CSE 360 Project Report Number 5

Team Tu57

Team Member Names:

1. name
2. Andy Vu
3. name
4. Ven Pattela
5. Vikas Mejari

Instructions: Replace the <Two Letters and Number> above with the team letter and number (e.g., Tu05). Replace each of the "name" items above with the names of the team members. Remove any unneeded numbers if there are fewer than six team members. Remove these instructions.

On these pages, remove information that is not required for this submission. (If there are only four members in the team there should only be 4 names above and if items below are tied to individual team members, reduce the items to the appropriate number.

Table of Contents

Table of Contents

1. Key Functions: Description, User Stories, and Use Cases

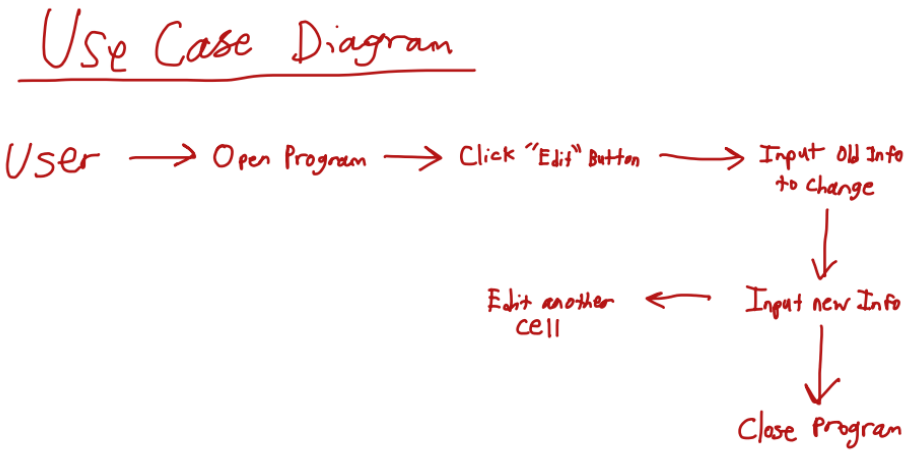
1.1. Edit Data

1.1. Vikas Mejari

1.1. This function is meant to allow Planning Poker Players to edit data within the logs, namely the project names and keywords of those logs. The User IDs are not able to be edited.

1.1. User Story 1: As a Planning Poker Player, I want to edit the data we created in our session in case information is inputted incorrectly.

User Story 2: As a Planning Poker Player, I want to be able to edit the data from past logs to reflect relevant information that I may have access to in the future but not in the present moment.



1.1.

1.2. Delete Data

1.23 Venkata Pattela

1.24. This JavaFX application, named "DeleteCode," implements a basic user management system with a graphical user interface. Upon clicking the button, the application attempts to delete the specified user from the database and displays a corresponding informational alert. The code follows a modular structure, utilizing JavaFX for the graphical components and adheres to good programming practices, including input validation and a clear separation of concerns.

1.25

User Story 1:

"As a registered user, I want the ability to delete my account, so that I can permanently remove my personal information from the system. Upon accessing the account settings, I expect to find a 'Delete Account' option, which, when selected, prompts me to confirm my decision. After confirming, the system should securely erase my account data, providing a confirmation message to ensure the successful deletion of my account."

User Story 2:

"As an administrator, I want to be able to remove inactive users from the system, so that the user database remains up-to-date and secure. In the administrator dashboard, I anticipate finding a 'Manage Users' section where I can view a list of users. For each user, I should have the option to 'Delete User,' which, when selected, triggers a confirmation prompt. Upon confirmation, the system should remove the user's account and notify me of the successful deletion for audit purposes."

Use Case Identification

1.1. Use Case ID: 1.1

1.2. Use Case Name: Place an Online Order

1.3. Traceability: This use case implements Requirements R1.1 (User should be able to place an order) and R1.2 (System should confirm the order).

1.4. Actor: Customer

1.5. Description: This use case enables customers to place an online order for products. It involves selecting items, providing shipping details, and confirming the order.

