SOFTWARE ENGINEERING CSC648/848
MOVAI
Team 02 Section 04

Team:
Ashmitha Pais – Team Lead
Steve Betts – Scrum Lead
Nathan Loo – Product Owner
Chris Farnsworth – Back-End Lead
Abdul Barrie – Front-End Lead
Preet Dhaliwal – Git-Hub Lead

Milestone 3
19/04/2023

Appendix I –
Rubrics and checklist for Part 1 Milestone 3 review:
MovAI Project
Status and UI Review.
Section: 04
Team: 02
Date: 19/04/2023
Number of students present:6

1. **UI and functionality feedback (P1 functions only)**

1. Provide more information like summary or synopsis for each movie recommended so the user can have more context

2. Provide thumbnails against each movie title for ease of user

3. Can we add movies manually to the list?

Our Plan –

1. **Planning on adding a brief summary against each title. Will make the call to openAI asking for an additional column titled summary.**

2. **Not sure if we can implement images for each movie as we have too many movies in our database and will require more space and currently, they are not present in IMDB, so we might have to do it manually and will end taking up too much space.**

3. **We could ideally add movies but it defeats the purpose as we want to see if the user likes the movies open-AI generates so we can make more useful suggestions, but we can probably add a feature to add movies if that's what is really needed to improve suggestions.**

2. **List of P1 features committed for delivery–**

1**. Login** – User enters username and password to login
2**. Logout** – User's token gets deleted and session auth token expires
3. **Sign Up** – User's auth token gets set and preliminary data is inserted by user
4. **Provide Input** – User inserts data to be sent as prompt to open AI to provide suggestions
5. **Get movie recommendations** – Based on prompt, user receives movie recommendations
6. **Rate the movies** – Based on recommendations generated, user can rate the movies
7. **See your movie list** – User can see all movies liked and disliked in their list

3. **Project status** –

We have demonstrated 7 key P1 functionalities.
We have connected front-end and back-end and have followed UX flow that we mentioned in M2. The current version of our SW prototype is running on deployment server

Risks:

1. **Extension of deadlines**
    a. Failure to meet internal deadlines due to other personal commitments or health issues
    b. Someone else from the team takes it up and works accordingly to get the feature working
    c. Move internal deadlines so it can be managed

2. **Team Work**

a. Good communication, coordination and energy levels
b. Have internal meetings on Discord and we maintain different channels for Front-End and Back-End so all information is properly maintained and can be reused by everyone later
c. No issues that can't be resolved so far by proper planning and chunking of tasks

3. Schedule

a. Sufficient time is allocated for each team member to work on a task
b. Internal deadlines are set, which are agreed upon by everyone
c. There is plenty of room for different team members to jump in and take a look
d. Schedule is quite flexible and has to be informed to other team members prior if it's too steep to meet due to unforeseen circumstances

4. Changes

a. If changes are required to the current implementation, after all the features are already built, it is mentioned during the team meeting and everybody listens and considers the input, makes valuable suggestions and finally agrees or disagrees to make changes to the current implementation and sprint plan

5. Handling Tasks

a. Microsoft whiteboard is used where tasks are assigned against each team member
b. A separate area with back-logs is also maintained
c. The new tasks are automatically assigned by the member who thinks they can take it up into their list

d. If it doesn't get assigned at all, it is discussed during the team meet and added to someone who has enough bandwidth to take it up

6. **Handling GIT merges and conflicts**
    a. As of now a branch is created for each team member and once it gets completed, branches are merged and tested out before they can be integrated to other team members branches
    b. Moving forward, we will create branches based on features so we can divide it and complete it to perfection and avoid chaos
    c. Merges into main branch have to be agreed by at least one other team member so it doesn't get added arbitrarily

7. **Confidential Data**
    a. Confidential data like keys and. env folders are not pushed into git
    b. They are stored locally and sent over discord and added manually after each fresh pull from git
    c. This makes sure no confidential data is leaked
    d. All passwords stored into our project's database is encrypted and a new session token and CSRF token is created to maintain security