Design Document for Stat Sheet Stuffer

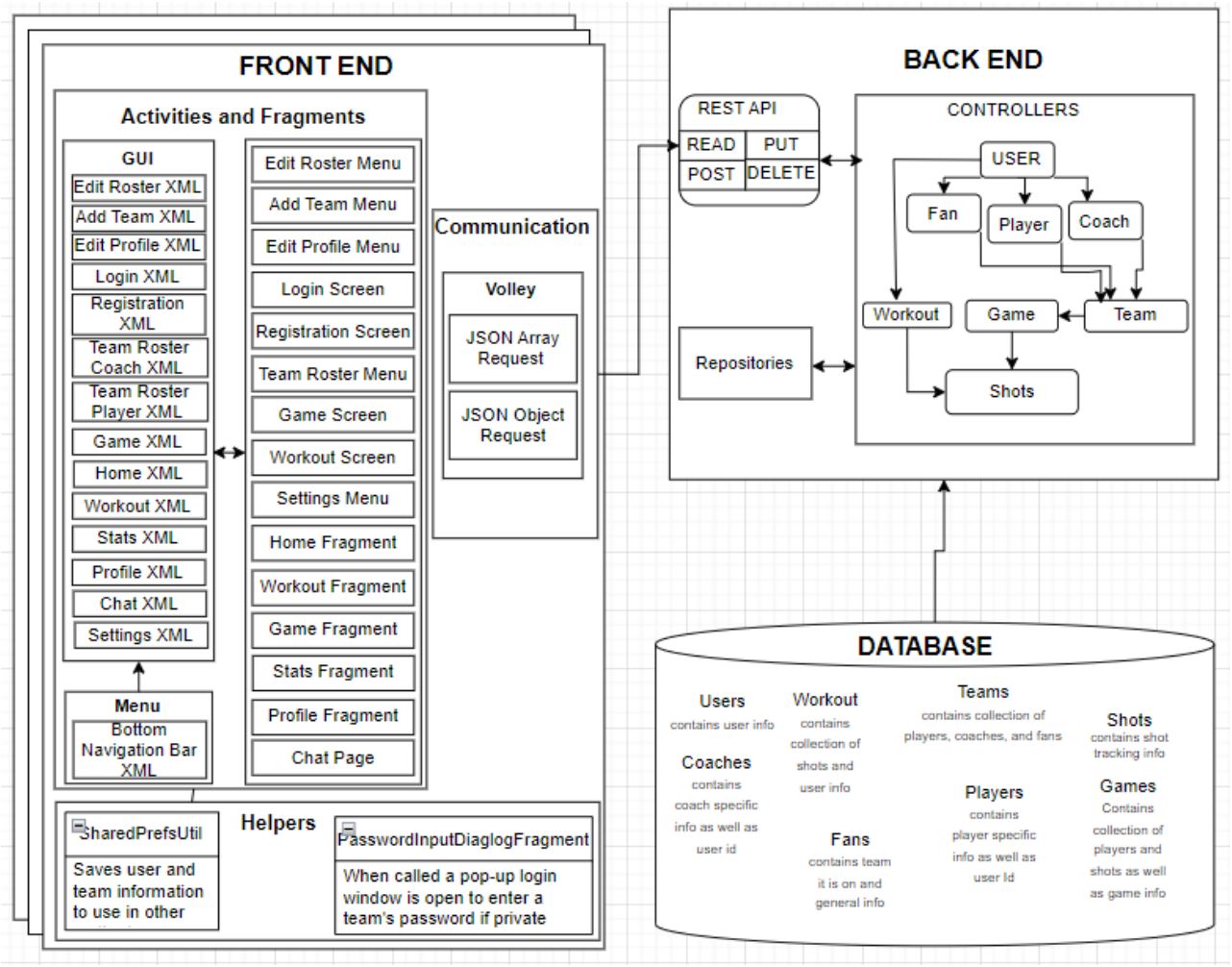
Group JK_224

Adam Hisel: 25% contribution

Mason Harsh: 25% contribution

Jay Patel: 25% contribution

Jagger Gourley: 25% contribution


FRONT END
Activities and Fragments
GUI
Edit Roster XML
Add Team XML
Edit Profile XML
Login XML
Registration XML
Team Roster Coach XML
Team Roster Player XML
Game XML
Home XML
Workout XML
Stats XML
Profile XML
Chat XML
Settings XML
Edit Roster Menu
Add Team Menu
Edit Profile Menu
Login Screen
Registration Screen
Team Roster Menu
Game Screen
Workout Screen
Settings Menu
Home Fragment
Workout Fragment
Game Fragment
Stats Fragment
Profile Fragment
Chat Page
Menu
Bottom Navigation Bar XML
Communication
Volley
JSON Array Request
JSON Object Request
Helpers
SharedPrefsUtil
Saves user and team information to use in other
PasswordInputDiaglogFragment
When called a pop-up login window is open to enter a team's password if private
BACK END
REST API
READ
PUT
POST
DELETE
CONTROLLERS
USER
Fan
Player
Coach
Workout
Game
Team
Shots
Repositories
DATABASE
Users
contains user info
Coaches
contains coach specific info as well as user id
Workout
contains collection of shots and user info
Fans
contains team it is on and general info
Teams
contains collection of players, coaches, and fans
Players
contains player specific info as well as user Id
Shots
contains shot tracking info
Games
Contains collection of players and shots as well as game info

Front End (not fully implemented yet)

Home Page

XML:

- Buttons with all the teams a user follows
- Create a Team Button
- Join a Team Button