1.6. Preconditions:

Customer is logged into their account.

The customer has items in their shopping cart.

1.7. Postconditions:

The order is confirmed and processed.

The customer receives an order confirmation email.

1.8. Primary Pathway:

Customer selects "Checkout" from the shopping cart.

System prompts the customer to confirm their order.

Customer provides shipping details.

System calculates shipping costs and displays the total.

Customer confirms the order.

System processes the order, deducts the payment, and sends an order confirmation email.

1.9. Alternate Pathways:

1.9.1: Customer modifies the order during checkout.

Customer clicks on "Modify Order."

System allows the customer to add/remove items.

Customer continues with the modified order.

1.10. Exception Pathways:

1.10.1: Customer attempts to place an order with an empty shopping cart.

System displays an error message.

Use case terminates.

1.10.2: Payment transaction fails during order processing.

System prompts the customer to use an alternative payment method.

If the issue persists, the system cancels the order.

1.11. Notes and Issues:

The system should provide a user-friendly interface for ease of order placement.

TBD: Integration with a third-party payment gateway needs to be finalized by the end of the development sprint.

1.3. Add Data

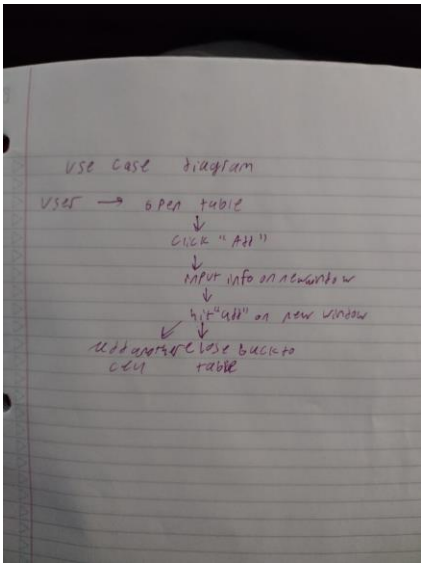
1.3. Vedant Kaushik

1.3. This function allows a user to add data to an existing set of data that is later to be used in planning poker. A user can input a userID, project name and a key word that is to be linked to the project and user.

1.3. User Stories

As a planning poker user, I want to be able to add new projects to the data set so that they can be called when intiating a planning poker session. Below the table that has all of the users, there will be an “add user” button that prompts the user to input a userID, project name and a key word that is to be linked to the project and user. This is then shown on the original list with the new cell being added to the table

1.3. Use Cases



1.4. View Data

1.4. Andrew Boban

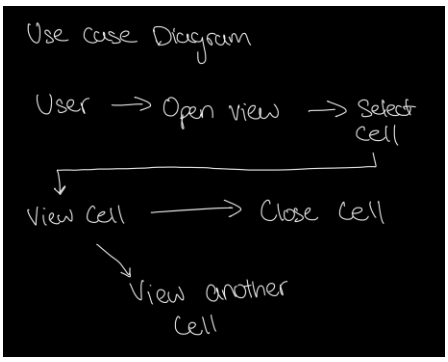
1.4. The function allows for the program user to view the cells that currently contain information in, when using the view function/button it opens the cells and allows for easy navigation of the currently saved projects that are stored in the system.

Example of a User Story:

"As a registered user, I want the ability to view my account information, so that I can access and review the personal details associated with my account. Upon accessing the account settings, I expect to find an 'Account Information' option, which, when selected, displays all the details tied to my account. This access should provide a comprehensive overview of my information, ensuring I can review and verify the accuracy of the data stored."

1.4.

1.4. Use Cases



This is a possible use case diagram for using the view fucntion, it will mainly be used for checking cells and confirming saved data that is on the database.

1.5. Search with Keyword

1.5. Andy Vu

1.5. This function allows a user to input a keyword into a search bar, and then displays all the historical projects that contain the searched keyword.

1.5. A user should be able to search a keyword to filter out specific historical projects that have that keyword to find historical projects similar to the current project.

