EZ-LFG

EZLFG

Universal Looking for Group Application

06/12/2024

Ethan Moore, Collin Rodgers, Thomas MacGuire

i. ..... Software Requirements Specification 3

1. Project General Description 3

2. Product Features 4
3. Functional Requirements 5
4. Non-Functional Requirements 6
5. Scenarios 7
Users (Ethan Moore / Thomas MacGuire) 7
User Not Logged In 7
User Posts Group Listing 8
User Joins Group Listing 9
System Admins (Collin Rodgers) 10
Admin Login 10
Admin Account Management 11
Admin Game/Group Listing Management 12
ii. .....Design Specification 14
Project Overview 14
Use-Case Model 15
State Machine Diagrams 16
User – Ethan Moore 16
Content Creator – Thomas MacGuire 17
Admin – Collin Rodgers 18
Database Schema 19
Software Architecture - MVC 20

i. Software Requirements Specification

1. Project General Description

EZLFG will serve as an application for gamers to find groups to play multiplayer video games with. The web app will allow the user to create listings to form new groups as well as view listings created by other users and join their group. Most looking for group (LFG) applications are designed specifically, where EZLFG will be designed to work with many games just through one account. Users will be able to make an account which will allow them to link their account from other services containing their game ID (Steam, PlayStation, Xbox, Uplay, etc.).

2. Product Features

EZLFG will serve as a universal group listing service with compatibility across many games. Listed below are the apps features:

Group Hosting: Users will be able to post group listings for their selected game. This listing has a title, tags, and a description of the group so group leaders can find likeminded teammates. Other users will be able to join this listing and leaders can remove members and invite friends.

Group Joining: Users can browse group listings for their favorite games and join groups hosted by other players. Users can ready-up for their group and manage their listings in the "Your Groups" tab within the navigation menu.

Account Creation: Users will create an account before joining or creating any listings. This account will allow the user to manage their groups, link 3rd party accounts, and view and manage friends.

3. Functional Requirements

- FR0: The app will allow all users to create a profile.
- FR1: The app will allow all users to modify their profile.
- FR2: The app will allow all users to import their owned games from steam
- FR3: The app will allow all users to create new groups
- FR4: The app will allow all users to join previously created groups
- FR5: The app will allow all users to search based on all available game
- FR6: The app will allow all users to modify their own ready/not ready flag
- FR7: The app will allow users to flag other users and rate them based off gaming performance
- FR8: Accounts will be protected by login pages with hashed and salted passwords
- FR9: Administrators will be able to remove content outside the content policy
- FR10: Administrators will be able to disable or ban user accounts.

4. Non-Functional Requirements

- NFR0: 5 potential groups will be generated upon login, 5 new groups will be generated as the user scrolls

-NFR1 User can decline and close group in less than 10 seconds

-NFR2 User will be able to create a new group in less than 10 seconds

-NFR3 Ideally each page will refresh every 15 seconds to keep up to date with new groups/listings to join

-NFR4 User data privacy and security will be ensured by using encryption and hash/salted passwords

-NFR5 Systems and servers will be running 24/7 with occasional downtime for maintenance

-NFR6 Scheduled downtimes for Maintenace should only last 2 hours

-NFR7 Will be compatible with most gaming platforms (Steam, PlayStation, Xbox, Uplay, etc.)

-NFR8 Severs will be honed to user's geolocation (if preferred and allowed) to allow more localized group forming

