CSE 563 Project Individual Report Number 5

Krishnaprasad Palamattam Aji

Table of Contents

1. Architecturally Significant Elements

1.1. Overview

- Concerns
    - Main concerns are security and functionality
    - Architectural elements that are needed to mitigate these are to be discussed
- Architectural Views
    - Authentication for security
    - Authorizations to access data from EffortLogger V2 database
    - Encryption of data stored in database
    - Ease of use and correctness for functionality

1.2. Logical/Quality Elements and why each is architecturally significant

- Security
    - Username and password authentication
    - Two factor authentication for improved security
    - Data encryption using hash functions like SHA256
    - Significance
        - Data should be encrypted so competitors do not get their hands on it
        - A person's data should only be accessible to them for planning poker
        - Even if somebody gets their hands on the device, they should not be able to steal data
- Ease of use and functionality
    - JavaFX application to validate functionality
    -

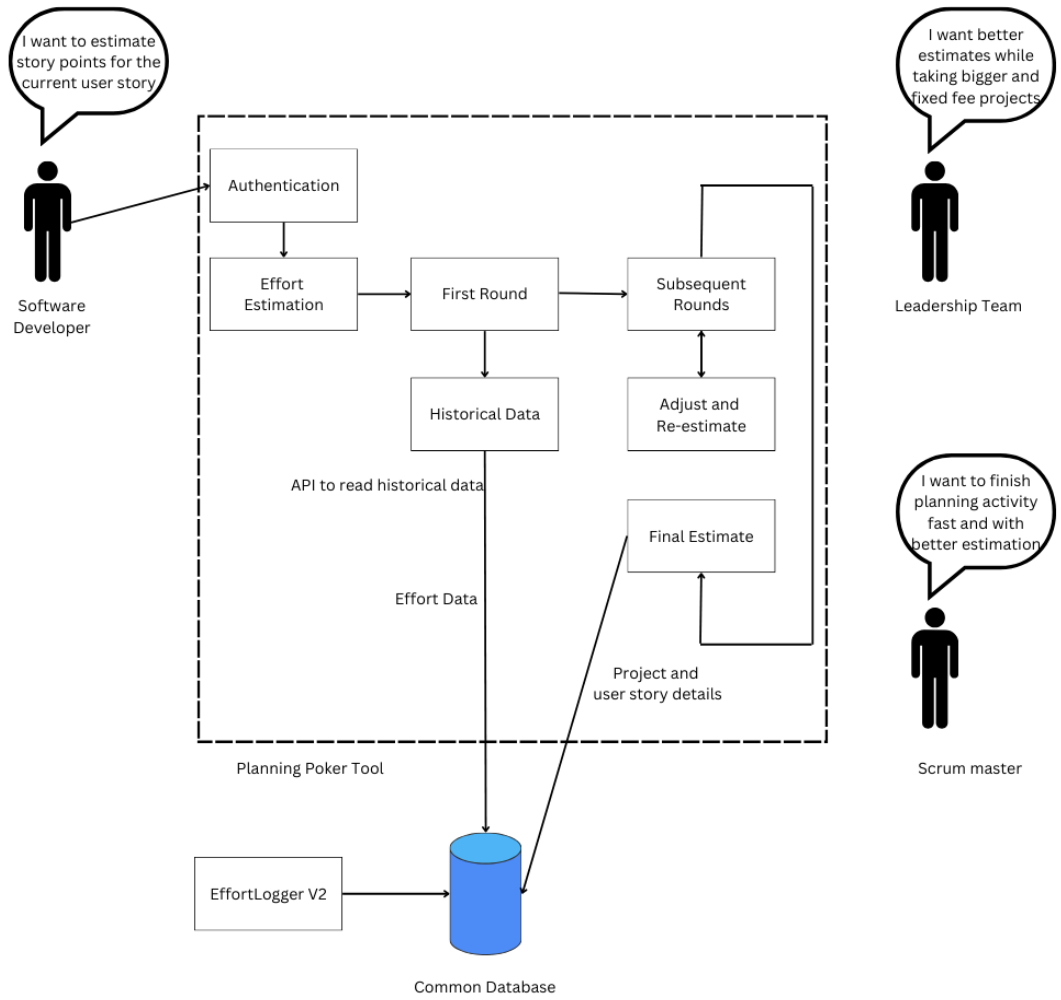
1.3. Database Elements and why each is architecturally significant

- Database common to EffortLogger V2
    - Columns in database includes
        - User story name
        - description
        - story points
        - actual effort
        - project details
        - role of implementer
    - API to fetch data from the common database
    - API to write data into database about new projects and user stories
    - Significance
        - user needs his own effort data to make better estimates
        - user story integrated effort data to help with agile project scaling