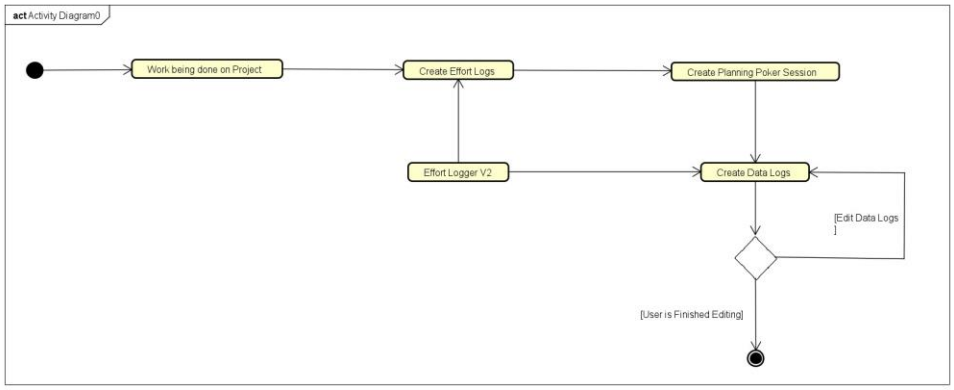
1.5. Main Success Scenario: A user enters a keyword into a search bar. The application will the display only the historical projects that contains the searched keyword for the user.

