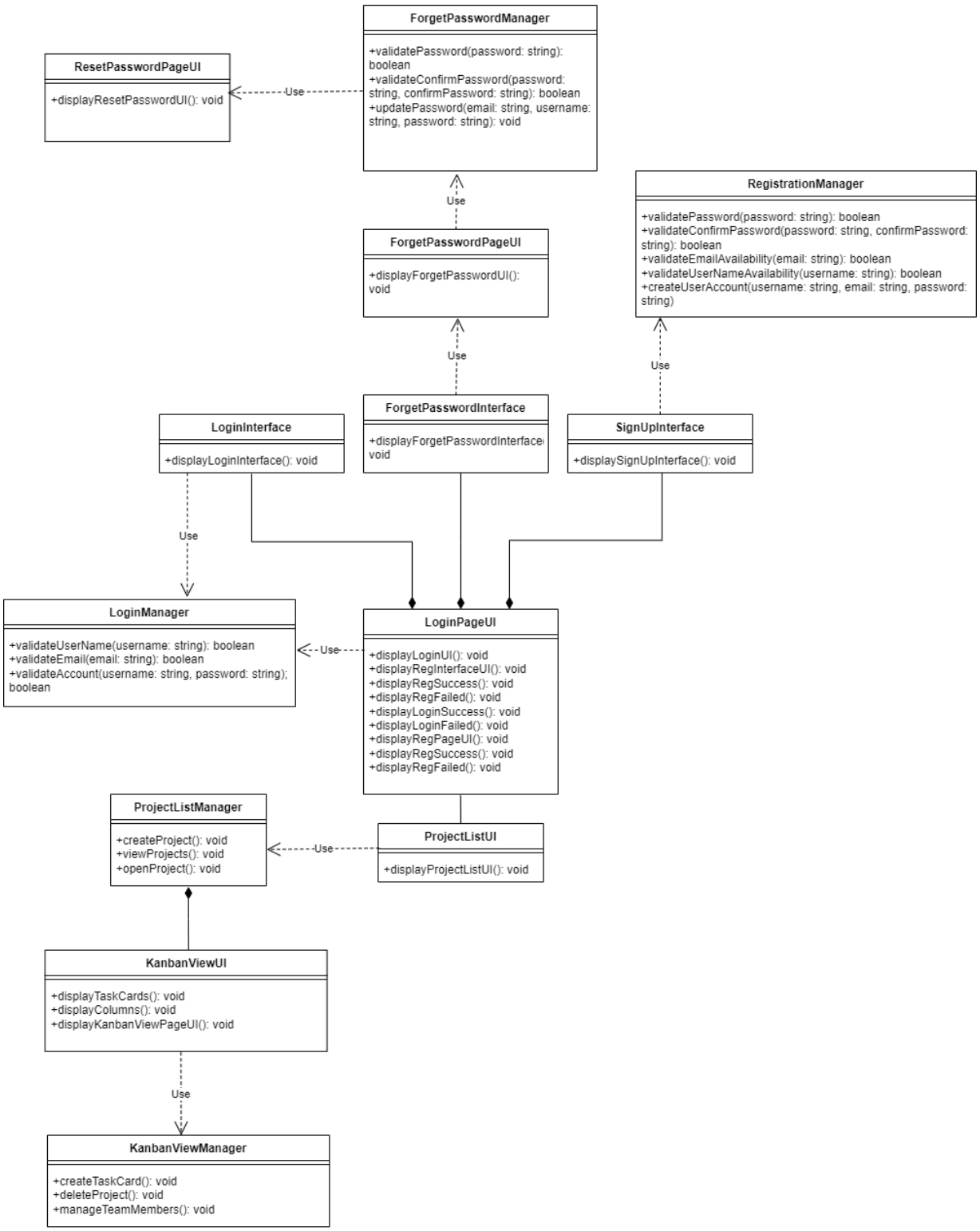


Design Model

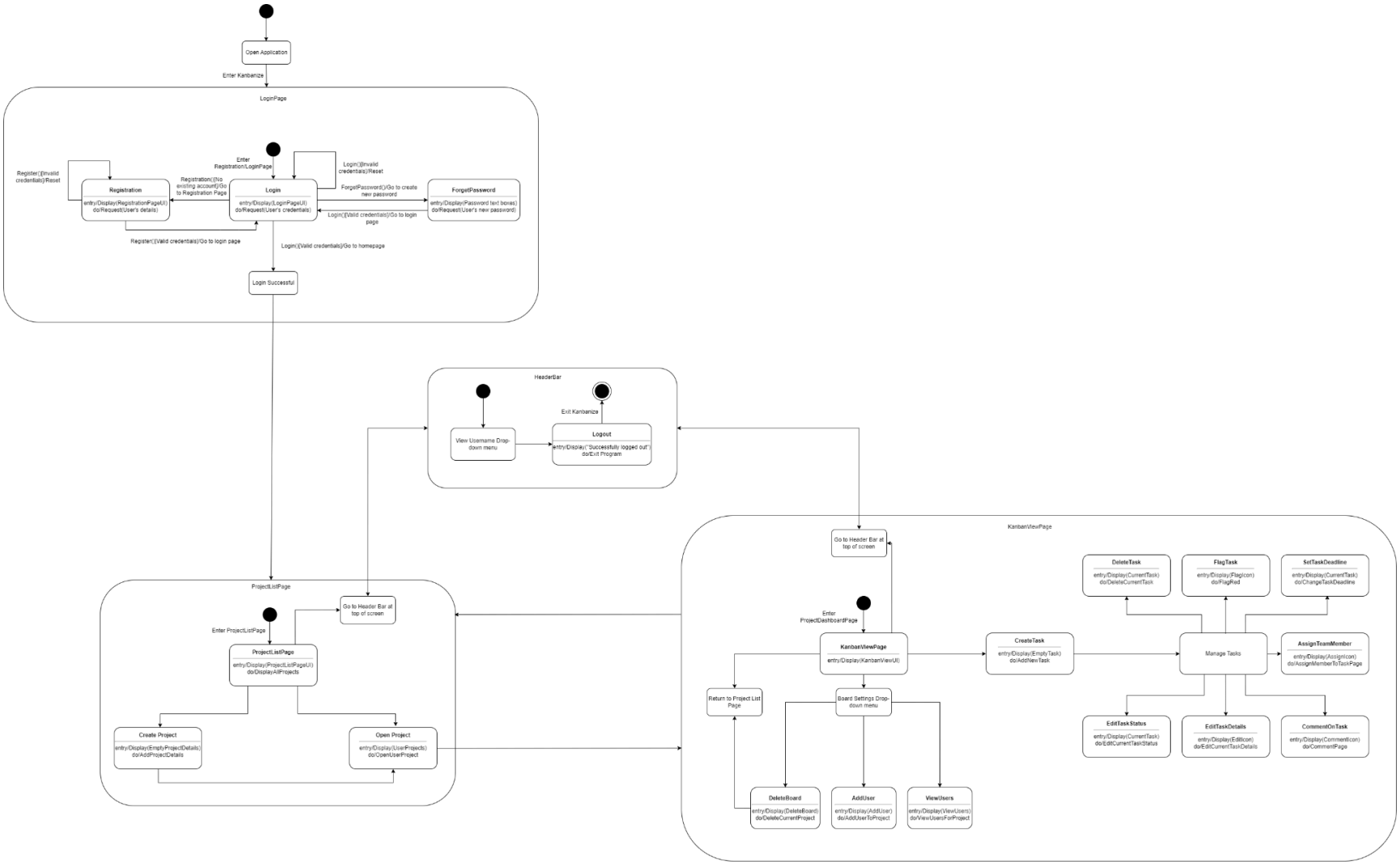
Class Diagram



Key Boundary and Control Classes



Dialog Map


Open Application
Enter Kanbanize
LoginPage
Enter
Registration/LoginPage
Login()[Invalid credentials]/Reset
Register()[Invalid credentials]/Reset
Registration
entry/Display(RegistrationPageUI)
do/RequestUser's details)
Registration()[No existing account]/Go to Registration Page
Login
entry/Display(LoginPageUI)
do/RequestUser's credentials)
ForgetPassword()/Go to create new password
ForgotPassword
entry/Display(Password text boxes)
do/RequestUser's new password)
Login()[Valid credentials]/Go to login page
Register()[Valid credentials]/Go to login page
Login()[Valid credentials]/Go to homepage
Login Successful
HeaderBar
Exit Kanbanize
View Username Drop-down menu
Logout
entry/Display("Successfully logged out")
do/Exit Program
ProjectListPage
Enter ProjectListPage
Go to Header Bar at top of screen
ProjectListPage
entry/Display(ProjectListPageUI)