Description: Clicking one of the team buttons will send the user to a screen with the teams roster. By clicking create a team button the user is able to type in a team name to send a POST request to the server. By clicking join a team button the user will be able to search a team to join by sending a GET request from the server and a PUT request to put the player in the teams players list.

Team Roster

XML:

- TableLayout with player information
- Chat Button
- Back Button
- Edit Roster Button (if coach)

Description: After clicking a team button in the home page the user is sent here where they are able to view the team roster. Byclicking the chat button the user can open the team's chat. By clicking the back button the user can return back to home. If a user is a coach of the team they are viewing they have a edit roster button to some player information and edit the teams settings

Backend

Communication

For our backend we mostly use mappings for communication with the database, these mappings use different url's based on the desired operation. These mappings include:

- Get: This allows us to retrieve information from the database, usually using some value like the id of an object.
- Post: This allows us to add a new object to the table, this generally creates the object and sends it to the database.
- Put: Takes existing objects and modifies the table based on the specified request,in our case either applying relations to objects or updating object fields.
- Delete: Removes an object from the database based on an id. Can remove lists of objects as well.

Classes

The classes of objects is where all the mappings are used to add, update and modify the database. These are also how the frontend communicates with the database:

Users: This has a many to many relation with teams, and a one to many with workouts.

Teams: has the many to many mentioned above, and a one to many with players, fans and coaches.

Players: Just has the many to one relation with a team.