2. UML Design Details

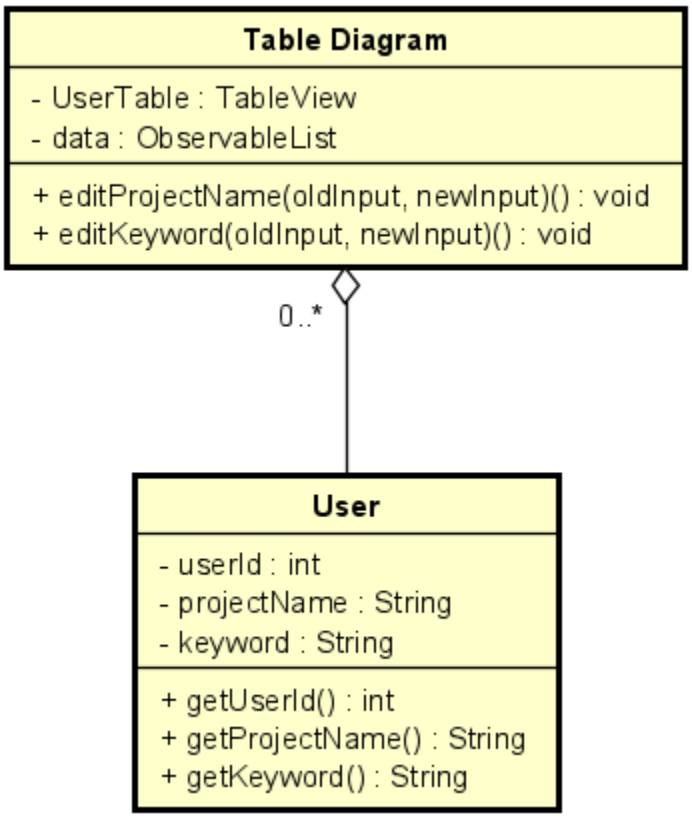
2.1. Edit Data

2.1. Vikas Mejari

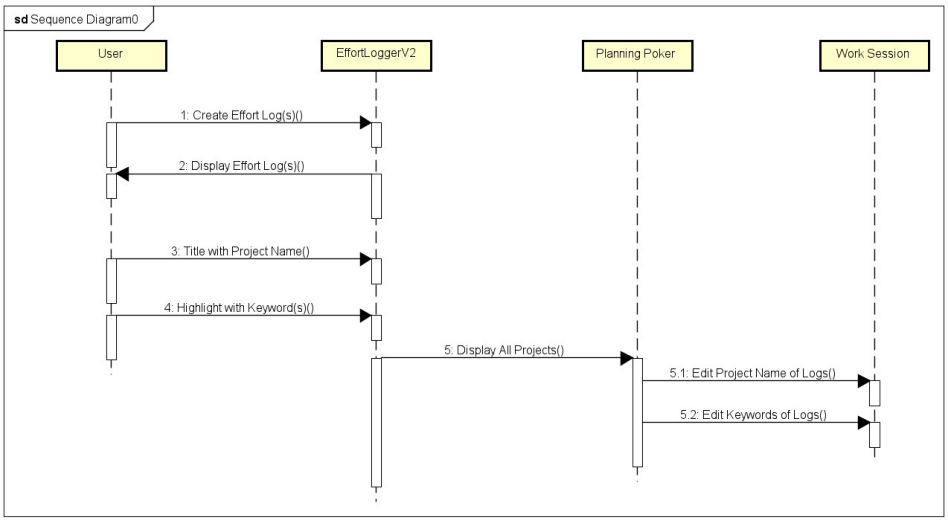
2.1. Activity Diagram



2.1. Class Diagram



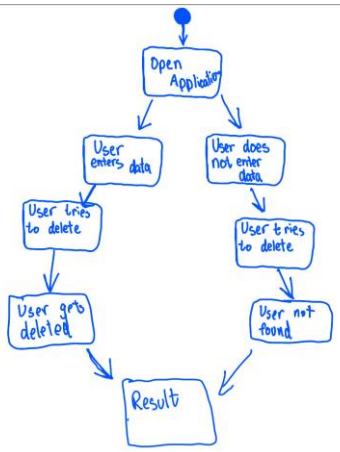
2.1. Sequence Diagram



2.2. Delete Data

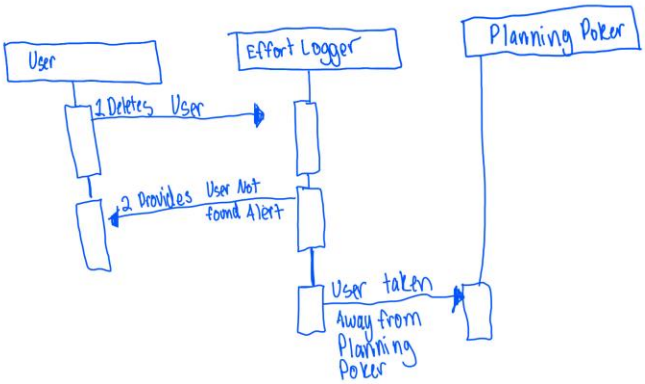
2.3. Venkata Pattela

2.3. Activity Diagram



2.3. Class Diagram

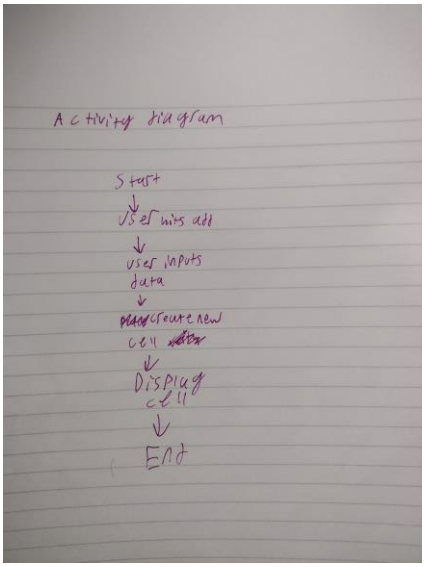
2.3. Sequence Diagram



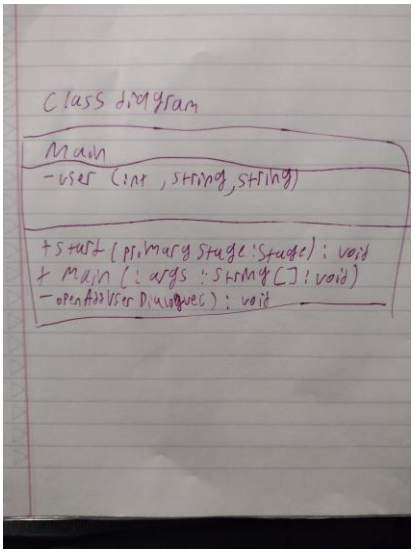
2.4. Add Data

2.4. Vedant Kaushik

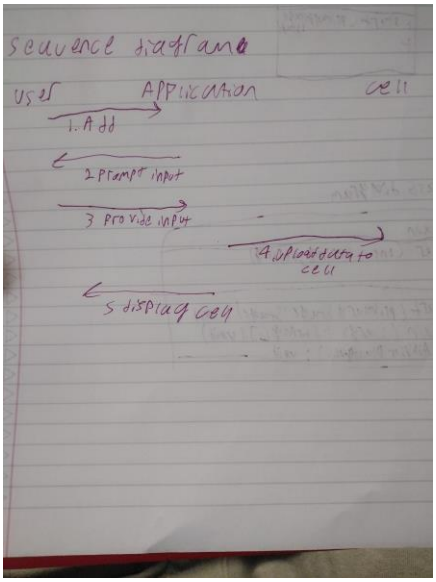
2.4. Activity Diagram



2.4. Class Diagram



