

CSE 360 Project Report Number 5

Team <Tu37>

Team Member Names:

1. Trevor Huss
2. Alma Babbitt
3. Ishan Yelnoorkar
4. Karryl Dumalag
5. Zachary Litwin

Table of Contents

Table of Contents.....	ii
1. Key Functions: Description, User Stories, and Use Cases.....	1
1.1. Key Function 1.....	1
1.1.1. Author of the Function.....	1
1.1.2. Description of the Function.....	1
1.1.3. Author of the Function.....	1
1.1.4. Description of the Function User Stories.....	1
1.1.5. Use Cases.....	1
1.2. Key Function 2.....	1
1.2.1. Author of the Function.....	1
1.2.2. Description of the Function.....	1
1.2.3. User Stories.....	1
1.2.4. Use Cases.....	1
1.3. Key Function 3.....	1
1.3.1. Author of the Function.....	1
1.3.2. Description of the Function.....	1
1.3.3. User Stories.....	1
1.3.4. Use Cases.....	1
1.4. Key Function 4.....	1
1.4.1. Author of the Function.....	1
1.4.2. Description of the Function.....	1
1.4.3. User Stories.....	1
1.4.4. Use Cases.....	1
1.5. Key Function 5.....	1
1.5.1. Author of the Function.....	1

1.5.2.	Description of the Function.....	1
1.5.3.	User Stories.....	1
1.5.4.	Use Cases.....	1
1.6.	Key Function 6.....	2
1.6.1.	Author of the Function.....	2
1.6.2.	Description of the Function.....	2
1.6.3.	User Stories.....	2
1.6.4.	Use Cases.....	2
2.	UML Design Details.....	3
2.1.	Key Function 1.....	3
2.1.1.	Author of the Function.....	3
2.1.2.	Activity Diagram.....	3
2.1.3.	Class Diagram.....	3
2.1.4.	Sequence Diagram.....	3
2.2.	Key Function 2.....	3
2.2.1.	Author of the Function.....	3
2.2.2.	Activity Diagram.....	3
2.2.3.	Class Diagram.....	3
2.2.4.	Sequence Diagram.....	3
2.3.	Key Function 3.....	3
2.3.1.	Author of the Function.....	3
2.3.2.	Activity Diagram.....	3
2.3.3.	Class Diagram.....	3
2.3.4.	Sequence Diagram.....	3
2.4.	Key Function 4.....	3
2.4.1.	Author of the Function.....	3
2.4.2.	Activity Diagram.....	3

2.4.3.	Class Diagram.....	3
2.4.4.	Sequence Diagram.....	3
2.5.	Key Function 5.....	3
2.5.1.	Author of the Function.....	3
2.5.2.	Activity Diagram.....	3
2.5.3.	Class Diagram.....	3
2.5.4.	Sequence Diagram.....	3
2.6.	Key Function 6.....	3
2.6.1.	Author of the Function.....	4
2.6.2.	Activity Diagram.....	4
2.6.3.	Class Diagram.....	4
2.6.4.	Sequence Diagram.....	4
3.	Screenshots.....	5
3.1.	Screenshot for Key Function 1.....	5
3.1.1.	Author of the Screenshot.....	5
3.1.2.	Overview Outline.....	5
3.1.3.	URL to the Screenshot.....	5
3.2.	Screenshot for Key Function 2.....	5
3.2.1.	Author of the Screenshot.....	5
3.2.2.	Overview Outline.....	5
3.2.3.	URL to the Screenshot.....	5
3.3.	Screenshot for Key Function 3.....	5
3.3.1.	Author of the Screenshot.....	5
3.3.2.	Overview Outline.....	5
3.3.3.	URL to the Screenshot.....	5
3.4.	Screenshot for Key Function 4.....	5
3.4.1.	Author of the Screenshot.....	5

Team Project Phase 5
Table of Contents

3.4.2.	Overview Outline.....	5
3.4.3.	URL to the Screencast.....	5
3.5.	Screencast for Key Function 5.....	5
3.5.1.	Author of the Screencast.....	5
3.5.2.	Overview Outline.....	5
3.5.3.	URL to the Screencast.....	5
3.6.	Screencast for Key Function 6.....	5
3.6.1.	Author of the Screencast.....	5
3.6.2.	Overview Outline.....	5
3.6.3.	URL to the Screencast.....	5
4.	Conclusion.....	7
4.1.	The most important conclusions.....	7
4.2.	Upcoming Important Activities.....	7
4.3.	Parking Lot for Items we need to address moving forward.....	7
5.	Appendix A: Credit Sheet.....	8
6.	Appendix A: Current Team Norms.....	9

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

1.1. Key Function 1: Planning Poker Flow

1.1.1. Author of the Function: Zachary Litwin

1.1.2. Description of the Function: This creates the logical flow of a planning poker session. It gets the number of people present in the session and each of their estimates and will return the maximum and minimum outliers

1.1.3. User Stories:

- As a Scrum Master, I want to initiate a planning poker session with a specific number of participants so that the team can collectively estimate the effort required for upcoming user stories or tasks during the sprint planning meeting.
- As a Team Member, I want to provide my estimation for a user story or task in the planning poker session so that the team can reach a consensus on the effort required.
- As a Scrum Master, I want the ability to reset the planning poker session so that the team can start a new round of estimation, ensuring a fresh start for each set of user stories or tasks.
- As a Participant, I want to receive feedback if I enter an invalid card value during the planning poker session so that I can provide an estimate within the predefined set of allowed values.

1.1.4. Use Cases:

- Use Case Name: Planning Poker Session
- Actor: Participant - A participant, acting as a team member in an Agile development environment, interacts with the system to contribute to a planning poker session. The planning poker session aims to estimate the effort required for a set of user stories or tasks.
- Preconditions: The planning poker session is initiated by the Scrum Master, and the team is assembled and ready for estimation.
- Postconditions: The participant's card value is recorded, and the planning poker session progresses to the next story or task.
- Primary Pathway:
 1. Participants are present with the Scrum Master
 2. Scrum Master initiates the planning poker session and presents a user story or task for estimation.
 3. The participant selects "Give Card Value" for the presented item. |
 4. The system prompts the participant to input their estimate by selecting a card from a predefined set.
 5. Participants enter a valid card value corresponding to their estimate.
 6. The system records the participants' estimates and then checks the estimates and displays the minimum and maximum outliers.
 7. The estimates match and the team moves on to the next item or ends the session

Key Functions: Description, User Stories, and Use Cases

- Alternate Pathways:
 - Participants view previous estimates for context before entering a new estimate.
 - The outliers do not match and a discussion then revote occurs
- Exception Pathways:
 - Planning poker session not initiated or not in progress.
 - Technical issues preventing the recording of the estimate. Example invalid options chosen
- Notes and Issues:
 - The use case focuses on the estimation aspect of a planning poker session.
 - A planning poker tool or application may vary in specific features and interface.
 - It assumes that the participant is familiar with the Agile planning poker process.

