Design Document for CyTime
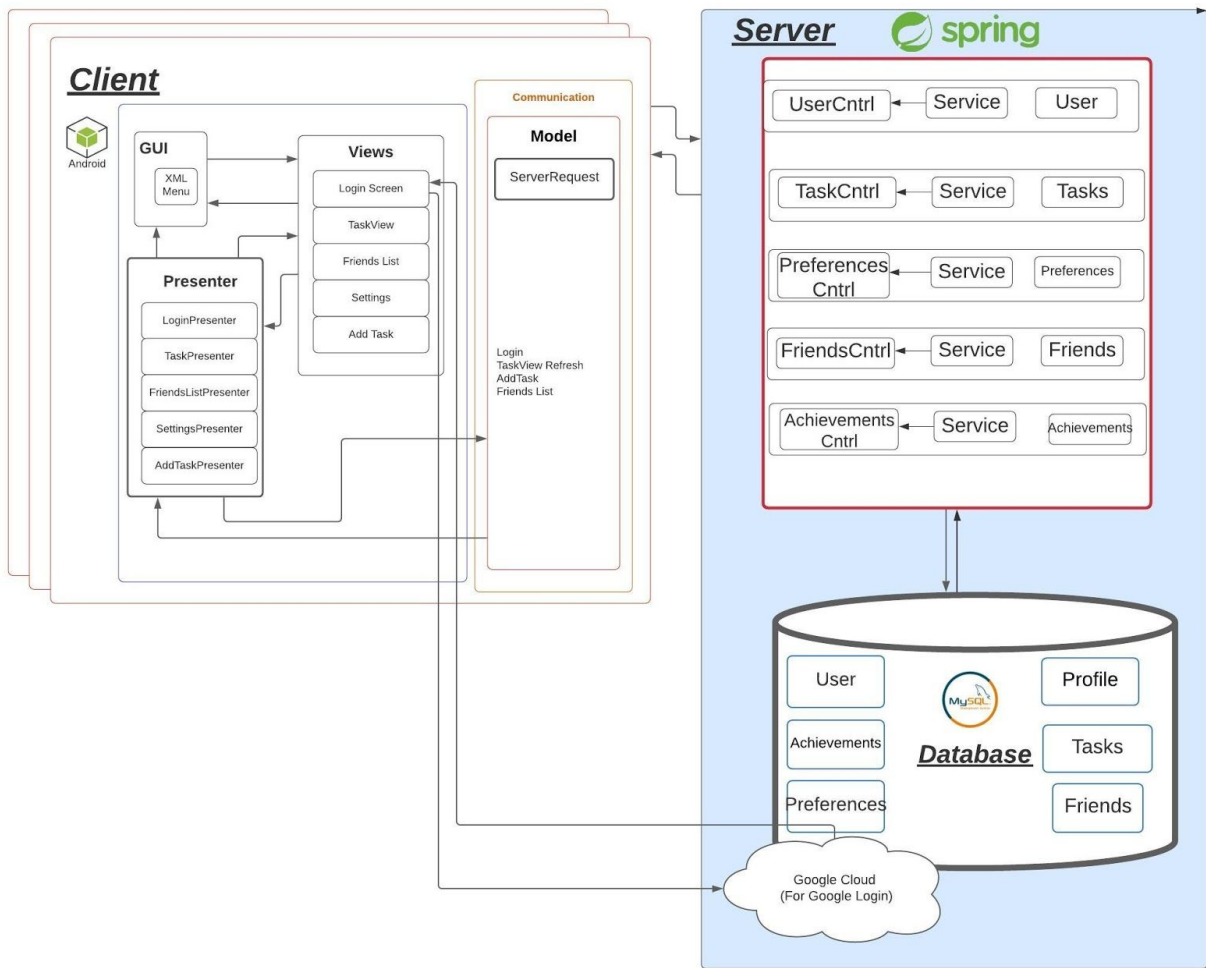
Group MC-10

Yden Da: 25% contribution

Bofu Zhu: 25% contribution

Joshua Naber: 25% contribution

Adnan Salihovic: 25% contribution

Design Descriptions

Client Side:

 The view classes handle all the UI display, the presenter classes handle the logic that view needs for the UI, and the model classes handle communication and other tools that all classes share. The logic and view displays all the tasks that the user has, the due date on each task, and the completion status for those tasks. The logic classes also handle user interactions. The login screen utilizes the google-login API, so students and faculty can login with their ISU issued credentials. Our app contacts the google cloud server to verify these credentials.

Server Side:

Using Java Spring Framework, we have controllers for each individual table in our database. The controllers have the methods necessary to modify or add items to the respective tables. The front end takes user input which gets converted into JSON files to send back to the backend to process and add to the tables. The server side also handles requests from the front end to retrieve data from the tables when needed.

Database:

Our database tables hold all the information needed for this application to operate. Tables include user information for their profiles, tasks to store the user's input task to complete, with due dates and completion, a table for the user's achievements, one for their application preferences and settings, and a table for the user's list of friends.

userLogin
- 🔑 id INT
- 🔷 email VARCHAR(255)
- 🔷 user1 VARCHAR(255)
- achievements_id INT
- friendslist_id INT
- usertasks_id INT
- preferences_id INT

Indexes ▶

preferences
- 🔑 id INT
- 🔷 user1 VARCHAR(255)
- 🔷 darkMode TINYINT(1)
- 🔷 notifications TINYINT(1)

Indexes ▶

usertasks
- 🔑 id INT
- 🔷 user1 VARCHAR(255)
- 🔷 task VARCHAR(255)
- 🔷 assignedDate DATE

Indexes ▶

achievements
- 🔑 id INT
- 🔷 user1 VARCHAR(255)
- 🔷 numOfEarned INT

Indexes ▶

friendslist
- 🔑 id INT
- 🔷 user1 VARCHAR(255)
- 🔷 user2 VARCHAR(255)

Indexes ▶