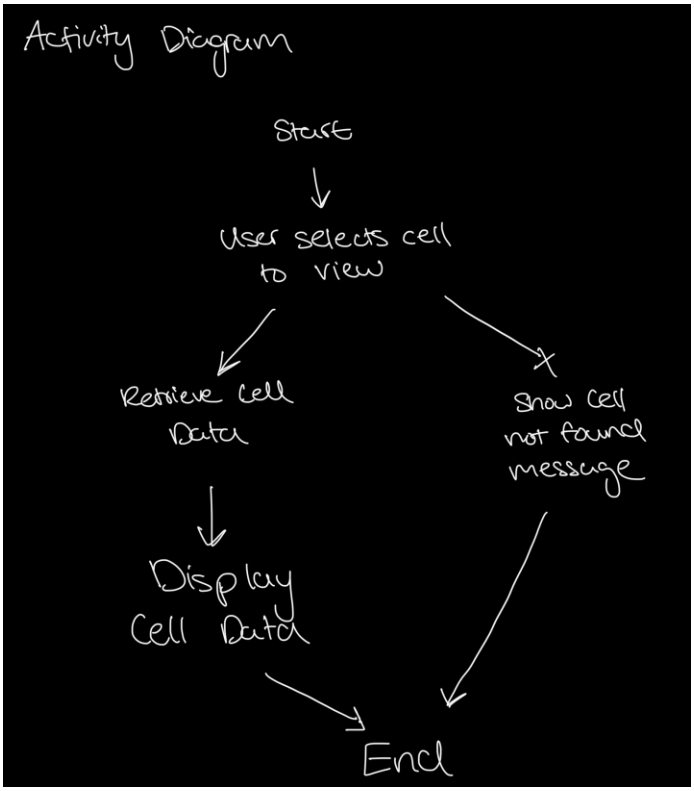
2.4. Sequence Diagram



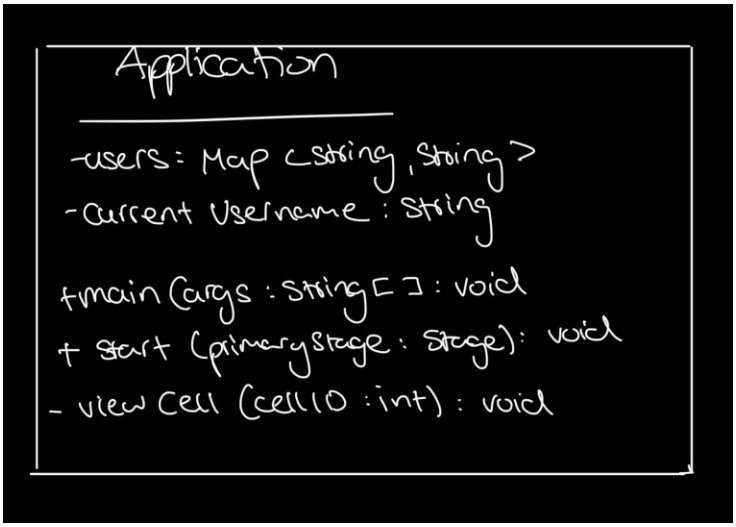
2.5. View Data

2.5. Andrew Boban

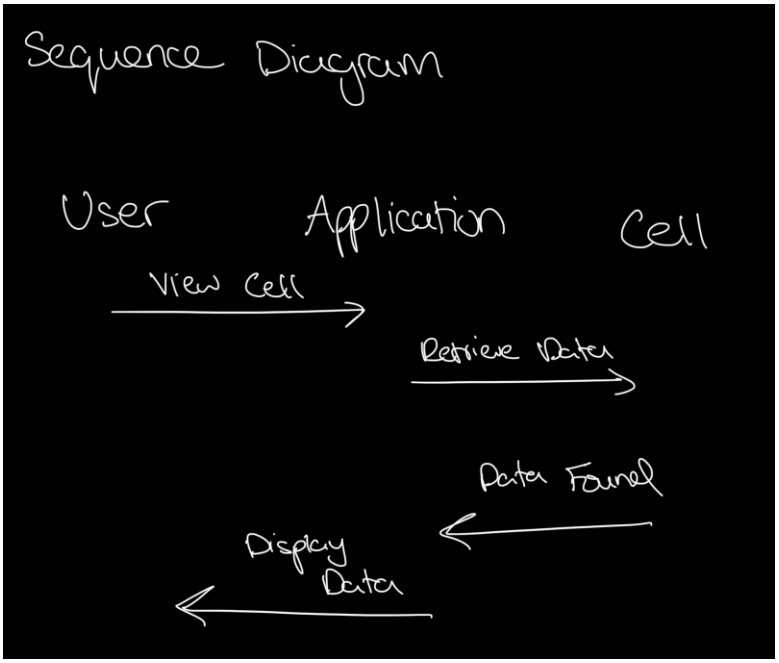
2.5. Activity Diagram



2.5. Class Diagram



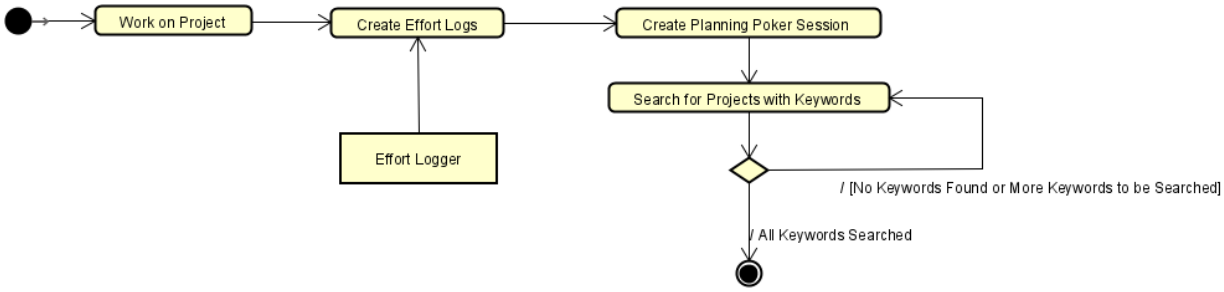
2.5. Sequence Diagram



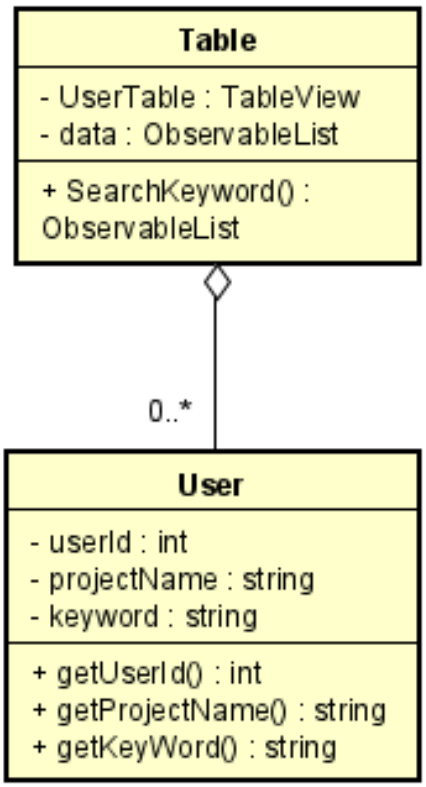
2.6. Search with Keyword

2.6. Andy Vu

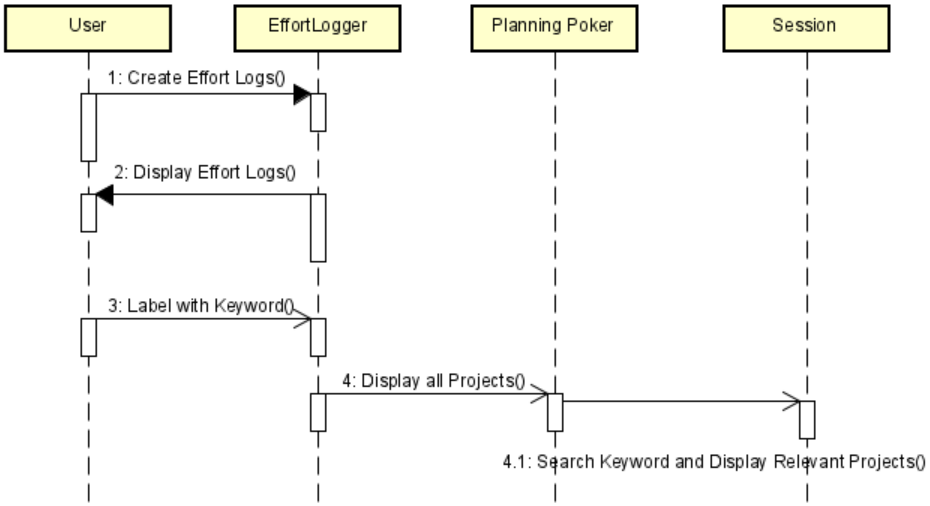
2.6. Activity Diagram



2.6. Class Diagram



2.6. Sequence Diagram



3. Screencasts

3.1. Screencast for Edit Data

3.1. Vikas Mejari

