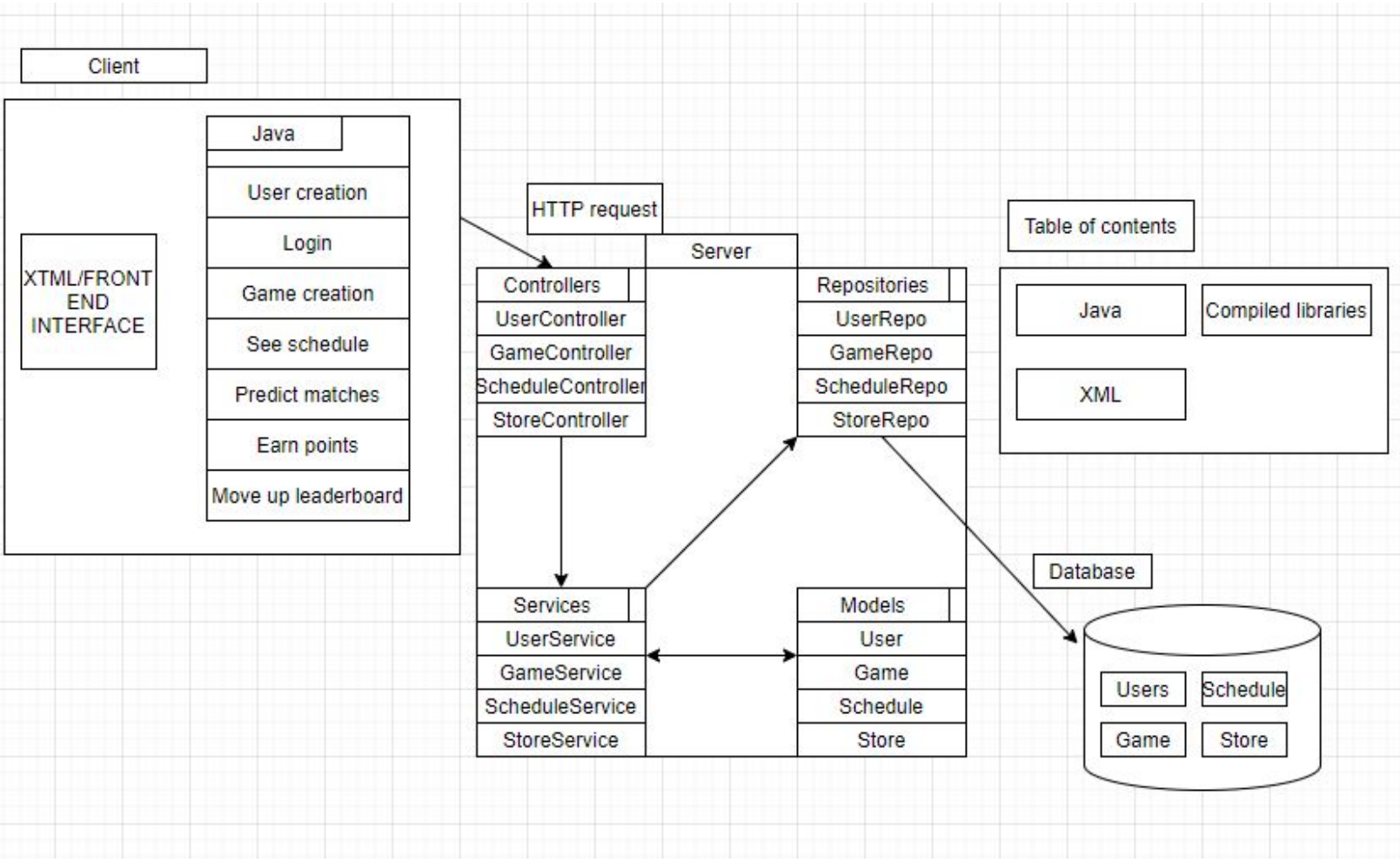
Block Diagram

Team Number:
KK_9

Team members:
Jacob Kinser: 25%
Brady Synstelien: 25%
Abdullah Anis: 25%
Richard: 25%

Project Title:
Sports League


Client
Java
User creation
Login
Game creation
See schedule
Predict matches
Earn points
Move up leaderboard
XTML/FRONT END INTERFACE
HTTP request
Server
Controllers
UserController
GameController
ScheduleController
StoreController
Repositories
UserRepo
GameRepo
ScheduleRepo
StoreRepo
Services
UserService
GameService
ScheduleService
StoreService
Models
User
Game
Schedule
Store
Table of contents
Java
Compiled libraries
XML
Database
Users
Schedule
Game
Store

Client side:

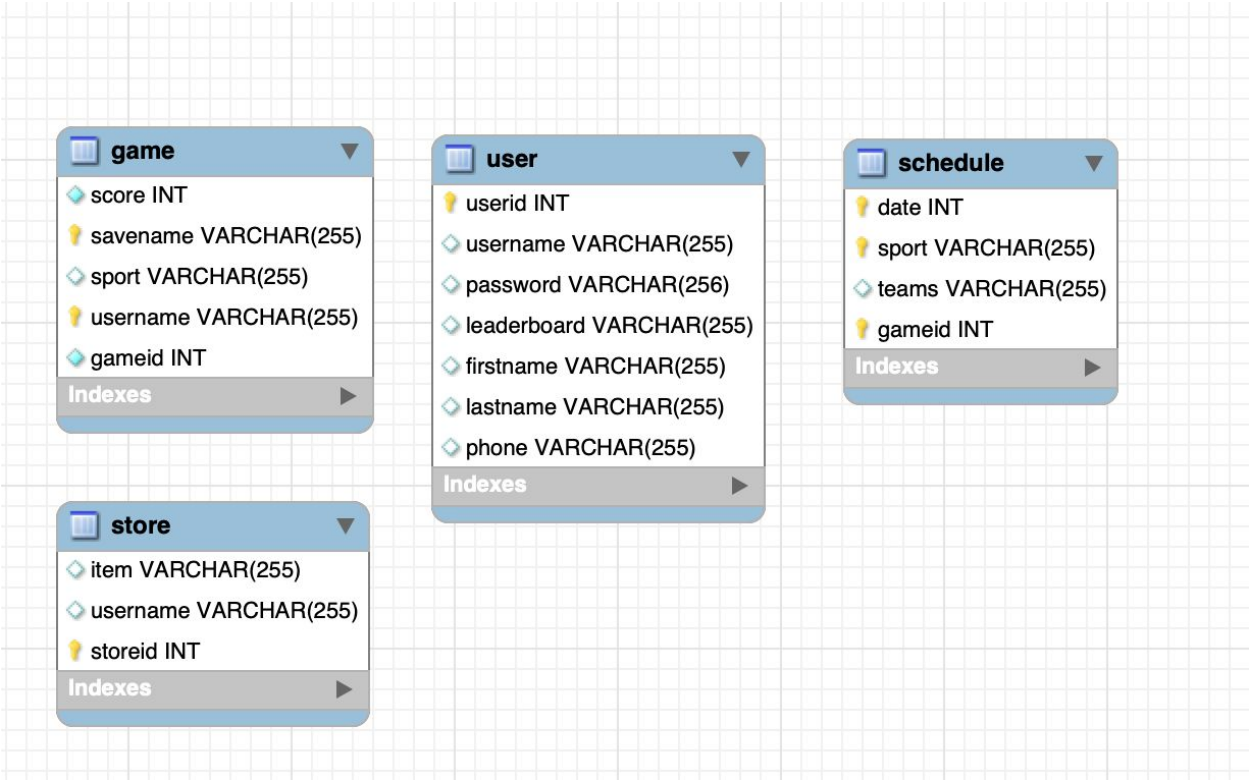
View: Our application's front end consists solely of Android clients;thus we utilize a combination of java and XML classes to display a sports betting app. With features such as a fully usable login screen and account page. Our login page is usable due to the user first inputting a username and password into our loginpage.java. Next the data gets transferred to our sql database to see if the same exact username and password is in the sql database. If true then the user enters into the app. If false the user is incentivized to enter the username or password again. Or to register a new page where the user enters into the register.java and the data in the register.java is then transferred over to the sql database and is then saved.When the user logins successfully they enter the account.xml page. This account.xml retrieves user information from the sql database and then showcases in the account.xml. The last feature is the calendar.xml which also retrieves data from the sql database.

Server side:
Our tables share attributes, for example in the store username is used and that is from the username in users. This relationship is not shown on mySQL as the user is authenticated through different methods on our springboot application and is added that way. The diagram below represents our DB, but the relationships between the tables and their entities are verified on the backend side.

For our server side flow, Client Side makes a request to server, the controller forwards the request to the corresponding service class, Which if needed, adds a new model, calls the corresponding repo class, which in turn, makes the appropriate changes to the Database. This flow is used for creating, adding, and editing entities stored within the database.

The database holds all of the crucial information for our app. This data is stored and retrieved from our springboot application.

MYSQL setup


game
score INT
savename VARCHAR(255)
sport VARCHAR(255)
username VARCHAR(255)
gameid INT
Indexes
user
userid INT
username VARCHAR(255)
password VARCHAR(256)
leaderboard VARCHAR(255)
firstname VARCHAR(255)
lastname VARCHAR(255)
phone VARCHAR(255)
Indexes
schedule
date INT
sport VARCHAR(255)
teams VARCHAR(255)
gameid INT
Indexes
store
item VARCHAR(255)
username VARCHAR(255)
storeid INT
Indexes