do/DisplayAllProjects
Create Project
entry/Display(EmptyProjectDetails)
do/AddProjectDetails
Open Project
entry/Display(UserProjects)
do/OpenUserProject
KanbanViewPage
Go to Header Bar at top of screen
Enter ProjectDashboardPage
KanbanViewPage
entry/Display(KanbanViewUI)
CreateTask
entry/Display(EmptyTask)
do/AddNewTask
Manage Tasks
DeleteTask
entry/Display(CurrentTask)
do/DeleteCurrentTask
FlagTask
entry/Display(FlagIcon)
do/FlagRed
SetTaskDeadline
entry/Display(CurrentTask)
do/ChangeTaskDeadline
AssignTeamMember
entry/Display(AssignIcon)
do/AssignMemberToTaskPage
EditTaskStatus
entry/Display(CurrentTask)
do/EditCurrentTaskStatus
EditTaskDetails
entry/Display(EditIcon)
do/EditCurrentTaskDetails
CommentOnTask
entry/Display(CommentIcon)
do/CommentPage
Return to Project List Page
Board Settings Drop-down menu
DeleteBoard
entry/Display(DeleteBoard)
do/DeleteCurrentProject
AddUser
entry/Display(AddUser)
do/AddUserToProject
ViewUsers
entry/Display(ViewUsers)
do/ViewUsersForProject

System Architecture

Application Skeleton