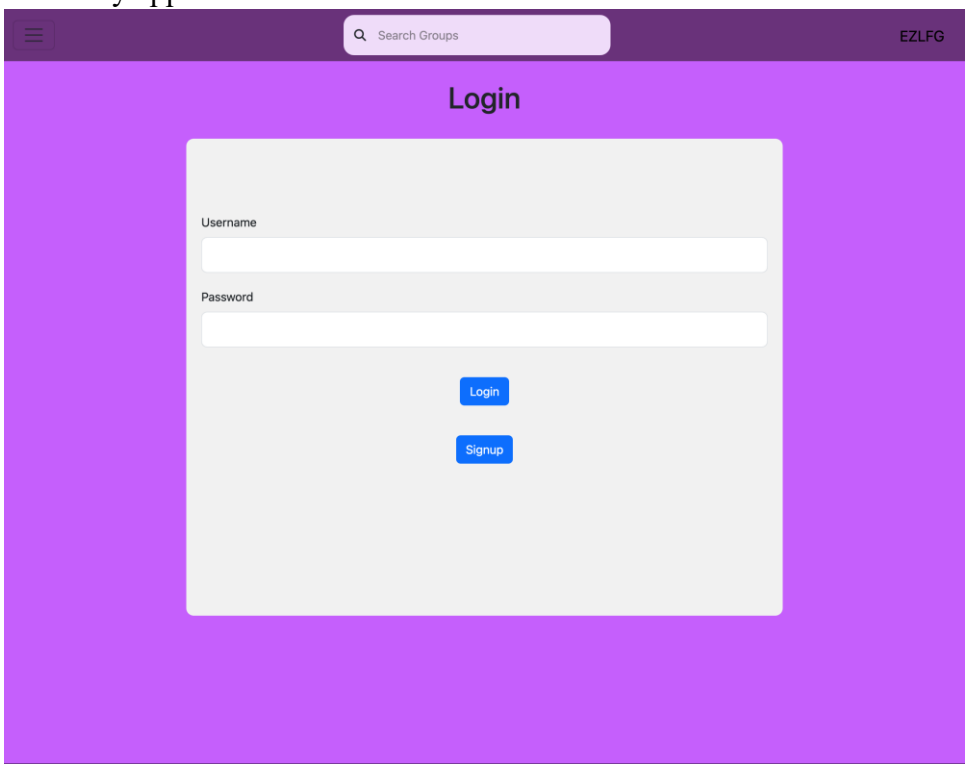
5. Scenarios

Users (Ethan Moore / Thomas MacGuire)

User Not Logged In

- **Initial Assumption**: The user will be able to view game and group listings, but upon selecting any option that requires an account the user will be prompted with an account login screen.
- **Normal**: If the user is not logged in, they will be required to log in if:
  - The user attempts to join a group listing. ○ The user attempts to create a group listing. ○ The user selects the "Friends"/"Your Account"/"Your Groups" tab in the navigational menu.
  - The user attempts to add a game listing to their favorites.
- **What Can Go Wrong**: Users may not have an account registered with the service.
- **Other Activities**: Users can sign up for an account if they do not have one.
- **System State on Completion**: Upon completion the user will be logged in, redirected back to the page they were on, and can perform

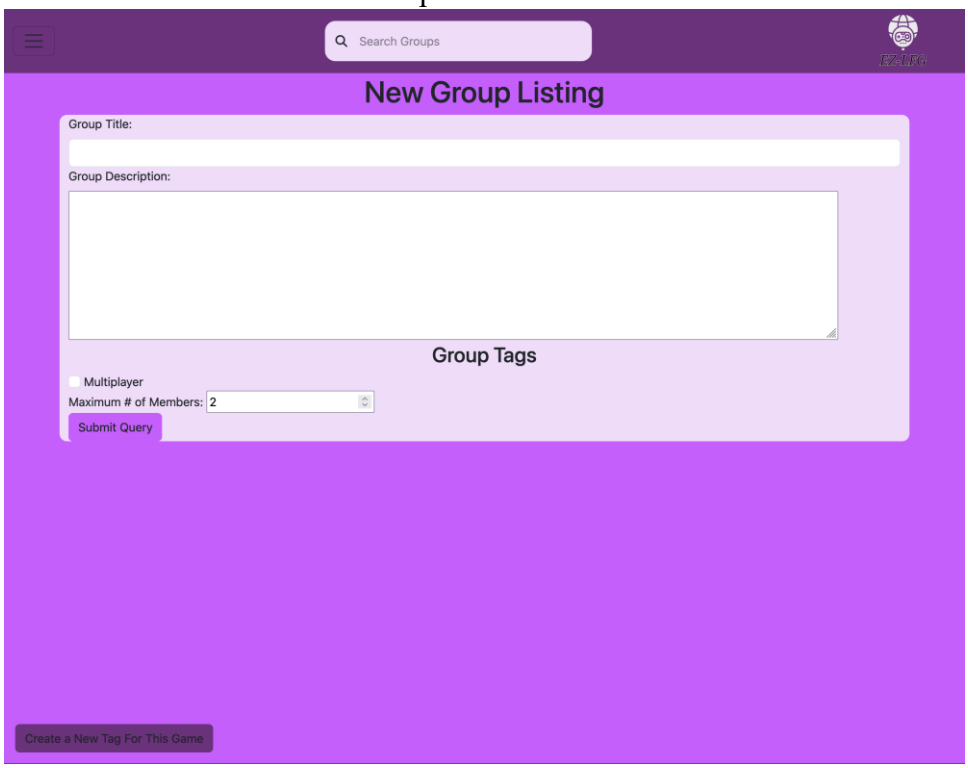
necessary application functions.



