Design Document for Tech Conquest

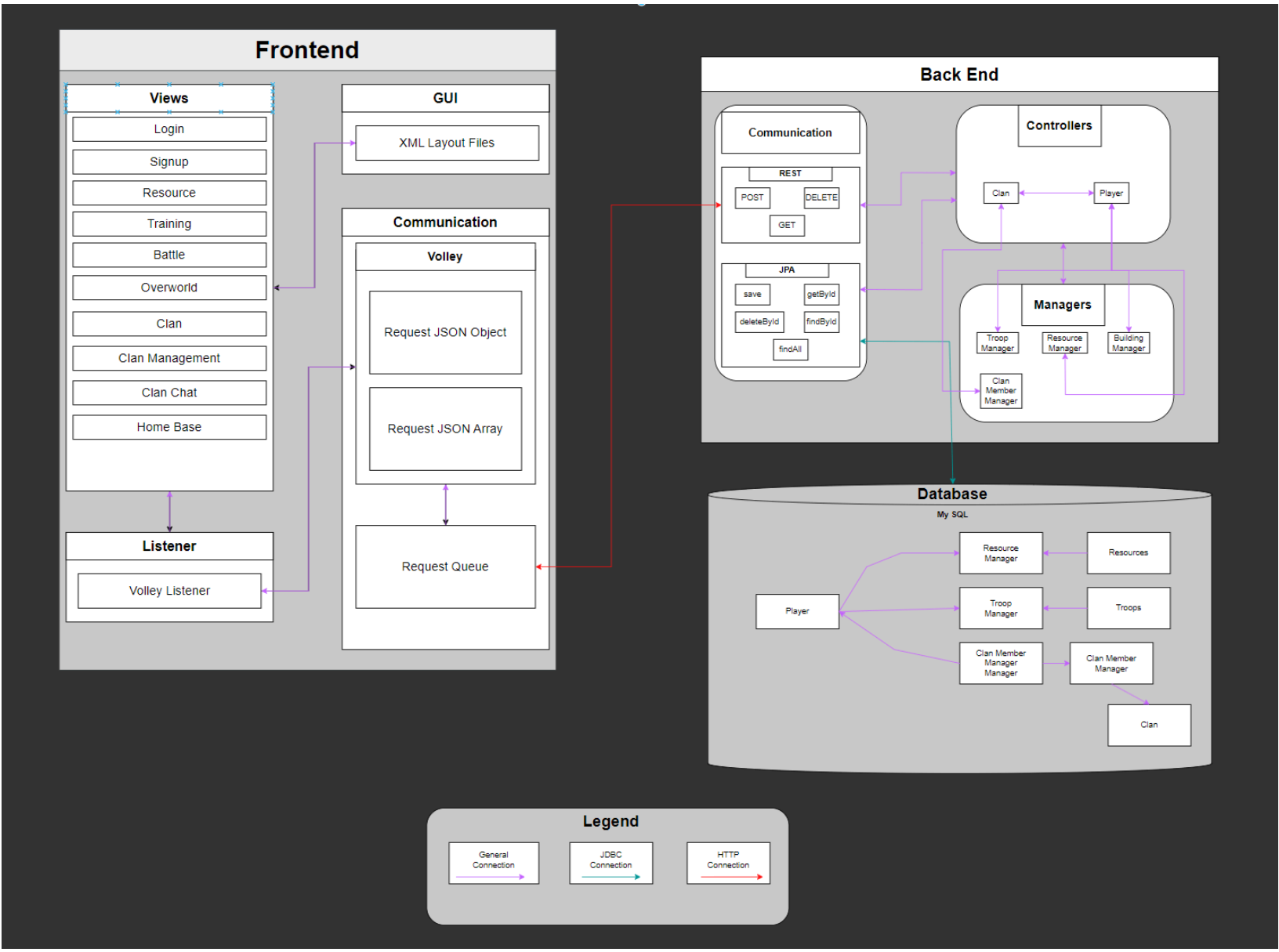
Group SD2_1

Member1 Name: Nicholas Lynch 25% contribution

Member2 Name: Michael Geltz 25% contribution

Member3 Name: Josh Dwight 25% contribution

Member4 Name: Ryan Johnson 25% contribution


Frontend
Views
Login
Signup
Resource
Training
Battle
Overworld
Clan
Clan Management
Clan Chat
Home Base
Listener
Volley Listener
GUI
XML Layout Files
Communication
Volley
Request JSON Object
Request JSON Array
Request Queue
Back End
Communication
REST
POST
DELETE
GET
JPA
save
getById
deleteById
findById
findAll
Controllers
Clan
Player
Managers
Troop Manager
Resource Manager
Building Manager
Clan Member Manager
Database
My SQL
Resource Manager
Resources
Player
Troop Manager
Troops
Clan Member Manager Manager
Clan Member Manager
Clan
Legend
General Connection
JDBC Connection
HTTP Connection