1.4. Reuse Elements and why each is architecturally significant

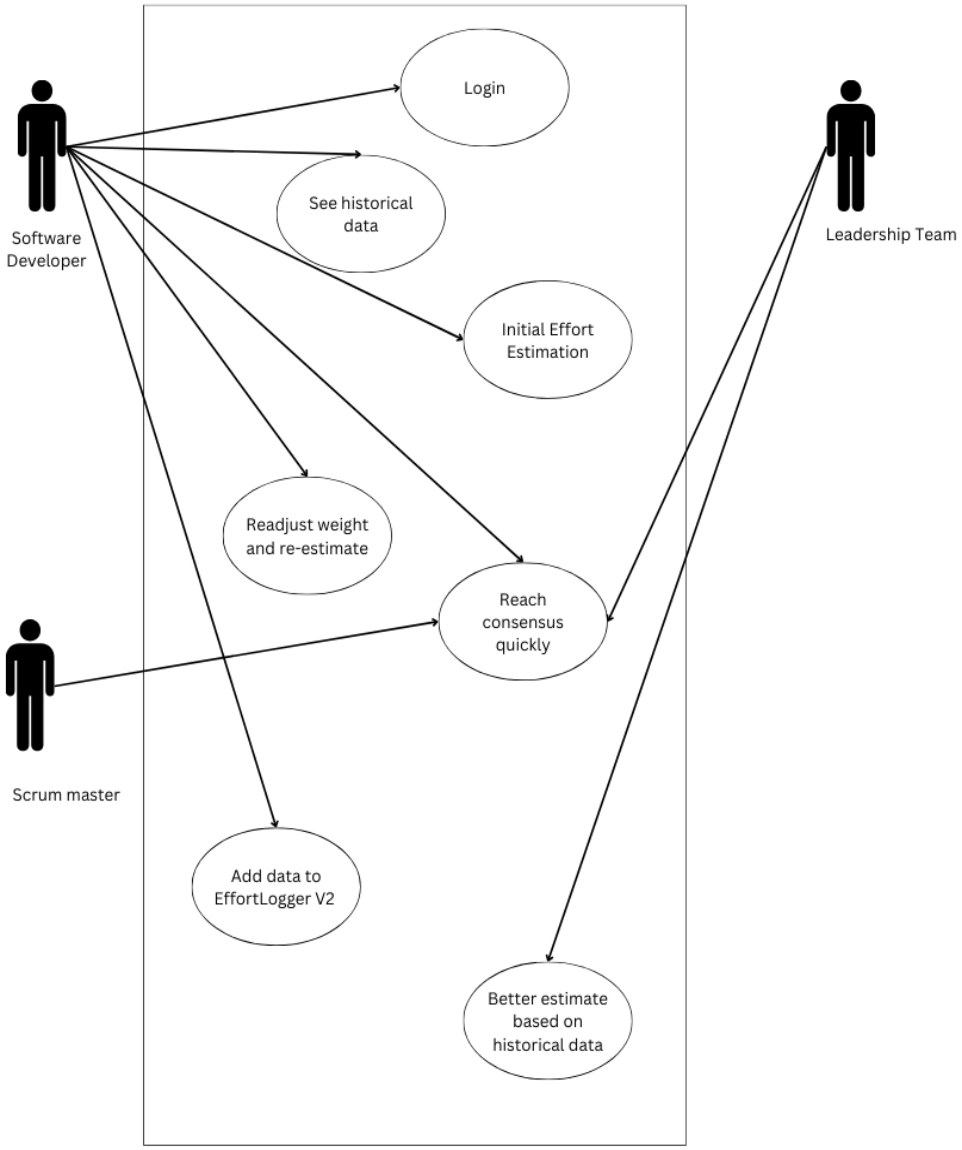
- Effort data fetching API
    - Reusable to form effort reports
    - To identify areas of improvement
    - Identify trends and other data
    - Significance
        - To take steps to improve team productivity
        - Leadership team can identify trends in effort data and act accordingly

1.5. System Context Diagram



2. Use Cases

2.1. System Use Case Diagram



Description

- The user (software developer) has to login to the planning poker tool
- The user should be able to see their historical data
- The user should be able to select relevant stories from their historical data
- The user should be able to readjust weights and re-estimate story points
- The user should be able to add the new data to the common database
- The scrum master should be able to reach consensus quickly
- The leadership team or head of the company should be able to have better data driven estimates for his company's projects

2.2. Use Case 1 : Login

2.2.1. Overview

- Discusses Login activity of user
- Significant as security is a major concern
- Authentication makes sure the data of the user is secure

2.2.2. Use Case Model

Use Case ID:	1.1		
Use Case Name:	Login after entering username and password		
Created By:	Krishnaprasad	**Last Updated By:**	Krishnaprasad
Date Created:	11/05/2023	**Date Last Updated:**	11/05/2023

Actor:	Software developer
Description:	To login to the planning poker tool, user needs to enter valid credentials
Preconditions:	1.Username is entered 2.Password is entered
Postconditions:	User is logged into the planning poker tool
Primary Pathway:	1.User enters username 2. User enters password 3. User clicks on login button 4. The user is logged into the planning poker tool
Alternate Pathways:	1.1.AP.1. Login fails 1. When username or password is incorrect, login fails 2. Enter valid credentials and login again
Exception Pathways:	1.1.EP.1. System crashes The planning poker tool will be closed
Notes and Issues:	**NA**

2.2.3. **Details**

- Login activity
- Enter username and password
- Primary pathway is to login to planning poker tool
- Alternate pathway can be login failure and new login attempt