User Posts Group Listing

- **Initial Assumption:** Within a game listing users can create a group listing for that game where others can join that group upon viewing the listing.
- **Normal:** If the user is logged in and selects the create listing option they will be asked to select/insert:
  - A group title
  - Group tags
  - A group description
  - A start and end date/time
- **What Can Go Wrong:** Users might try to create a listing without editing or filling out the required fields.
- **Other Activities:** Users can edit or remove their group listing in the "Your Groups" tab.
- **System State on Completion:** Once a group is created by the user it will become available for other users to join and the listing will be

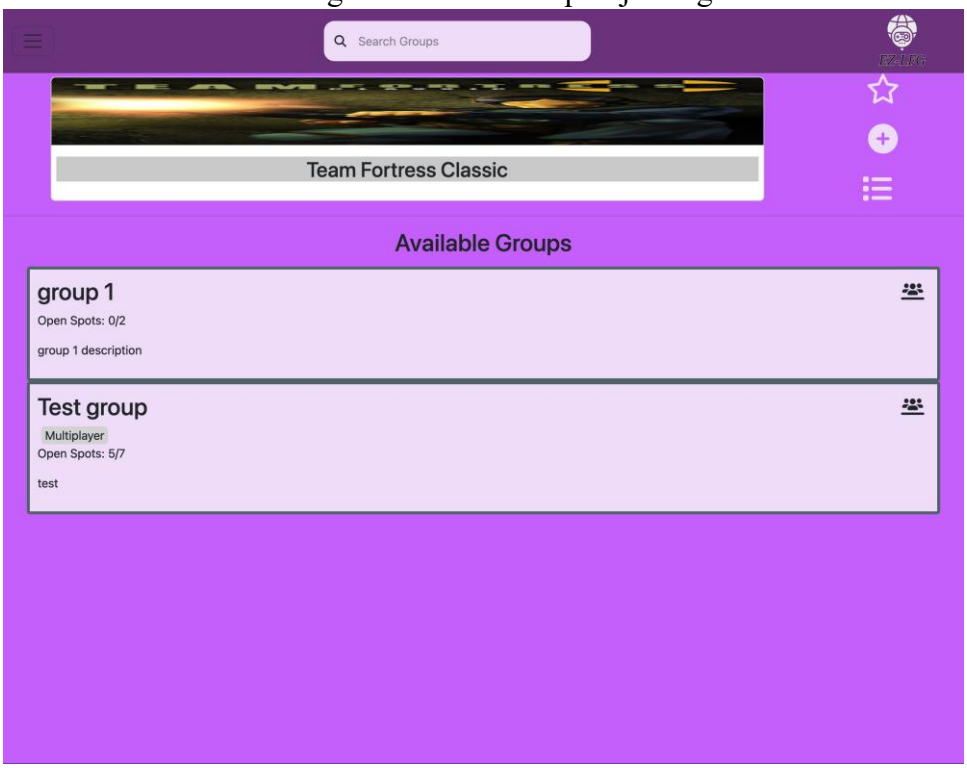
added to the owners “Your Groups tab.



User Joins Group Listing

- **Initial Assumption:** Users can join group listings that suit their interest. Users can also manage their status within the group and message group members
- **Normal:** Users will select the join option on group listings they would like to be in. Users can view and manage their joined groups in the “Your Groups” tab in the navigation menu. Users may also message group members on the same page.
- **What Can Go Wrong:** Users could come across toxic users that diminish the experience of the app. In this case users have the ability to report accounts for system admins to flag and review.
- **Other Activities:** Users can leave groups at any time or may be removed by group leaders.
- **System State on Completion:** Users will become members of the joined group and will coordinate with other members of the group and the group leader to have an enjoyable gaming experience. User IDs

will be added to the listing in the database upon joining.