1.2. Key Function 2: Effort Logger Integration with Planning Poker

1.2.1. Author of the Function: Ishan Yelnoorkar

1.2.2. Description of the Function: This function allows a user to perform planning poker. After logging in an effort entry or a defect entry, the user can proceed to perform planning poker. The integration of the two streamlines the process of estimating and tracking work efforts in a project. This allows a team to improve their project planning and execution.

1.2.3. User Stories:

- As a manager, I want to have an integrated planning poker with the effort logger so that I can easily transfer estimation data to track real-time effort spent on tasks
- As a team member, I want to view historical data from previous planning poker sessions and compare it with my effort log so that I can improve my estimation accuracy
- As a Scrum master, I want to synchronize the planning poker estimate with the effort logger to maintain a single source of truth for task estimation and actual effort log.

1.2.4. Use Cases:

- Effort Logger Integration with Planning Poker
- Scope- Effort Logger, Planning Poker tool
- Level- User goal
- Primary actor- Team members
- Stakeholders and interests
 - Team members- Want to have access to a collaborative, streamlined planning poker experience to improve user story estimation.
 - Project Manager- Want an effective planning poker session so that the team can better discuss and reach a consensus for backlog estimation.
- Preconditions- The user is authorized to access the effort logger and attend a planning poker session.
- Success guarantee (Postcondition)- The estimation process is streamlined; the team members were able to collaborate effectively

Key Functions: Description, User Stories, and Use Cases

- Primary Pathway
 - i. A team member logs their effort data and that data is used to initiate planning poker
 - ii. The team initiates the planning poker session for discussing new user stories or old user stories and setting up a collaborative estimation.
 - ii. The team members load relevant past projects and data within the planning poker tool. This allows for informed decision-making during the estimation process.
 - iii. The planning poker tool has an intuitive and user-friendly interface designed to facilitate estimating story points, these elements include elements such as visual poker cards representing different point values, a place to input criteria, and item details for context.
 - iv. Team members enter an estimated number of story points for the backlog item under consideration.
 - v. Effort Logger has recorded the estimates and presented the cards on the interface and the associated data for clearer visibility.
 - vi. The planning poker tool points out outliers, they are highlighted and prompted to discuss their response.
 - vii. The process is repeated by the team to refine the estimates and ensure all member's perspective is considered.
 - viii. Final Estimations are recorded and discussed, these estimations are stored in a database, and the data also has been encrypted.
 - ix. Collaboration takes place and the team reaches a consensus on backlog items and story points.
- Alternate pathways- The team directly moves on to planning poker to discuss relevant effort requirements for a user story
- Exception pathways- Planning poker is not initiated, and relevant project data is not available. Or unexpected technical error that prevents the recording of estimation points.

1.3. Key Function 3: Planning Poker Button Functionality

1.3.1. Author of the Function: Karryl Dumalag

1.3.2. Description of the Function: This function provides a structured and organized interface, making it easier for users to understand the flow of the application. By incorporating buttons, a streamlined workflow reduces information overload and allows users to focus on one stage at a time.

1.3.3. User Stories:

Key Functions: Description, User Stories, and Use Cases

- As a team member, I want to be able to view and understand the user stories or tasks assigned to me for estimation, provide my effort estimates, and view the collective estimation results in discussions.
- As a Product Owner, I want to create and prioritize user stories for estimation by the development team, review and accept/reject estimation outcomes based on project priorities, and provide clarifications during sessions.
- As a Scrum Master, I want to initiate and schedule Planning Poker sessions for upcoming sprints, and capture and store results of estimation sessions for future reference.
- As a manager, I want to access the Planning Poker tool to oversee, review, facilitate, and generate reports on the team's effort assessments.

1.3.4. Use Cases:

- Use Case Name: Planning Poker Sessions
- Actors:
 - Dev team
 - Scrum master
 - Product owners
- Description: The Planning Poker program facilitates Agile teams in estimating the effort required for project criteria, fostering collaboration and discussion among team members
- Preconditions:
 - The planning poker program is installed and accessible to all team members
 - All members are connected to the same session
- Postconditions:
 - Estimated and reasoning for each team member are recorded.
 - Outlying Estimates, common estimates, and other variables are populated
- Mainline:
 - Opening page
 - Upon launching, the program displays multiple criteria options for the project
 - Team members can connect in real-time to the planning session
 - Selecting criteria
 - Members select a criterion they want to estimate
 - They are presented with cards representing different weights for the selected criteria
 - Playing Planning Poker
 - Members individually select a card representing their estimate for the criteria
 - Each team member writes down their reasoning and details for their estimate
 - Criteria details are visible to all team members during this stage

Key Functions: Description, User Stories, and Use Cases

- Displaying Estimates
 - Outlying estimates: The system displays estimates that significantly deviate from the average, prompting discussion
 - Common estimates: The system shows estimates that align closely, indicating a consensus among team members
 - My Estimate: Each team member's estimate is displayed
- Alternatives
 - **Re-estimation:** If there is a significant variance in estimates, the team may choose to re-estimate the criterion after further discussion
 - **Adjusting Criteria:** Team members can request to adjust the project criteria if they believe it is necessary for a more accurate estimate
 - **Late Entries:** In case a team member joins the session late, they can catch up by reviewing previous estimates and providing their own
 - **Anonymous voting:** Depending on the team's preference, there may be an option for anonymous voting to encourage unbiased estimates
 - **Review and confirm:** Before finalizing estimates, team members can collectively review and confirm their choices

1.4. Key Function 4: SQL Database Flow

1.4.1. Author of the Function: Alma Babbitt

1.4.2. Description of the Function: This function creates the back end of this project. All user information will be stored with SQL and all information will be accessed with SQL.

1.4.3. User Stories:

- As an employee, I want my information stored accurately and securely so I can later review key items in the project.
- As a manager, I want employee efforts and projects saved in a reliable database that is more secure than using Excel sheets. I want this so I can better track the company's progress in releasing projects under a time constraint.

1.4.4. Use Cases:

Use Case 1: Planning Poker Data Management

Description: This use case involves storing and retrieving Planning Poker session data to ensure accurate and efficient estimation processes.

Actors:

1. **Participant:** The individual actively involved in the Planning Poker session, providing input and participating in the estimation process.

Key Functions: Description, User Stories, and Use Cases

2. **Facilitator:** The person responsible for managing the Planning Poker session, ensuring it runs smoothly, and resolving any issues that may arise.
3. **Administrator:** Overseeing the overall database and system, managing access controls, and ensuring the security of the Planning Poker data.

Steps:

1. The Participant logs into the system and initiates a new Planning Poker session.
2. The Facilitator joins the session and guides participants through the estimation process.
3. As participants provide estimates, the system stores the data securely in the SQL database.
4. The Facilitator can access and review past Planning Poker sessions for analysis and improvement.
5. The Administrator manages user roles, and database security, and ensures the overall integrity of the system.

Use Case 2: Transparent Data Flow

Description: This use case focuses on providing a user-friendly interface with privacy explanations and allowing users to provide explicit consent during various data-sharing points.