3.1. The screencast shows the different user stories/use cases and presents each of the corresponding diagrams to explain them. Afterwards, the program code is shown to display how the edit data function would work if it was fully implemented into the application.

3.1. https://drive.google.com/file/d/1HNmt7c3zkRDfoDK0Ykd8B8pAaEKu8nm6/view?usp=sharing

3.2. Screencast for Delete Data

3.3. Venkata Pattela

3.4. I showed my code with the explanations and then I made sure that I showed the demo.

https://asu.zoom.us/rec/share/FRzaCaeVm78HU1tM9K311V1A-Rk5vcrj7tjQNQWHzrv4kdy0Q9GcFgtUwxh7mxgo.ib_FRr8q8GBA4Ndj?startTime=1699851786000

Passcode: @eYU5y3f

3.5. Screencast for Add Data

3.5. Vedant Kaushik

3.5. I start by showing the code and how each part of the code works, and then move to demonstrating the functionality of the add button on my list.

3.5. https://drive.google.com/file/d/1BfSgnmBjPsTTENAYAOHgHHnM3m1Lj-2I/view?usp=sharing

3.6. Screencast for View Data

3.6. Andrew Boban

3.6. The screencast shows the implementation of the view function, initially starting with the code and explaining each section, it then goes to showing off example solutions of the program working, there were some issues with making javafx work on my machine, so it is samples ran by the code I have showcased on a different computer.

