

---

## Design Document for **Casino**

---

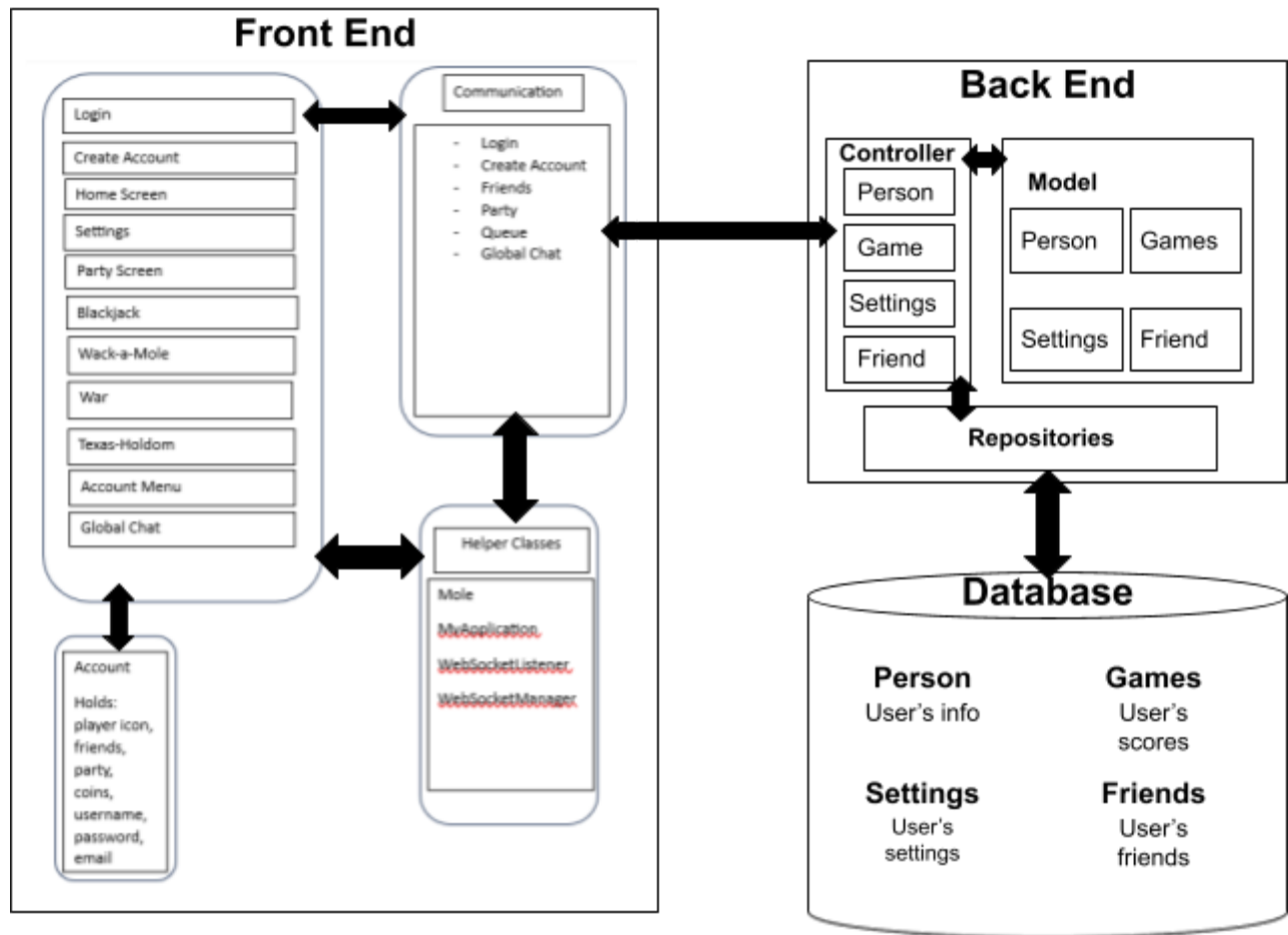
Group <MS-313>

Member1 Name: Owen 25% contribution

Member2 Name: Kevin 25% contribution

Member3 Name: Brock 25% contribution

Member4 Name: Kyle 25% contribution



Communication: These are the screens that communication with the server is required. These screens communicate with the backend via the Volley library or via a websocket connection.

Account: Holds the information about the account including username, email, password, profile icon, coins, friends, and party. These things are put and pulled from the server.

The screen names are the main screens that are used for this project. All but Login and Create Account are connected to the Home Screen.

Helper: These classes are used frequently, we made these into classes to reduce code duplication.

Server:

We utilize Spring Boot with Apache Tomcat to act as our server. Our server is split up into three different parts. The first one is the Controller, which facilitates communication to the front end. The controller is a cumulation of constructors, getters, and setters which enables the front end to interact with all of the models on the Backend. Models are Java objects which correspond to a table in the database. Finally, we also have the Repositories for each object which serve as a communicator between the Database and the Server. For example, a usual call from the front end would first go to the correct controller that made the call. Then the controller would make a call to the correct model to create the desired variables and fill them with any data given by the call. Then this model would be sent to and saved into the database.

Database:

Our database is a combination of tables that store all needed information. This information includes each user's data (username, password, isOnline, account privileges, etc), settings, game data(each high score), and friends. Our database is the main source of data used in our application. We utilize MySQL for our database software. MySQL is a relational database that we use as it supports client-server architecture, unlike strictly DBMS options. The tables we use allow us to have relationships between our data points.