Actors:

1. **User:** The individual interacting with the system, providing data, and making decisions regarding data privacy.
2. **System:** The software and database responsible for presenting information, managing privacy mechanisms, and obtaining user consent.

Steps:

1. The User logs into the system and navigates to a page requiring data input.
2. The System displays tooltips and notification pop-ups explaining the privacy mechanisms associated with the data input.
3. The User has the option to provide explicit consent for data sharing during relevant points.
4. The System records and respects the privacy choices made by the User.
5. This process ensures transparency and empowers users to make informed decisions about their data.

Use Case 3: Access to Performance Reports

Description: This use case involves enabling authorized users and improvement teams to access and analyze performance data for individuals categorized into their working teams.

Key Functions: Description, User Stories, and Use Cases

Actors:

1. **Authorized User:** An individual with permission to access specific performance data within their working team.
2. **Improvement Team:** A group responsible for analyzing performance data and implementing improvements.
3. **Administrator:** Manages access controls, ensures data security, and oversees the overall system.

Steps:

2. The Authorized User logs into the system and navigates to the performance reports section.
3. The System verifies the user's credentials and grants access to the relevant data for their working team.
4. The Improvement Team uses tools provided by the system to analyze and visualize performance data.
5. The Administrator oversees access controls, ensuring that only authorized users can view and analyze specific performance data.
6. This use case supports a data-driven approach to improvement initiatives and facilitates informed decision-making by supervisors and improvement teams.

6.1. Key Function 5: Accessing Data as Employee/Manager

6.1.1. Author of the Function: Trevor Huss

6.1.2. Description of the Function: This function separates the data access from the database based on whether or not the user is a manager or an employee.

6.1.3. User Stories:

- As an employee, I want to access my individual performance metrics to track my progress so I can understand my contributions to the team and the company.
- As a manager, I need to access team performance data to monitor productivity so I can identify areas for improvement or recognition.
- As an employee, I want easy access to company-wide information, such as defects and project step progress so I can stay informed.

6.1.4. Use Cases:

- Use Case Name: Accessing Employee Data as a Manager
- Actors:
 - Manager, System
- Description:

Key Functions: Description, User Stories, and Use Cases

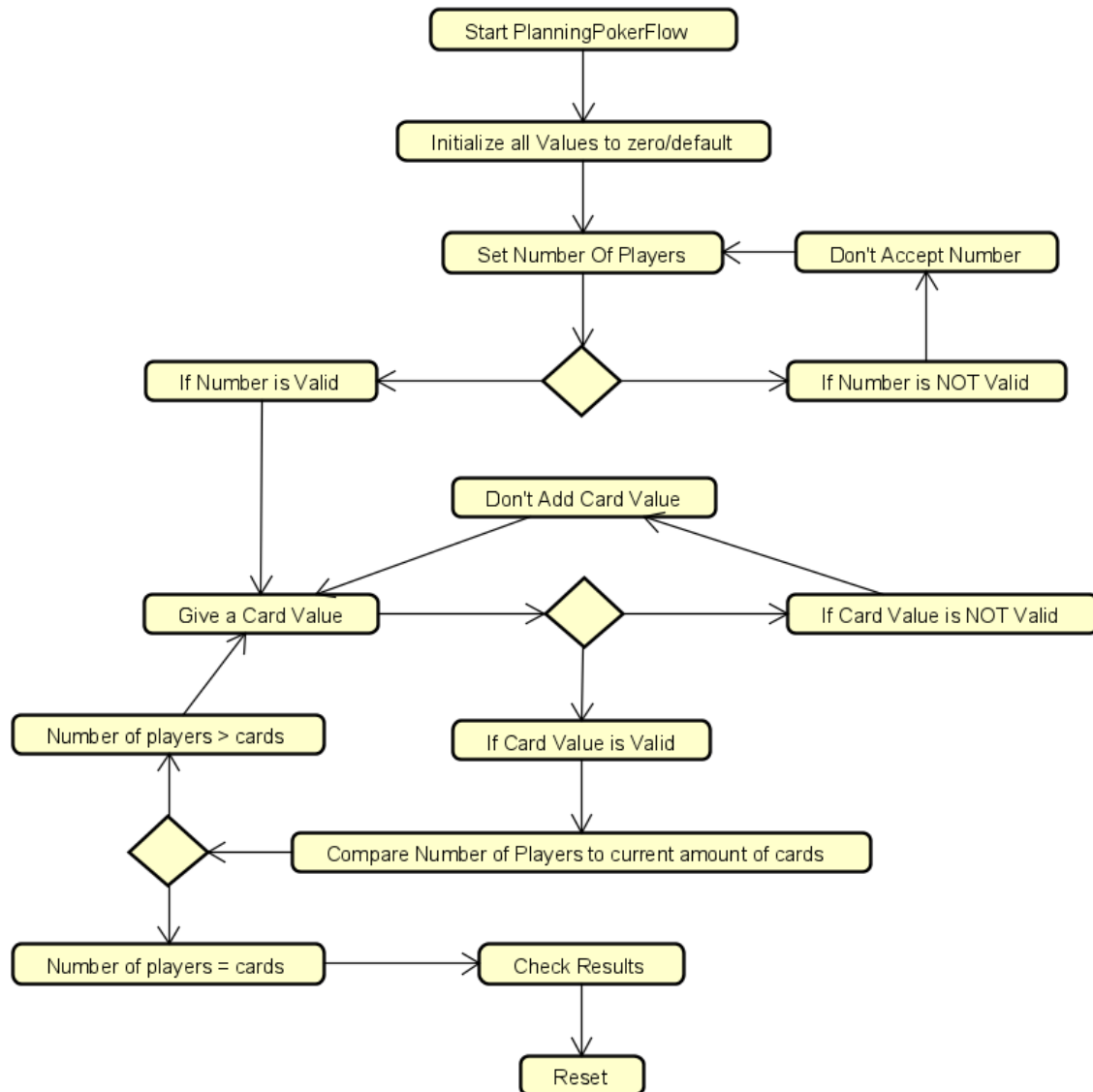
- This use case outlines the steps for a manager to access and review employee data within the company's database for performance evaluation, resource allocation, and decision-making.
- Preconditions:
 - The manager must have appropriate login credentials and authorization to access employee data.
 - The system must be operational and accessible.
- Postconditions:
 - The manager gains insights into employee performance and relevant data for making informed managerial decisions.
- Mainline:
 - The manager logs in with valid credentials.
 - The manager navigates to the database.
 - The manager retrieves desired data from the database.
 - The manager makes decisions based on the data.
- Alternatives:
 - The Effortlogger console is not working
 - The manager credentials are invalid
 - Data is not found in the database
- Notes:
 - This use case shows the difference between accessing data as a manager as opposed to an employee.

