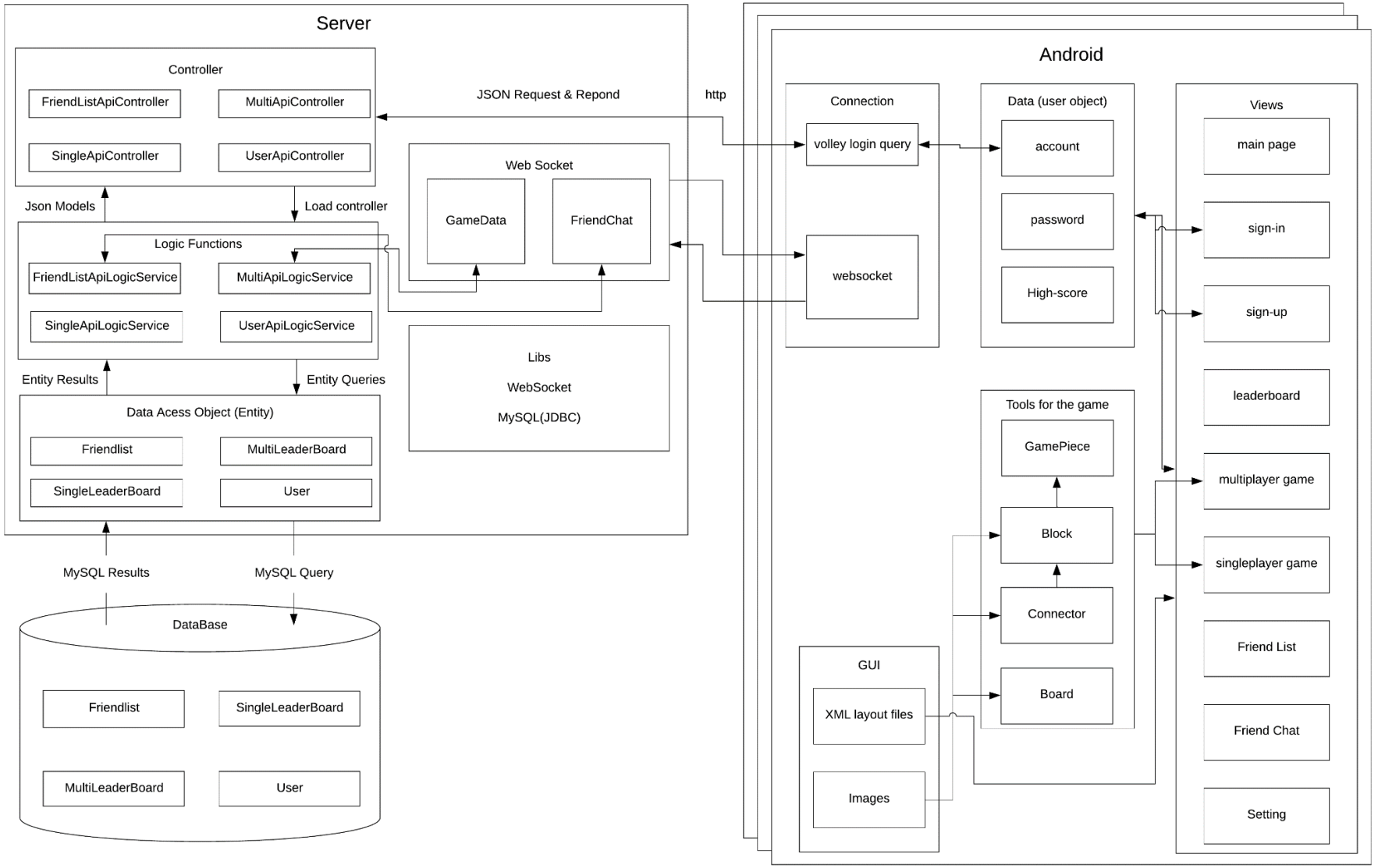
Block Diagram

NV_5 Dongwoo Kang, Ji Hoo Kim, Joseph Hudson, Zian Li

Project Name : Color Tetris.


Server
Controller
FriendListApiController
MultiApiController
SingleApiController
UserApiController
JSON Request & Repond
http
Web Socket
GameData
FriendChat
Json Models
Load controller
Logic Functions
FriendListApiLogicService
MultiApiLogicService
SingleApiLogicService
UserApiLogicService
Libs
WebSocket
MySQL(JDBC)
Entity Results
Entity Queries
Data Acess Object (Entity)
Friendlist
MultiLeaderBoard
SingleLeaderBoard
User
MySQL Results
MySQL Query
DataBase
Friendlist
SingleLeaderBoard
MultiLeaderBoard
User
Android
Connection
volley login query
websocket
Data (user object)
account
password
High-score
Views
main page
sign-in
sign-up
leaderboard
multiplayer game
singleplayer game
Friend List
Friend Chat
Setting
Tools for the game
GamePiece
Block
Connector
Board
GUI
XML layout files
Images

Android User GUI

The application has a thread dedicated to the GUI which manages the activities. Each activity has its own XML file(s) and layouts to manage and create the desired displays, the game pages having dynamic views are assisted by the code helpers, block, board, gamepiece, and connector. Most activities are self-contained in their own functionality with the active user being the only information needing to be carried between each activity.