3.6. URL to the Screencast

https://drive.google.com/file/d/1sBFY5P1YI3pAOpBJLm5KrbktTCq3-FUJ/view?usp=sharing

3.7. Screencast for Search with Keyword

3.7. Andy Vu

3.7. The screencast starts by presenting the code for the implementation of the keyword search function, then shows the keyword search function being used. It then shows the UML activity, class, and sequence diagrams for the function.

3.7. URL to the Screencast

https://drive.google.com/file/d/1nZsArA15A4wnrS1Ypl638ug5eQlv_T0S/view?usp=sharing

4. Conclusion

4.1. The most important conclusions

4.1.1. Data Retrieval

- Retrieving data from Effort Logger for Planning Poker sessions is an extremely important feature, and one that must be continuously worked on.
- Adding more functions that would allow for users to be more selective in their ability to retrieve data would be helpful as well. This includes users being able to search for multiple keywords, or being able to search for projects that don't contain keywords.

4.1.2. Security and Privacy

- Going forward, making sure that the data being displayed is only displayed to those who have permission to see it is important. This means that users should not be able to see project information that isn't their own, or that supervisors should be able to see user data at all.
- Making sure that the data flow is secure is also important. Vulnerabilities in the data flow could be leveraged by hackers, who can then gain access to sensitive information.

4.1.3. Scalability

- Making sure that these functions are still working when many people are using the application, or that these functions are still working when the data size becomes extremely large is important in making sure the product is scalable.
- Being able to add additional functions without compromising the integrity of existing functions is also important.

4.2. Upcoming Important Activities

4.2.1. More Planning Poker Functions

- Adding a planning poker summary card which would display all the necessary information in a concise manner would be helpful to users.
- Adding a feature to search or display projects based on story weight or user stories that are associated with the projects would also be important.

4.2.2. Team Implementation

- Being able to access and retrieve data shared with the entire team would be important in implementing an Effort Logger application created for team projects.
- Making sure that privacy works for teams is also important.

4.2.3. Project Descriptions

- Having more in-depth descriptions during planning poker would allow for users to understand which historical projects are relevant to the current project.
- Adding a search function for descriptions would also add to this, giving users the ability to search for more relevant projects.

4.3. Parking Lot for Items we need to address moving forward

4.3.1. Displaying Information for Supervisors

- Making sure that supervisors also see important data would be helpful as well.
- Guaranteeing that supervisors don't see private information is an important thing to remember when implementing this.

4.3.2. Additional Comprehensiveness for Projects

- Adding additional information that can be displayed for projects would allow for users to understand the relevancy of historical projects.

- This information would also have to be made secure and private.

4.3.3. Defect Information being Displayed

- Displaying defect information would help users during planning poker sessions greatly.
- Searching based off of defect information could also be helpful to users.

5. Appendix A: Credit Sheet

Team Member Name	Contributions
Andrew Boban	Did the implementation for the viewing function, also completed all the necessary diagrams associated with said function
Andy Vu	I did the implementation of the search for keyword function, as well as all the UML diagrams and the screencast for the function.
Vedant Kaushik	I did the implementation of the add function along with the use case, UML, activity diagrams and the screen cast for the function.
Venkata Pattela	I implemented the delete function along with its use case, UML, activity diagrams
Vikas Mejari	I implemented the edit data function, as well as created user stories/use cases, UML diagrams, and did a screencast explaining these diagrams and my edit data function.
Team Member 6 Name	Line 1 Line 2 Line 3 Line 4 Line 5 Line 6

6. Appendix A: Current Team Norms

This appendix is an updated version of the original Team Norms contract the team has created as well as a URL to an ASU Google Drive PDF of the signed copy. The graders must be able to access that PDF!

Google docs:

https://drive.google.com/drive/folders/1Hv4OvU6nwKkErbRh6-vAesYKOrZsj7vy?usp=sharing