2. UML Design Details

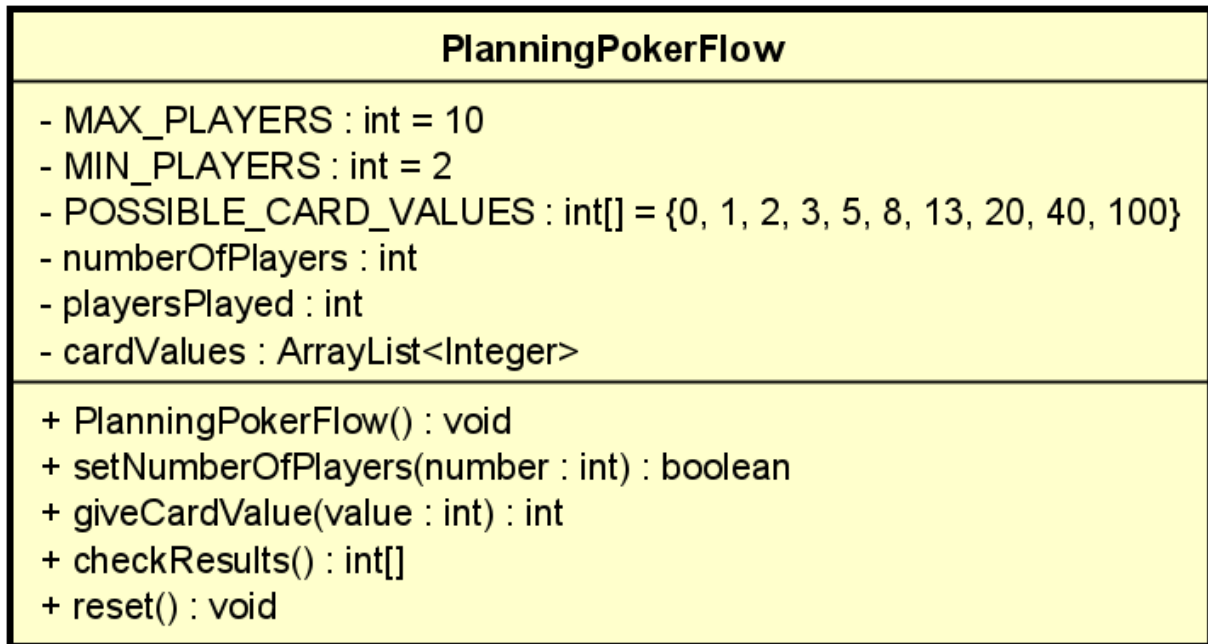
2.1. Key Function 1: Planning Poker Flow

2.1.1. Author of the Function: Zachary Litwin

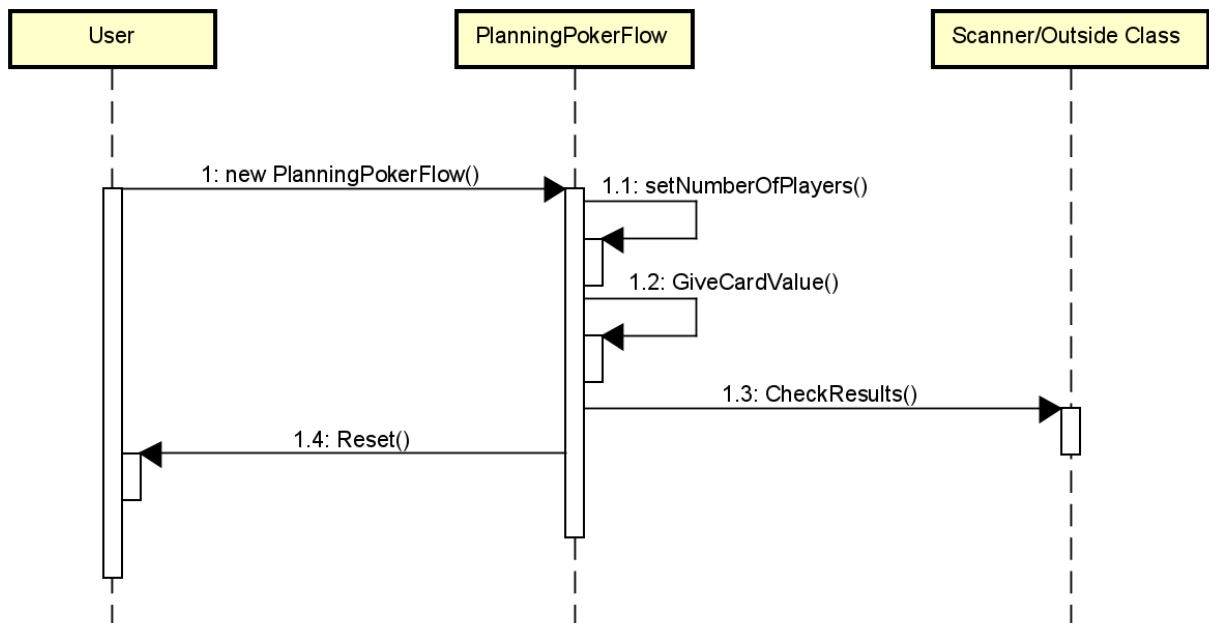
2.1.2. Activity Diagram:



2.1.3. Class Diagram:

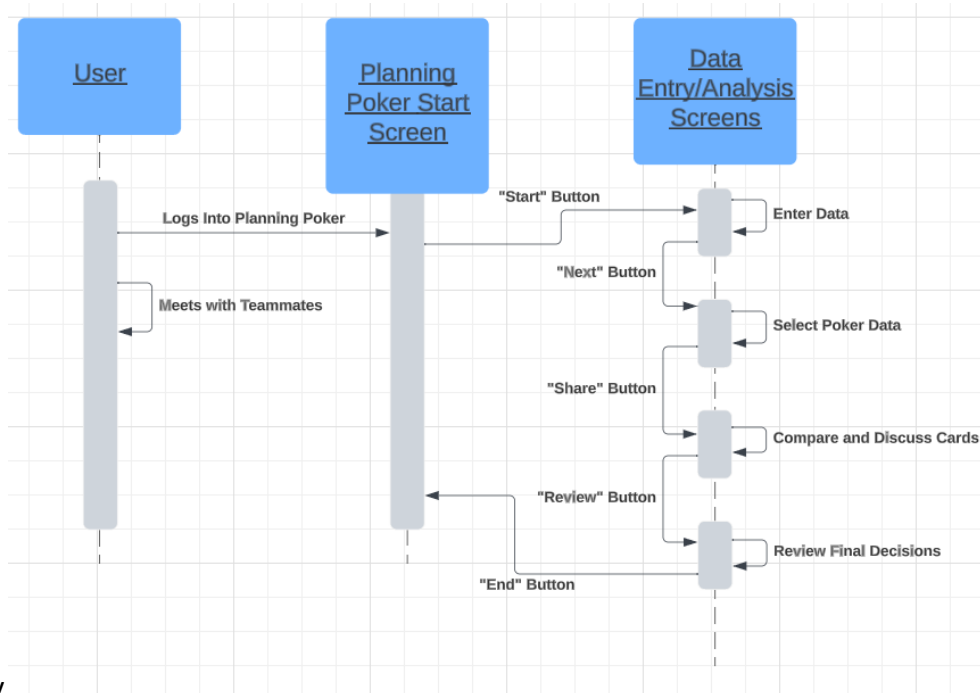


2.1.4. Sequence Diagram:

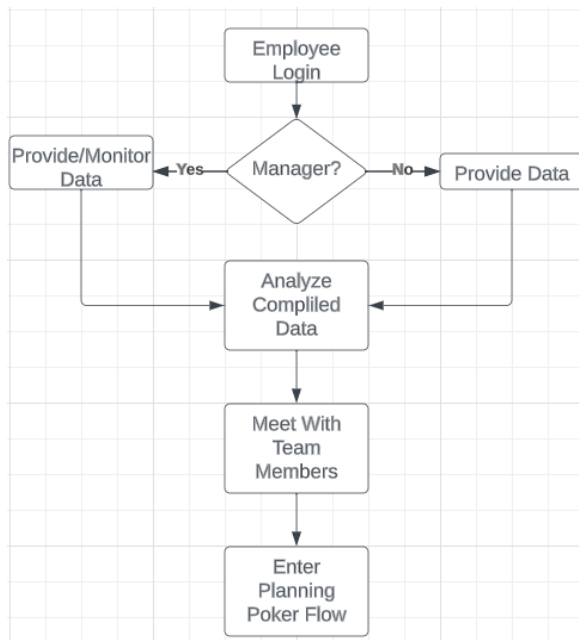


2.2. Key Function 2: Effort Logger Integration with Planning Poker

2.2.1. Author of the Function: Ishan Yelnoorkar

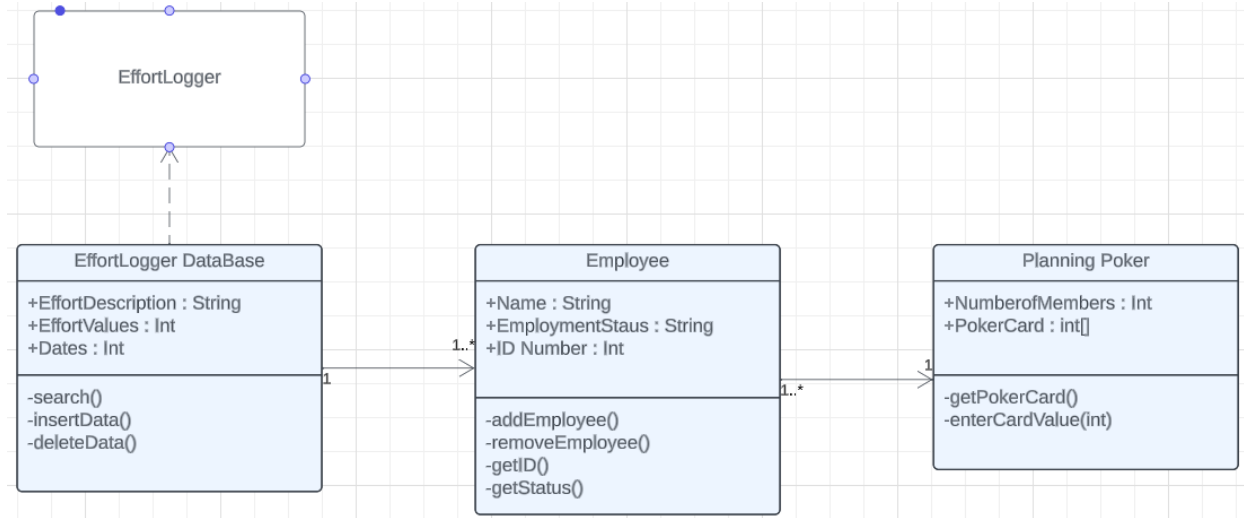


2.2.2. Activity Diagram:

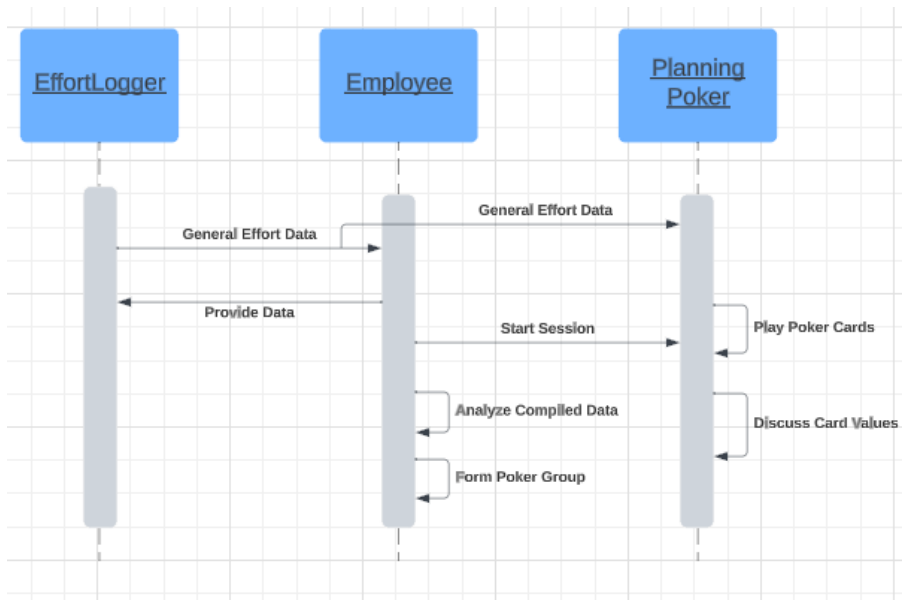


2.2.3. Class Diagram:

Team Project Phase 5
UML Design Details



2.2.4. Sequence Diagram:

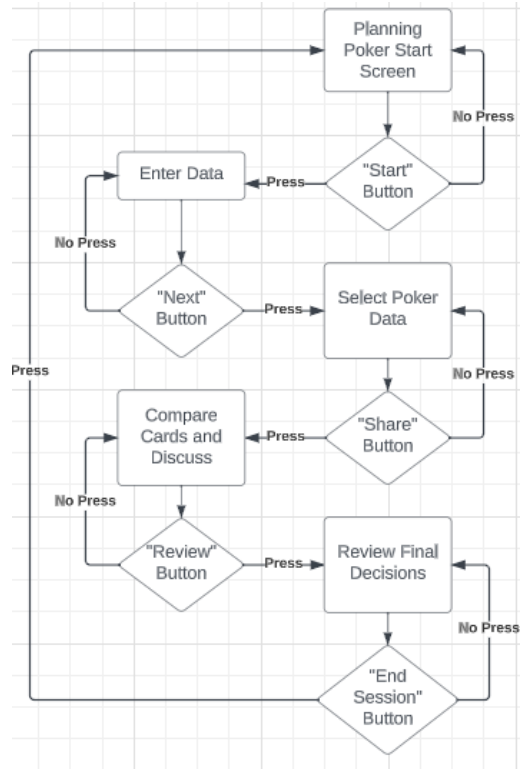


2.3. Key Function 3: Planning Poker Button Functionality