2.3. Use Case 2 : Estimate story point based on historical data

2.3.1. **Overview**

- Discusses first round of story point estimation
- Relevant stories from historical data is selected
- Estimates based on these relevant stories is calculated

2.3.2. **Use Case Model**

Use Case ID:	1.2		
Use Case Name:	Estimate story points based on historical data in the first round		
Created By:	Krishnaprasad	**Last Updated By:**	Krishnaprasad
Date Created:	10/30/2023	**Date Last Updated:**	10/30/2023

Actor:	Software developer
Description:	The user wants to get an estimate of story points for the current user story in discussion based on historical data
Preconditions:	1. List of user stories completed by the user is shown 2. The user selects relevant stories 3. The user assigns suitable weight to the selected stories
Postconditions:	Story point estimate of the first round is shown
Primary Pathway:	1.The user is shown the list of user stories and effort data 2. The user selects relevant user stories related to the current user story 3. The user assigns suitable weights to the historical user data 4. The user clicks on Estimate button 5. The user is shown the estimate story points for the story in discussion
Alternate Pathways:	NA
Exception Pathways:	1.1.EP.1. System crashes The planning poker tool will be closed
Notes and Issues:	If the user has no relevant historical data, the user should not make use of the planning poker tool for estimates.

2.3.3. **Details**

- First round of story point estimation
- Select relevant historical data
- Adjust weights
- Estimate story points

2.4. Use Case 3 : Readjust weights and re-estimate story points

2.4.1. **Overview**

- After first round and discussions, user can re-evaluate their calculations
- The user can add or remove certain historical data
- The user can adjust weights of the historical data
- The user can now re-estimate the story points

2.4.2. **Use Case Model**

Use Case ID:	1.3		
Use Case Name:	Readjust weights and re-estimate story points		
Created By:	Krishnaprasad	**Last Updated By:**	Krishnaprasad
Date Created:	10/30/2023	**Date Last Updated:**	10/30/2023

Actor:	Software developer
Description:	In the subsequent rounds of planning poker, the user needs to readjust and re-estimate story points
Preconditions:	1. First round of estimation is completed 2. The user selects new data and removes some as per relevancy to the current user story 3. The user readjusts the weight of historical data as needed
Postconditions:	Story point estimates of subsequent rounds are shown
Primary Pathway:	1. The user reevaluates the historical data to be selected 2. The user can readjust the weights of historical data if needed 3. The user clicks on estimate button 4. The estimate of story points is shown
Alternate Pathways:	1.1.AP.1. User goes ahead with no changes The user need not necessarily change historical data and weights. They can go with their initial selections as well
Exception Pathways:	1.1.EP.1. System crashes The EffortLogger application will be closed. Clock is not started
Notes and Issues:	**NA**

2.4.3. **Details**

- Subsequent rounds until the final round follows this use case
- The user can readjust and reevaluate their historical data weights
- The user can remove or add user stories based on their relevance
- The user finally reaches an estimate where there is consensus with other players

2.5. Use Case 4 : Adding data to EffortLogger V2 database

2.5.1. **Overview**

- Discusses addition of data to database after consensus is reached
- Significant as user story needed in effort logger to keep track of effort

2.5.2. **Use Case Model**

Use Case ID:	1.4		
Use Case Name:	Add data to EffortLogger V2 database		
Created By:	Krishnaprasad	**Last Updated By:**	Krishnaprasad
Date Created:	11/05/2023	**Date Last Updated:**	11/05/2023

Actor:	Software developer
Description:	To add data to EffortLogger V2 common database so that the EffortLogger can have the user story information
Preconditions:	1.Project name is provided 2. User story name is provided 3. User story description is provided 4. User story points after consensus is provided
Postconditions:	The user story data is stored in the EffortLogger V2 common database
Primary Pathway:	1.User enters project name 2. User enters user story name 3. User enters user story points and description 4. The user clicks on Add to EffortLogger V2 button
Alternate Pathways:	1.1.AP.1.Discrepancy with entered data 1. User will have to correct data in case of duplicates 2. Enter valid data in fields and click on Add button
Exception Pathways:	1.1.EP.1. System crashes The planning poker tool will be closed
Notes and Issues:	**NA**

2.5.3. **Details**

- Final step where user story details are added to the database to be used by EffortLogger V2
- After this, the user can start estimation of next user story

3. Conclusion

3.1. Planning Poker Tool eases planning activity

3.1.1. Data driven estimation

- Takes historical data from EffortLogger
- User can select relevant data and base their estimates on these stories

3.1.2. Reach consensus faster with better estimates

- With reference to previous stories, estimation will be faster
- Users with extreme estimates can justify them by referring to the historical data

3.2. Confidential information security will be provided

3.2.1. Authentication

- Login activity to validate user
- Only qualified people get their hands on the data

3.2.2. Encryption of data in database

- Data stored will be encrypted using hash functions
- Even if device is stolen, data will not be accessible
- Offline storage also prevents online attacks

3.3. Items for future consideration

- To automatically select relevant stories based on the descriptions provided
- Communication features like chat on the Planning poker tool
- Further stakeholder feedback integration