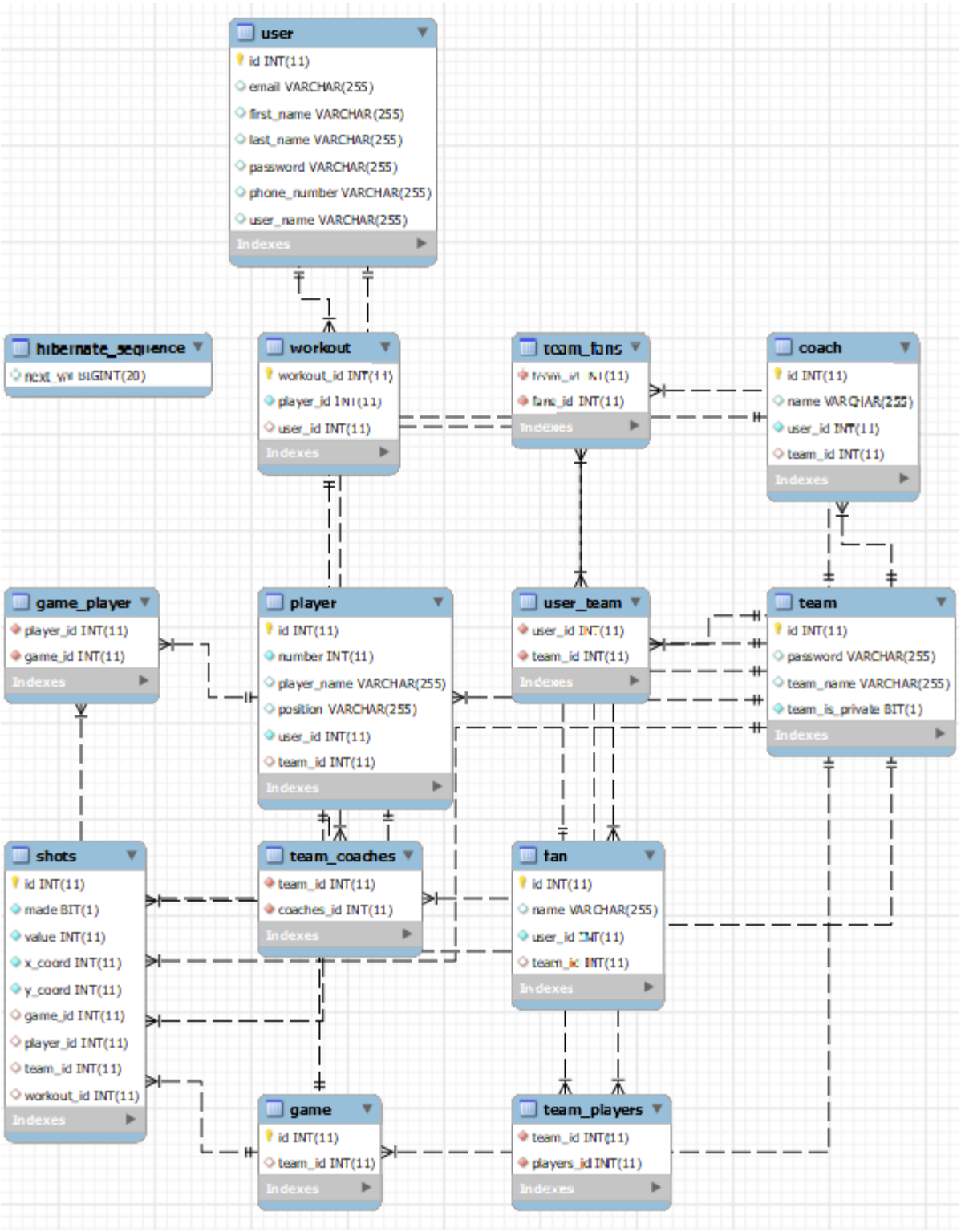
Coaches: Also only has the many to one with a team.

Fans: has a many to one with a team.

Games: This has a one to many with a team, and a one to many with shots.

Shots: just has a many to one with games.

Workouts: has a many to one with a user, and a one to many with shots.


user
id INT(11)
email VARCHAR(255)
first_name VARCHAR(255)
last_name VARCHAR(255)
password VARCHAR(255)
phone_number VARCHAR(255)
user_name VARCHAR(255)
Indexes
hibernate_sequence
next_val BIGINT(20)
workout
workout_id INT(11)
player_id INT(11)
user_id INT(11)
Indexes
team_fans
fans_id INT(11)
Indexes
coach
id INT(11)
user_id INT(11)
team_id INT(11)
Indexes
game_player
player_id INT(11)
game_id INT(11)
Indexes
player
id INT(11)
number INT(11)
player_name VARCHAR(255)
position VARCHAR(255)
user_id INT(11)
team_id INT(11)
Indexes
user_team
user_id INT(11)
team_id INT(11)
Indexes
team
id INT(11)
password VARCHAR(255)
team_name VARCHAR(255)
team_is_private BIT(1)
Indexes
shots
id INT(11)
made BIT(1)
value INT(11)
x_coord INT(11)
y_coord INT(11)
game_id INT(11)
player_id INT(11)
team_id INT(11)
workout_id INT(11)
Indexes
team_coaches
team_id INT(11)
coaches_id INT(11)
Indexes
fan
id INT(11)
name VARCHAR(255)
user_id INT(11)
Indexes
game
id INT(11)
team_id INT(11)
Indexes
team_players
team_id INT(11)
players_id INT(11)
Indexes