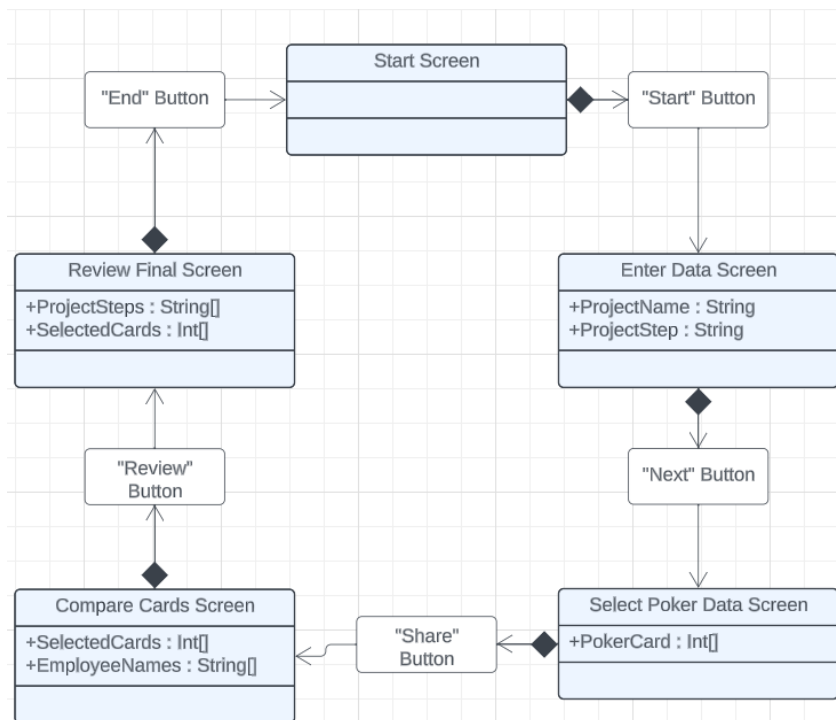
2.3.1. Author of the Function: Karryl Dimalag

2.3.2. Activity Diagram:

Team Project Phase 5
UML Design Details



2.3.3. Class Diagram:

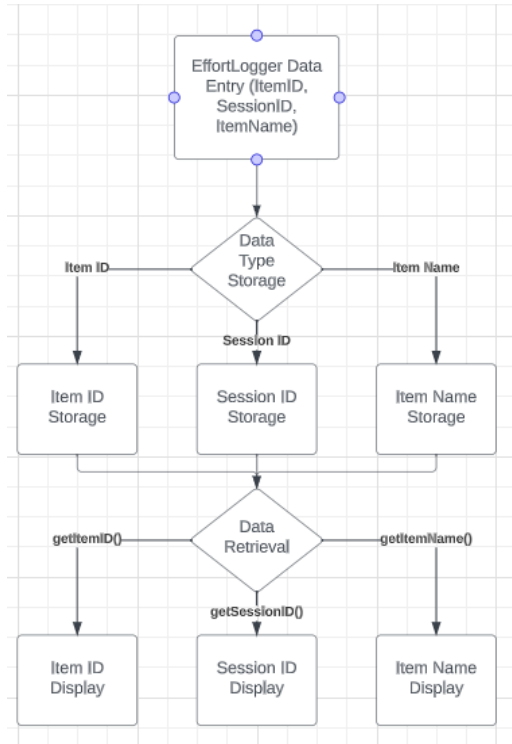


2.3.4. Sequence Diagram:

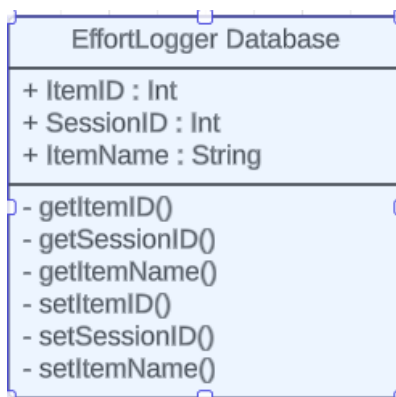
2.4. Key Function 4: SQL Database Flow

2.4.1. Author of the Function: Alma Babbitt

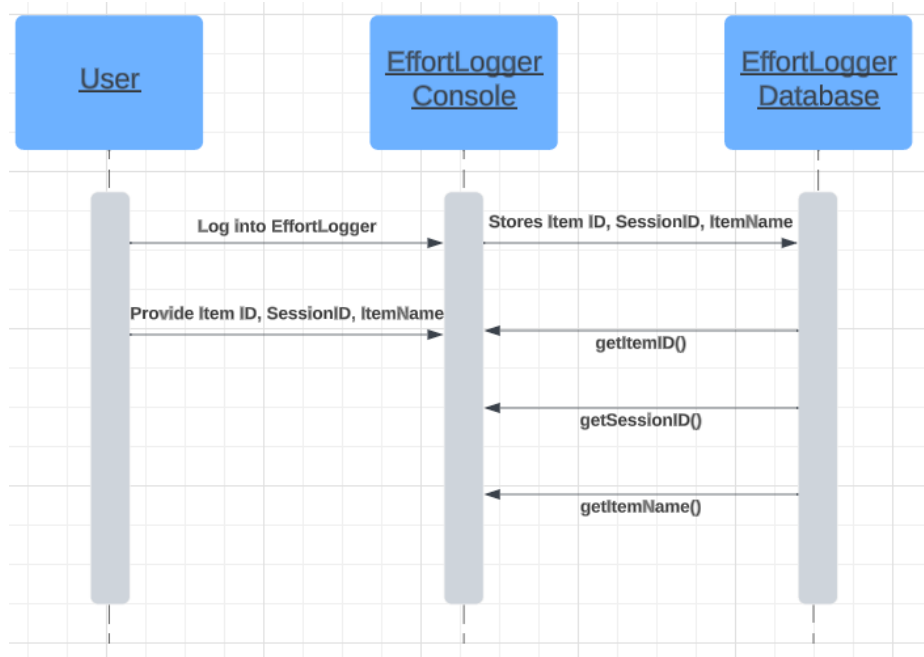
2.4.2. Activity Diagram:



2.4.3. Class Diagram:



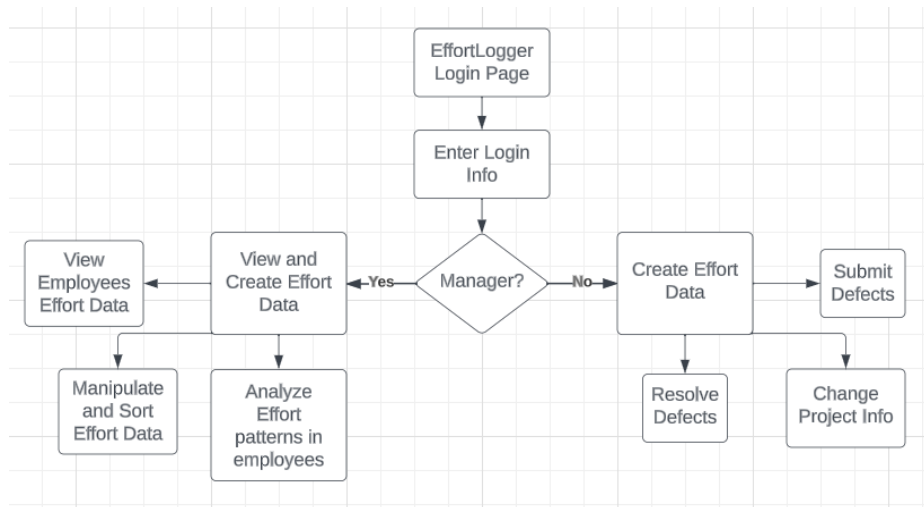
2.4.4. Sequence Diagram:



2.5. Key Function 5: Accessing Data as an Employee/Manager

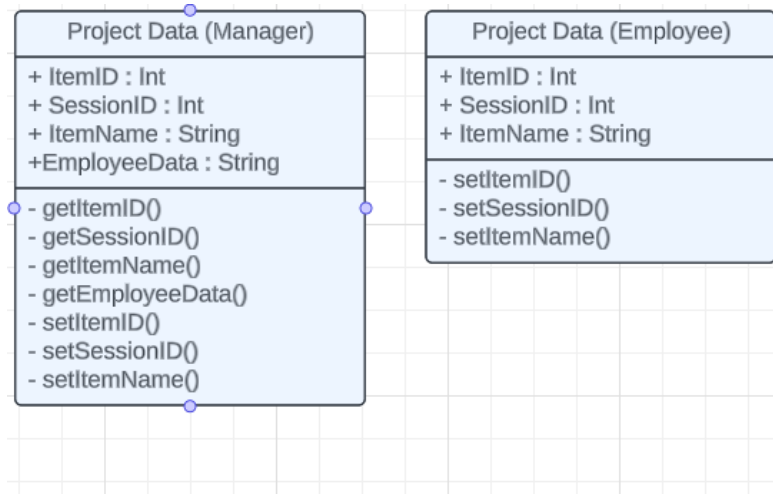
2.5.1. Author of the Function: Trevor Huss

2.5.2. Activity Diagram:

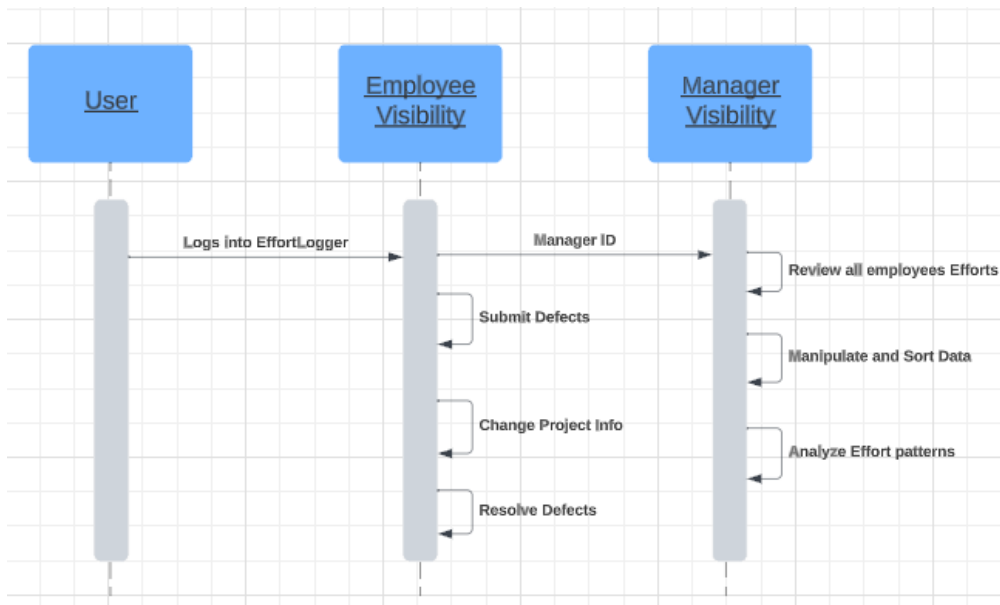


2.5.3. Class Diagram:

Team Project Phase 5
UML Design Details



2.5.4. Sequence Diagram:



3. Screencasts

3.1. Screencast for Key Function 1

3.1.1. Author of the Screencast: Zachary Litwin

3.1.2. Overview Outline:

- The activity diagram represents the order and flow of what a user using the function would experience and that experience was used to define a class diagram and sequence diagram.
- There are three static constants used in the class, they are MAX_PLAYERS, MIN_PLAYERS, and POSSIBLE_CARD_VALUES. Both max and min are just integers but the possible card values are an array of integers that contains the list of modified Fibonacci numbers that represent the card values in planning poker.
- The two functions *setNumberOfPlayers()* and *giveCardValue()* are fairly straightforward and just contain a couple of checks to make sure the given value is valid with basic outputs that show the failure or success of the function.
- However, the output of the *checkResults()* function is an integer array that has two elements called outliers. If the number of *playersPlayed* and *numberOfPlayers* is the same then the function will iterate through all the card values and put the minimum and maximum card values in the first and second elements of outliers. If the number of *playersPlayed* is less than *numberOfPlayers* then the difference between them is put as both elements of outliers.
- Then all of the functions of the code were demonstrated to be working and basic invalid input cases were shown to be handled properly as intended

3.1.3. URL to the Screencast:

https://drive.google.com/file/d/1v0ELXBGBvh257_re0W6f1X1wqNW3AyA/view?usp=sharing

3.2. Screencast for Key Function 2

3.2.1. Author of the Screencast: Ishan Yelnoorkar

3.2.2. Overview Outline:

- The class diagram represents how effort logger and planning poker are set up following each other. The sequence diagram dictates the behavior and the interaction with the effort logger as an employee and then transitioning to planning poker from the effort logger
- The sequence diagram shows the access to the effort logger
- Explain how the class diagram is implemented by the code, PlanningPokerFlow takes a Project object to interact with the project details of the concerned project.
- The explain how an effort entry is created when an activity is stopped, and how that effort entry is preloaded into the effort log editor.

- Cut back to the code to show that planning poker takes a Project object.
- Finally, simulate planning poker through the terminal. Display that the planning poker screen already has your effort entry in it. It then prompts you to discuss a particular user story.
- Planning poker is then performed
- Finally, explain how planning poker facilitates accurate estimation by discussing outliers in story estimations.

3.2.3. URL to the Screencast

<https://drive.google.com/file/d/1Xj28BRJSbfqzhCpA6Olc0bBY4e76NqTe/view?usp=sharing>

3.3. Screencast for Key Function 3

3.3.1. Author of the Screencast: Karryl Dumalag

3.3.2. Overview Outline:

- The formatting of each main.java, controller, and .fxml files are explained and the purposes for each
 - main.java: Creates each page and sets up the scene size.
 - controller:
 - Responsible for closing each scene when the “next” button is pressed to move on with the planning poker process, linked each page together along with their corresponding .fxml files.
 - Also responsible for storing the data inputted on the present screen. The “bulk” of the program.
 - .fxml:
 - We use AnchorPane as the root for designing each page.
 - Buttons, Labels, and other objects for UI/UX purposes are created and styled on FXML.
 - Calls EventHandler for buttons
- The Activity diagram, class diagram, and sequence diagrams are represented with each planning poker scene in which the next submit, and text fields are responsible for workflow and data.

