**CS673 Software Engineering**

**Team 4 : Recommendation Engine**

**Meeting Minutes**

**Meeting 3**

**Date and Time:** May 26, 2022

**Place**: Discord voice chat

**Participants:**

**Minutes taker:** Ellie

**Timekeeper:**

**Purpose:** regularly scheduled meeting

**Agenda:**

1. Update
2. Cover what’s due on tuesday
3. Divide deliverable into sections and assign to team members

* Readme.md (updated)
* Doc/CS673\_presentation1\_teamX
* Doc/CS673\_SPPP\_teamX (updated)
* Doc/CS673\_SPPP\_RiskManagment\_teamX(updated)
* Doc/CS673\_SDD\_teamX (new)
* Doc/CS673\_STD\_teamX (new)
* Doc/CS673\_progressreport\_teamX(updated)
* Doc/CS673\_meetingminutes\_teamX(updated)
* Code/… : runnable source code

**Assignments** (everyone contributes individual progress update)

Update

* Alex
  + SDD- software architecture
  + Presentation video
  + SPP update
* Ellie
  + SDD- database design, software architecture
  + progress report team summary
  + SPP update
* Eric
  + STD
  + Presentation video
  + SPPP update?
* Rui
  + SDD
  + SPP update?
* Sujeet
  + Risk management?

**Discussion**:

.ipynb file with test data from disk

Discussion of the data (users have 20+ ratings, movies have mean 18 ratings

Eric’s:

Show 30 movies to user and get their input (like, dislike, superlike, dont care)

These movies have the highest number of ratings (most data)

Compare to a control model

(eliminate the 30 data-collection movies)

30 highest-rated movies that have at least 5k ratings

Rui:

Prototype of neural network

Todo: split training data into test and training

Alex:

Github configuration

Don’t push to main branch, make a pull request from your own branch

**Meeting 2**

**Date and Time:** May 19, 2022

**Place**: Discord voice chat

**Participants:** Rui, Alex, Sujeet, Eric, Ellie

**Minutes taker:** Ellie

**Timekeeper:** Eric

**Purpose:** Touch base and plan for the next iteration

**Agenda:**

1. Intro
2. Recap Alex’s work
3. Discuss plans and features

**Discussions:**

Talked about the viability of using tensorflow on the server

Angular/css

Pivotal tracker- ticket creation

Feature discussion-

User input: selects movies from a list of movies in database (like, dislike, superlike)

Output: ordered list of movies you might like

User features (written by Eric)

Draft Core Feature User Stories 1. As a user, I want to use my browser to access the RecFlix application. 2. As a user, I want to see a list of possible movies so I can let the application know my preferences. 3. As a user I want to give information to the application by selecting one of these options for each movie: a. I love this movie b. I like this movie c. I dislike this movie d. I haven’t seen this movie 4. As a user, I want to receive an ordered list of movies that I am most likely to enjoy.

**Meeting 1**

**Date and Time:** May 14, 2022

**Place**: Discord voice chat

**Participants:** Rui, Alex, Sujeet, Eric

**Minutes taker:** Eric

**Timekeeper:** Eric

**Purpose:** Decide what to do for our project and assign work for iteration 0 due Monday 3/16

**Agenda:**

1. Introduction
2. What project should we do?
3. How should we assign work for iteration 0.

**Discussions:**

Rui: from China. 4 years experience, works for company in Kentucky, mostly backend. Interested in python

Sujeet: from India, 4 years experience at Boeing in .NET, some python. Also interested in python. Up for anything since I want to learn

Alex:, works at Mass Mutual as site reliability engineer, bachelors in computer science. Familiar with Python, and some other languages.

Eric: works at NYC Department of Education. Most of my prior technical work is in SAS, SQL, and Python. Not familiar with front-end but open to learning anything!

***Project Ideas:***

Rui: Recommendation System (e.g. like Spotify or Netflix), Kaggle. Front end (Angular) would be some kind of emulation of a Netflix-like app. Python back end (using TensorFlow) to analyze the training data and select recommendations based on that data. Relational Database in background that Python could access through pyodbc.

Eric: Database of student data at a K-12 school. Front-end would print report cards, transcripts, and schedules. SQL back-end database.

Sujeet: System to manage human resources. After COVID, most companies are doing work and hiring online. Many companies are struggling with this (e.g. one person making hundreds of calls to candidates in a day). Automating work like this (e.g. automated messages) could be very valuable to companies. Python would drive the software itself. Front end would be Angular. Database SQL server.

Rui: Has chatbot project on git we might build on. Python would run the bot itself. Web interface via Angular or something. Lots of data in database to emulate conversations.

Infrastructure: maybe use free AWS tier to host

**Key Decisions**

Recommendation system is a good balance of interest from a learning perspective and doable in terms of a short timeframe. Since all members are supposed to contribute to code, Technology could work something like this:

* Rui: Python recommendation engine
* Ellie: SQL database
* Sujeet: Angular front-end
* Eric: Testing code
* Alex: Configuration and help where needed (?)

For tracking the project tasks, user stories, etc… Alex suggested using GitHub project which has features similar to Jira but is free.

Work due **Tuesday Morning 5/17:**

* Readme.md
* Doc/CS673\_presentation0\_teamX
* Doc/CS673\_SPPP\_teamX
* Doc/CS673\_SPPP\_RiskManagment\_teamX
* Doc/CS673\_progressreport\_teamX
* Doc/CS673\_meetingminutes\_teamX

**Plan for Iteration 0 Work Due Tuesday Morning 5/17**

Try to have good drafts written in by Sunday evening

* Management plan - Ellie
  + SPPP management section (except risk management)
  + Progress Report team summary and individual contribution
  + Part of video on management plan
* Risk management - Sujeet
  + SPPP Risk Management Section
  + Progress Report individual contribution
* QA plan - Eric
  + Meeting Minutes
  + SPPP QA section
  + Progress Report individual contribution
* Configuration Plan - Alex
  + SPPP configuration section
  + Progress Report individual contribution
* Project requirements and summary - Rui
  + SPPP overview, related work, and high-level requirements
  + Brief Readme.md on GitHub
  + Part of video on project summary
  + Progress Report individual contribution

5-10 minutes video:

* + - Rui - summarize project
    - Ellie - summarize management plan

**Action Items:**

* Fill out all these documents - good draft filled in by Sunday evening eastern time
* Create video
* Finish Lab 1

Since it is hard to find a time when we can all meet we will plan to coordinate via the discord channels between now and the end of Iteration 0.