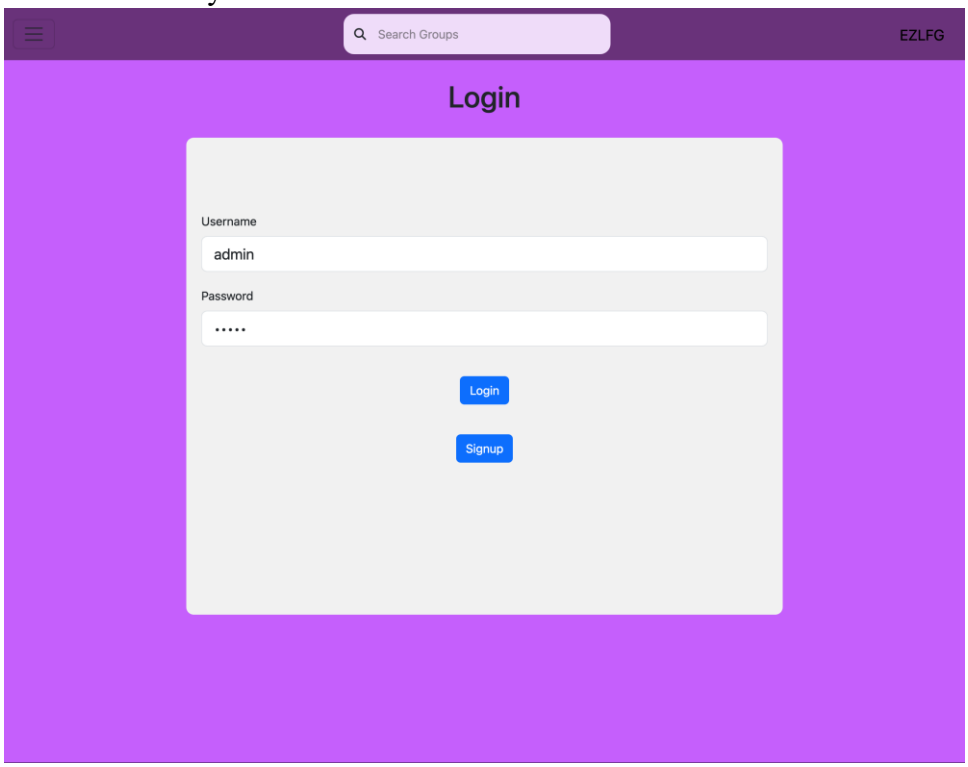


System Admins (Collin Rodgers)

Admin Login

- **Initial Assumption:** System admins will be able to log in through a dedicated Admin Login page. Upon successful login they will be redirected to the system admin interface.
- **Normal:** If the client is not logged in to a user account and visits the admin login page through a specific URL, they will be able to attempt to log into an admin account. Unlike a user account, the client is logged out upon session termination for security purposes.
- **What Can Go Wrong:** An application user can make their way to the admin login page. To counteract this, if the client is logged into a user account, they will be redirected to the application home page upon attempting to access the admin login page.
- **Other Activities:** There are no other activities associated with this action.

