Final Report: [Event Recommend-er]

[Dipak Sunar, Sean Clewis]

May 11th 2024

1 Problem Statement and Motivation

While the world is advancing rapidly in technology, fostering innovation and connectivity like never before, it's become increasingly evident that these advancements have inadvertently pushed face-to-face social interaction to the back burner. People are often more engrossed in their smartphones, tablets, and computers, engaging with virtual friends and online communities rather than connecting with those physically around them. This shift has led to a decline in genuine interpersonal relationships and a sense of disconnectedness in society. However, amidst these challenges, there's an opportunity for technology to bridge the gap and bring people together in meaningful ways. "Event Recommender" is an app designed to address this issue by leveraging technology to facilitate real-world social interactions. By recommending local events, gatherings, and activities based on users' interests and preferences, "Event Recommender" encourages face-to-face engagement and fosters the creation of genuine connections within communities. Through this innovative solution, we aim to reignite the importance of personal interactions in an increasingly digital world, fostering a sense of belonging and shared experiences among individuals.

2 Developed Solution

The solution We came up with was to create an android application that runs on a client server architecture. The server implements a wide array of features to manage data from the client, search for events, and provide recommendations based on preferences. The search function takes the user's location, radius, specified sport, and a limit of events to display. It uses this information to return a list of events matching the criteria the user has specified. The recommendation function utilizes user preferences, such as favorite team and their favorite sport, as well as the same preferences for who they have as friends, to provide a list of relevant events that the user may want to attend. This is done automatically without user input. The client will have our application running on the background. It will collect data from the user to see what times they normally open the app. Once it has the appropriate data, it should send the request to

the server to get the recommended event. The Server also implements functions such as adding and removing data from the database. Data such as bookmarks, liked events, friends lists, account creation, account login and validation, and the initializing of the database itself. The client's user interface is designed to reflect the commands available on the server. It also displays gathered user input in preparation to communicate with the server.

3 Completed Features

- Search For Events
- Add/Remove Bookmark
- Add/Remove Friend
- Add/Remove Like
- Set Favorite Team and Sport
- Login/Create Account
- Display User Information
- Initialize Database
- Client User Interface
- Gathering User Input

4 Incomplete Features

The recommendation for events to users needs to be expanded. It currently uses two points of data for users and their friends to make suggestions. Ideally, once the database and features of the server are expanded, we would like to introduce more points to give better recommendations. For instance, we would like to take note of the venue an event takes place in order to recommend other events that happen at the same location. Additionally, we will need to access another database to acquire more information about the sports teams. This would allow us to be more granular in our recommendations. We could then recommend events based on particular players, popular teams or events, and even provide recommendations regarding team statistics and win rates.

Encryption is currently not implemented in our design. We need to make sure the data being sent in our client server architecture is encrypted. Along with the transmission of data, our database needs to store client data in an encrypted format. The passwords need to be hashed and salted to bolster our security.

Social features to interact with people on your friends list is also not implemented. The design is to have a feed tab for each user. Inside the feed tab you will see events that your friends have liked, bookmarked, and previously attended. In addition, major sports news will be displayed based on the preference of that user.

The client and server were developed separately and have not been tested to communicate with each other. They currently work with a simulated connection sending JSON data back and forward. In the future we would like to make sure the client and server are properly communicating using real sets of data and test them on physical android devices.

5 Challenges during development

During the development of this project, we encountered numerous challenges that required careful consideration and problem-solving. One of the initial hurdles we faced was deciding which programming language to use for the project. After examining the options and considering factors such as compatibility, time to learn the language, and be able to implement in the project, we made a decision on the most suitable language to proceed with. Once the language was chosen, we stated learning and implementing, watching countless hours of YouTube tutorials to learn the necessary knowledge and skills. This process involved understanding the concepts and techniques required to bring our ideas and desired features to life in the mobile app.

Despite our determination and effort, we encountered another obstacle in the form of time constraints. The time-frame for the project was limited, and this added pressure to our development process. Setting up the development environment proved to be a challenge in itself, hours of troubleshooting to ensure everything was configured correctly. Additionally, we faced technical difficulties and compatibility issues along the way, which further prolonged the setup process.

Despite these challenges, we pushed through and committed to overcoming each obstacle in our path. We eventually managed to overcome the challenges we encountered and successfully set up the development environment for the project. This experience taught us valuable lessons in problem-solving, collaboration, and perseverance, which we carried forward into the subsequent stages of the project's development.

6 Team Collaboration and Work Distribution

We split the work evenly among team members, assigning specific tasks to each. We divided the project into Client and Server side development sections to streamline our efforts. Despite juggling multiple classes and projects, we communicated effectively and collaborated to get the job done. We used collaborative tools to coordinate our efforts and share resources. Once we designated who was responsible for the client and who was responsible for the server, We followed our project specification to implement features sequentially. Through our hard work and dedication, we tackled the challenges and delivered a final project.

7 Conclusion

This project has been an invaluable learning experience, providing us with the opportunity to develop and enhance our skills in software and app development. Through this project, we have gained a deeper understanding of the processes involved in building apps and software, as well as the challenges and opportunities that come with it. It has reinforced the importance of perseverance, collaboration, and problem-solving in achieving our goals. While this project marks a significant milestone in our journey, it is just the beginning of what we aspire to achieve in the future. We are excited to continue building upon the knowledge and experiences gained from this project, and to explore new possibilities in creating innovative solutions for the benefit of others.