3.3.3. URL to the Screencast:

https://drive.google.com/file/d/1OiSqLVGAxyDB796q7e06BH_BZXdqSLqX/view?usp=sharing

3.4. Screencast for Key Function 4

3.4.1. Author of the Screencast: Alma Babbitt

3.4.2. Overview Outline

- The Effort Logger and Planning Poker require a database in which data can be stored, accessed, and changed.

- This video shows the implementation of the database for planning poker
- Each table in the planning poker database contains necessary columns in which user input can be stored
- Item Evaluation Table: This holds the information of each item estimation in the project for each employee.
- Items Table: This holds all the items in all projects but each item is associated with a specific project and item average
- Project Table: This holds all projects and their associated names and session ID.
- All this information is stored and will be later on accessed by the EffortLogger V2 effort entries
- My code did not have classes but used classes that were defined in other prototypes. I used SQL databases to construct the flow of object interaction of different classes.

3.4.3. URL to the Screencast

https://drive.google.com/file/d/1N60bMaGu31WuhDBLQ8qFD65B_msfnGnV/view?usp=sharing

3.5. Screencast for Key Function 5

3.5.1. Author of the Screencast: Trevor Huss

3.5.2. Overview Outline

- The activity diagram shows the basic mainline of a manager or an employee accessing data from Effortlogger.
- The class diagram shows a functional visibility level of the manager and the Employee with added visibility on the manager side.
- The sequence diagram shows the levels of access to data and functionality and the main point is that the manager has employee access as well as managerial access.

3.5.3. URL to the Screencast

<https://drive.google.com/file/d/1J3xDbgiLGiX9mgEV5Fk9oFeEFPOCFQZg/view?usp=sharing>

4. Conclusion

4.1. The most important conclusions:

4.1.1. Conclusion 1

- The different methods used for the flow of poker planning is important to allow the user to easily use the effort logger tool
- This is done by creating different polling on how many players in a poker session gave a response or if the card value for an item is an appropriate value for the context of the project. Overall, the planning poker flow creates ease of use for employees and managers alike.

4.1.2. Conclusion 2

- Integrating planning poker with effort logger allows for a streamlined estimation and tracking of effort data. It has a strong focus on collaboration and the utilization of historical data for a more accurate story estimation.
- To enable this, planning poker takes a project object. Effort Entries are used to facilitate planning poker, by discussing historical effort data. Discussion amongst outliers is encouraged.

4.1.3. Conclusion 3

- A well-designed UI simplifies the navigation and interaction process to make Planning Poker sessions as efficient as possible while allowing all features to be accessible without confusion of unnecessary complexity
- A Streamlined and user-friendly interface contributes to the collaboration and discussion among team members. It reduces cognitive load and provides clear feedback to its users while making sure to prevent mistakes during the logging process.

4.1.4. Conclusion 4

- Integrating the databases with EffortLogger will allow for better data sharing and collaboration. I will plan out the different tables to be used by EffortLogger V2
- Text boxes will also need to be implemented since this will allow the employees to see what is written about different items of the poker planning sessions.

4.1.5. Conclusion 5

- Having different visibility levels creates an importance on managerial positions and their ability to see others' effort logs.
- Keeping functionality of employee versus manager is strong in upping the performance of the team in the planning poker sessions.

4.2. Upcoming Important Activities

4.2.1. Activity 1: Future Edge-Case Testing

- In the future we must run EffortLogger 2.0 through further testing to attempt to reveal any edge cases that may lead to functional or performance issues.
- The testing must consider all forms of inputs

4.2.2. Activity 2: Project Finalization

- As we near the end of the deadline for the company, we must begin to wrap up the project and prepare for a final presentation
- Every aspect of the project should be covered, functional, business, and financial.

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

Conclusion

4.3.1. Parking Lot Item 1: Team Collaboration

- The team must prepare for the finalization of the project.
- The most important aspects must be thought of for the final presentation of the project. This requires high-level group thinking.

4.3.2. Parking Lot Item 2: Tone Changes

- The team's tone needs to change from what will be done to what has been done.
- The tone needs to be definitive and final.

5. Appendix A: Credit Sheet

Team Member Name	Contributions
Team Member 1 Trevor Huss	Key Function 5 Parking Lot items Upcoming Activities
Team Member 2 Alma Babbitt	Key Function 4: SQL database
Team Member 3 Ishan Yelnoorkar	Key Function 2: Integrating Planning Poker and Effort Logger Conclusion 2
Team Member 4 Karryl Dumalag	Key Function 3 Conclusion 3
Team Member 5 Zachary Litwin	Key Function 1: Planning Poker Flow

6. Appendix A: Current Team Norms

Unsigned Norm Agreement:

<https://drive.google.com/file/d/1u-mFyAV-puvFaXHqHHKCrbE5Cu1RqBAJ/view?usp=sharing>

Goals

- The team will try to abide by the client's requirements and put forward best efforts to make that a reality

Meeting and communication norms

- Class time will be utilized to gather notes and information regarding software project management processes.
- Class time will be utilized to brainstorm how these ideas could be used in the project
- The team will meet every Friday to perform a scrum, discuss, manage backlog and plan future action
- The team will meet at Noble Library, per convenience and the meeting will last 2 hours
- Apart from weekly meetings, the team will communicate via discord/text chain, giving minor updates every time, a task is complete
- The team will communicate effectively and swiftly to avoid delays in work
- During holidays and long weekends, work will be allocated per the team member's unavailability for the holiday. Work will be redistributed and overworked individuals will be compensated with fewer workloads for the following week after the holiday

Work norms

- The team will work 5 hours every week to ensure timely delivery of deliverables ●The team will split work according to the necessary skills required for the task to be completed
- In case a member of the team is not getting work done, they will be given a warning first, and then a mail to the TAs will be sent for a repeat offense
- The team will set deadlines based on the urgency and time requirements of a task
- Every week a different member of the team will be allocated to proofread work that has been done so far
- Everyone is allowed to work in their manner as long as progress is made and it does not impede the group's ability to make progress
- All team members are expected to adhere to team norms and meet its expectations

Decision Making

Team Project Phase 5
Appendix B: Current Team Norms

- For a decision to be made, the majority of team members should agree with the agenda put forward, and simultaneously try and understand and help understand why the other's point of view may or may not work with the task at hand
- Team members will listen actively and take into consideration everyone's point of view, as well as try and resolve disagreements

We, group members of Tu37 have agreed to follow the terms listed above and plan to adhere to them until the culmination of this project. Our listed names below indicate our acceptance of the norms and are used as our digital signature.

Member 1: Alma Babbitt

Member 2: Zachary Litwin

Member 3: Trevor Huss

Member 4: Karryl Dumalag

Member 5: Ishan Yelnoorkar