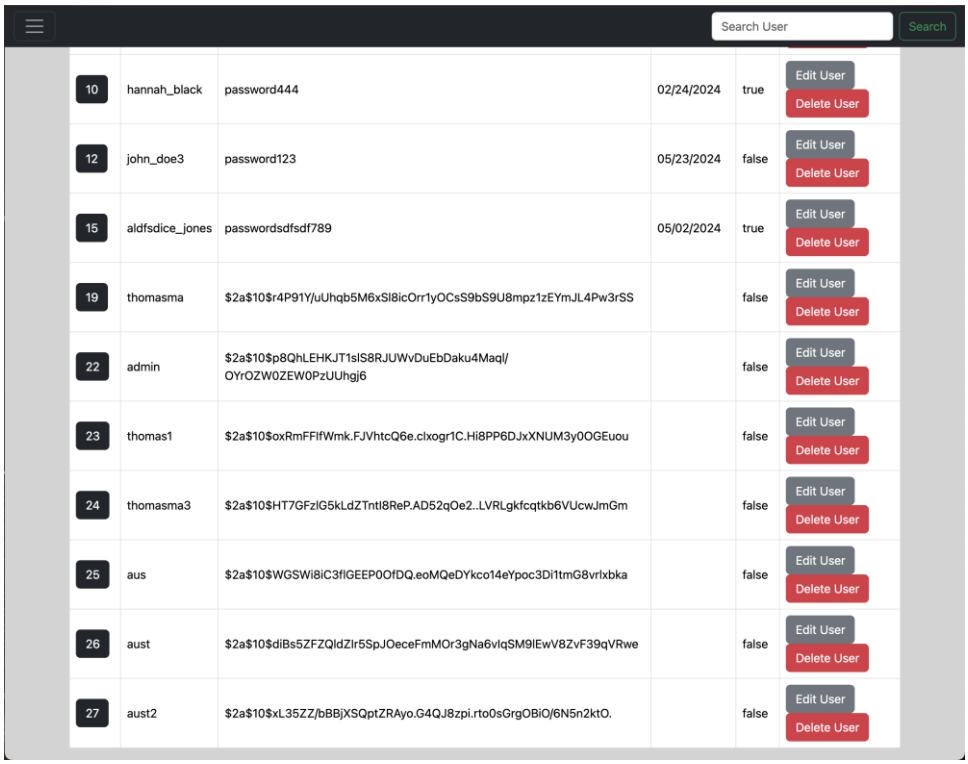
- **System State on Completion:** Upon successful login the client will be directed to the system admin interface



Admin Account Management

- **Initial Assumption:** System admins will be able to manage user accounts via the admin interface.
- **Normal:** System admins can query user accounts by username. Admins will be able to review accounts, flag user accounts and remove accounts.
- **What Can Go Wrong:** Admins might remove a wrong account or wish to revert their decisions. A log of all transactions made by admins to the database will be stored on the server.
- **Other Activities:** Admins can refresh their queries to account for database changes.
- **System State Upon Completion:** Changes made by admins will be reflected in the database and user accounts will be able to be successfully managed by system admins. This allows for a level of

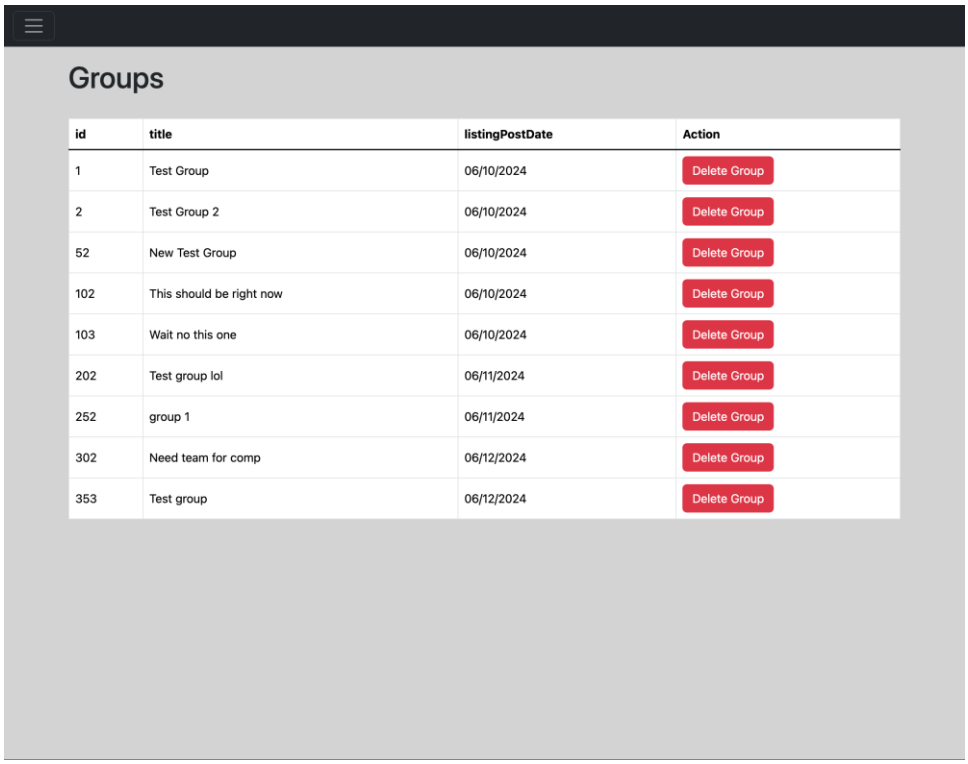
moderation on user accounts.