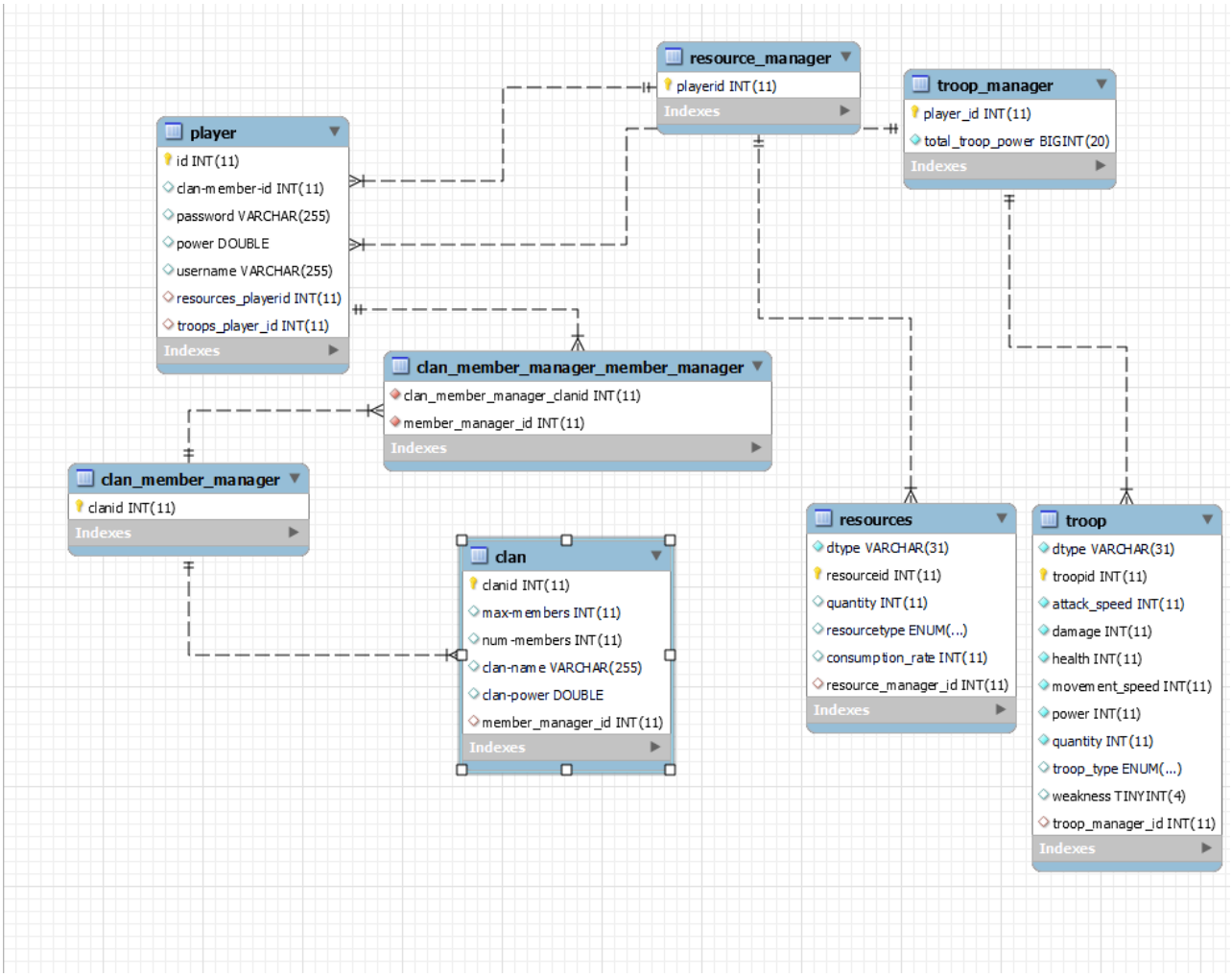
Frontend

- Login
  - User inputs username and password into a textedit.
  - The username is compared to the list of users obtained from a POST request.
  - If the username exists and the corresponding password matches the inputted password, the user is logged in.
- Signup
  - User inputs a username and password into a textedit.
  - Those are sent in a POST request and are added to the list of users in the backend if the username isn't already taken.
  - The user will then be logged into their new account.
- Resource Manager
  - The Resource manager shows the resources the user has and allows them to collect more.
  - When a user chooses to collect, the user ID is sent in a POST request to the backend to retrieve the amount of resources a player should get.
  - The resource manager also retrieves the new resource total from the backend using a POST request.
- Troop Manager
  - The troop manager allows the user to train and view troops.
  - The user selects the amount of troops they wish to train, which is added to a training queue.
  - The queued troops and the user ID are sent to the backend in a POST request, which will calculate the total time needed to train all the troops.
  - After the time needed to train passes, the troops will be added to the user's total.
- Fighting
  - The two users that are fighting will have their IDs stored.
  - The two user IDs are used in a POST request, which sends back data on who won the battle and the casualties suffered on both sides.

Backend

- Communication
  - The backend uses endpoint mappings to update the database based upon the information sent through the URL.
    - POST: Post mapping is used to create OR modify objects/fields stored in the database
    - DELETE: Delete mappings are used to entirely wipe an entity from the database
    - GET: Get mappings are used to retrieve info on whatever request/parameter is passed into it from the database
- Controllers
  - The controllers, only clan and player, at the moment, are used to communicate between the frontend and backend via the endpoint mappings