Android Code Helpers

As stated before, there are four code helpers that help make the game playable and viewable which are block, board, gamepiece, and connector. Block and connectors help manage the visuals of the games by holding and editing the properties of images on the board. The board is the 2d array that holds all the blocks, so the data is stored in a manner that allows the game code to operate smoothly. The gamepiece manages active blocks on the gameboard to do things such as move and rotate blocks.

Android Models

We have one model in our project that maintains how we contain, transfer, and store data within the application. It can be treated as an object that resides outside of the activities as it necessary for most of them.

Android Communication

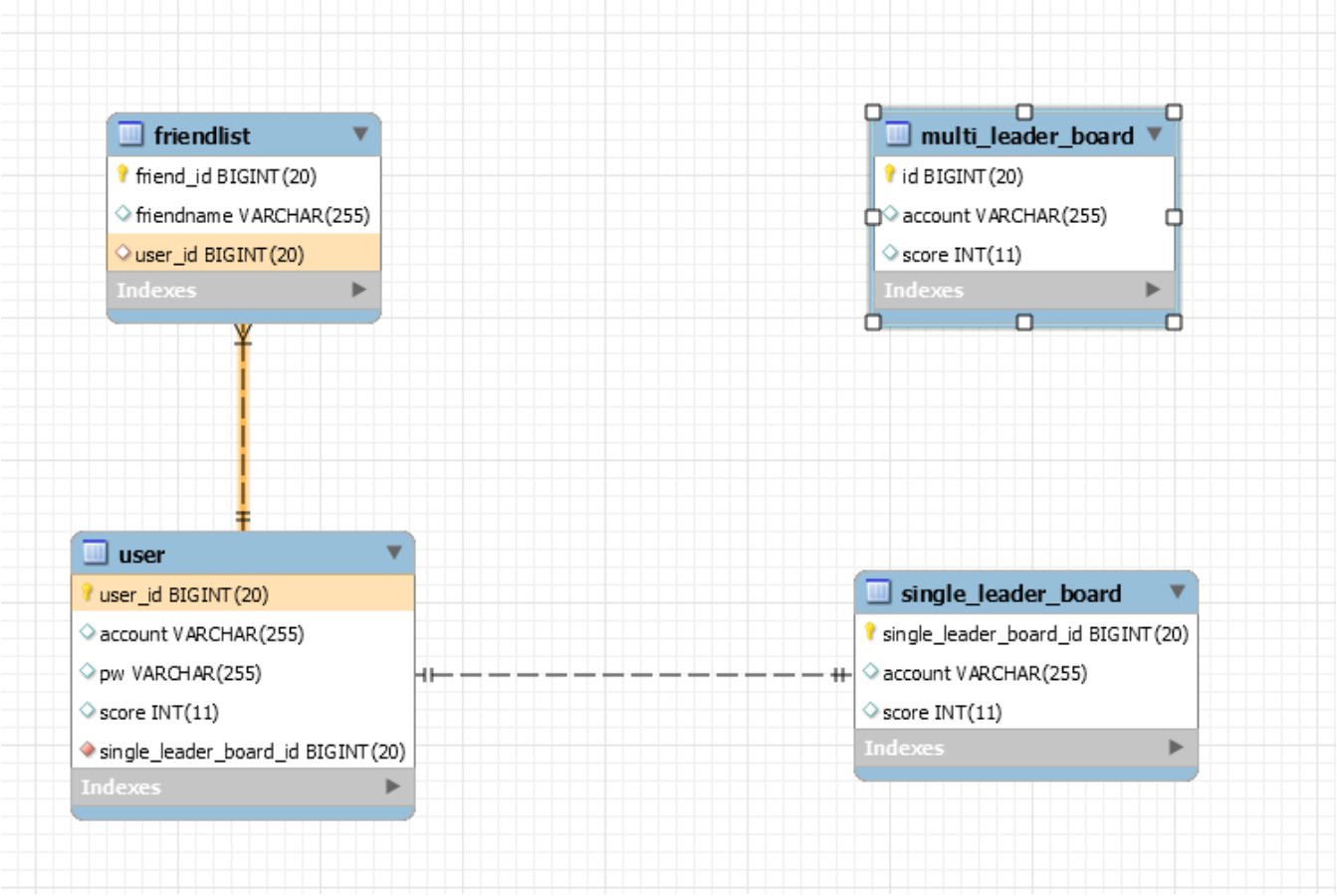
The volley communication sends and retrieve a Json object about user that contains username password and highscore of said user, a Boolean value might sometime be returned as in the login and sign in phase to deter main if the login or sign in is successful.

Sever part

We make controller, service, the entity for each data such as User, Leaderboard for multi and single, and fried list in order to get a receive and send data as JSON form from the front end. In the entity part, they are connected with the database on MySQL. The service part, as a called logic function, is to access some specific data from MySQL. Last is the controller part indicates endpoints in order to use the data table. We use Web Socket for game data in multi play and chatting with friends in game.

Data Base

We use MySQL as Data base. There are four data base; User, single leaderboard, multi leaderboard, and Friend list. User and single leaderboard have relationship as 1 : 1. User and Friend list are 1 : N relationship.



user : The table is about the information of user. The individual score save at user table with each account.

friendlist : The table is about the friend list of each user. The relationship is 1 : N. The table of friendlist is N and the user is 1.

single_leader_board : The table is about single play score. The relationship is 1 : 1.

multi_leader_board : The table is about score leaderboard for multi play.