    - Player: The player controller contains the mappings to create new players, delete players, modify a player's troops, modify a player's resources, and modify their buildings.

        - Clan: The clan controller contains the mappings to create new clans, delete clans, remove player(s) from clan(s), get a list of all clans, get a list of players in a clan, and add players to clan(s). The player management part is done via the Clan Member Manager.
- Managers
    - The managers, Troop Manager, Resource Manager, Building Manager, and Clan Member Manager manage the aspects of the things they are respectively named after.
        - Troop Manager: Manages a player's troops by, adding, removing, calculating a combat result, figuring out how many of each troop type a player has, and calculating power total of troops.
        - Resource Manager: Manages a player's resources by, adding, removing, and figuring out how much of each resource a player has.
        - Building Manager: Manages a player's buildings by, leveling, returning build cost, training time, resource production time, calculating individual or total buildings power, and returns what levels a player's buildings are.
        - Clan Member Manager: Manages adding players from a clan, removing players from a clan, and returning the member list of a clan.


resource_manager
playerid INT(11)
Indexes
troop_manager
player_id INT(11)
total_troop_power BIGINT(20)
Indexes
player
id INT(11)
clan-member-id INT(11)
password VARCHAR(255)
power DOUBLE
username VARCHAR(255)
resources_playerid INT(11)
troops_player_id INT(11)
Indexes
clan_member_manager_member_manager
clan_member_manager_clanid INT(11)
member_manager_id INT(11)
Indexes
clan_member_manager
clanid INT(11)
Indexes
clan
clanid INT(11)
max-members INT(11)
num-members INT(11)
clan-name VARCHAR(255)
clan-power DOUBLE
member_manager_id INT(11)
Indexes
resources
dtype VARCHAR(31)
resourceid INT(11)
quantity INT(11)
resourcetype ENUM(...)
consumption_rate INT(11)
resource_manager_id INT(11)
Indexes
troop
dtype VARCHAR(31)
troopid INT(11)
attack_speed INT(11)
damage INT(11)
health INT(11)
movement_speed INT(11)
power INT(11)
quantity INT(11)
troop_type ENUM(...)
weakness TINYINT(4)
troop_manager_id INT(11)
Indexes