Admin Game/Group Listing Management

- **Initial Assumption:** Within the admin interface, system admins will be able to manage game and group listings.
- **Normal:** Admins can query, update, and remove game and/or group listings within the admin interface. Queries can be made based on any of the attributes associated with the listing. Updates can be made by editing those same attributes.
- **What Can Go Wrong:** Admins may abuse their power, remove a wrong listing or just wish to revert their decisions. A log of all transactions made by admins to the database will be stored on the server.
- **Other Activities:** Admins can refresh their queries to account for database changes.
- **System State Upon Completion:** Game and group listings will stay updated, moderated and managed by system admins. Changes made by

admins will be reflected within the database.

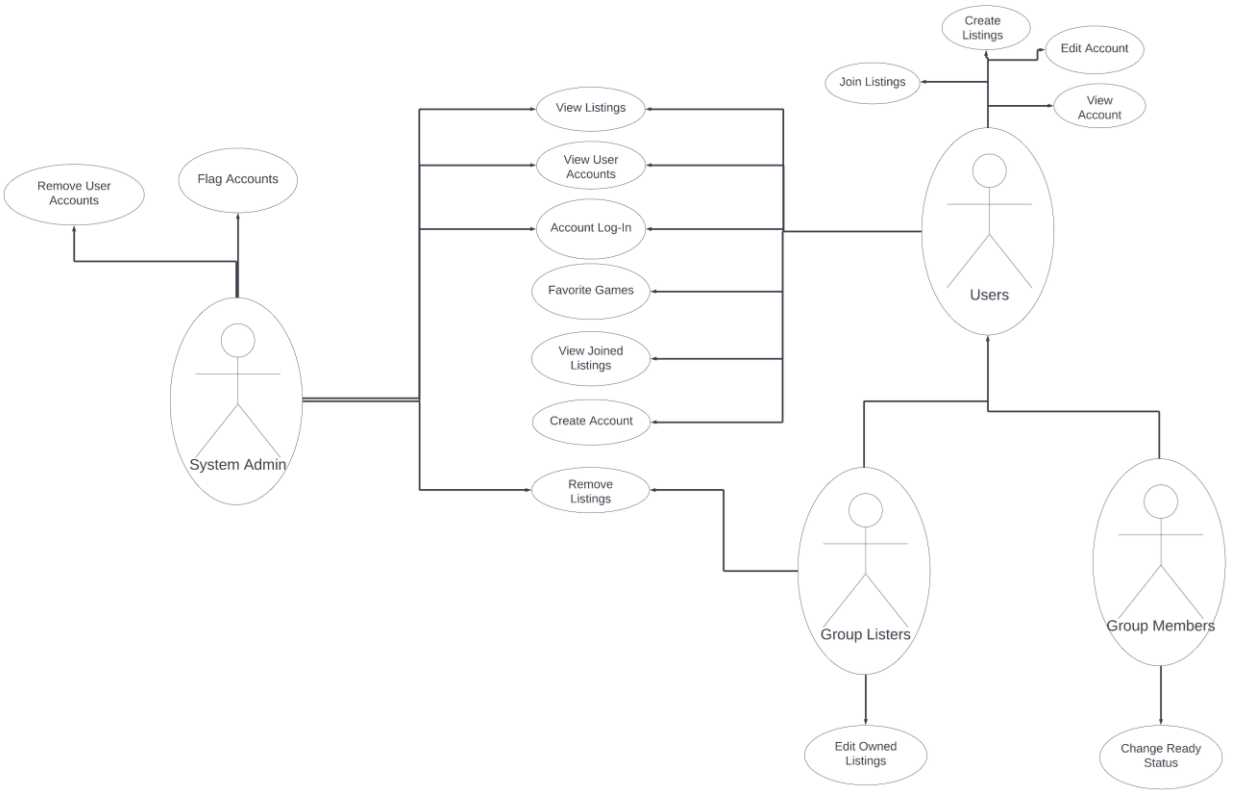


ii. Design Specification

Project Overview

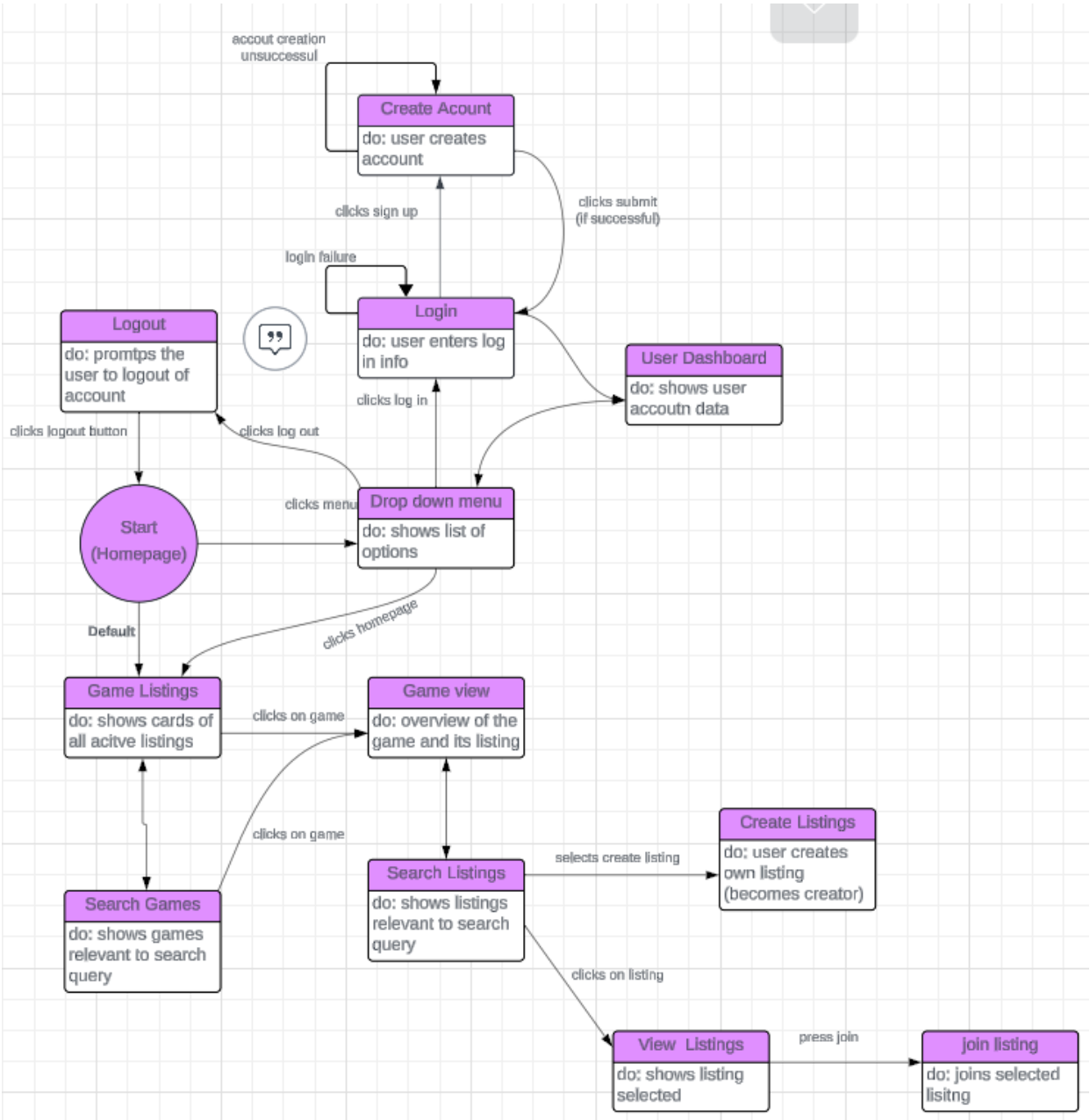
EZLFG will serve as an application for gamers to find groups to play multiplayer video games with. The web app will let the user create listings to form new groups and view listings created by other users and join their group. Most looking for group (LFG) applications are designed specifically, where EZLFG will be designed to work with many games just through one account. Users will be able to make an account which will allow them to link their account from other services containing their game ID (Steam, PlayStation, Xbox, Uplay, etc.).

Use-Case Model

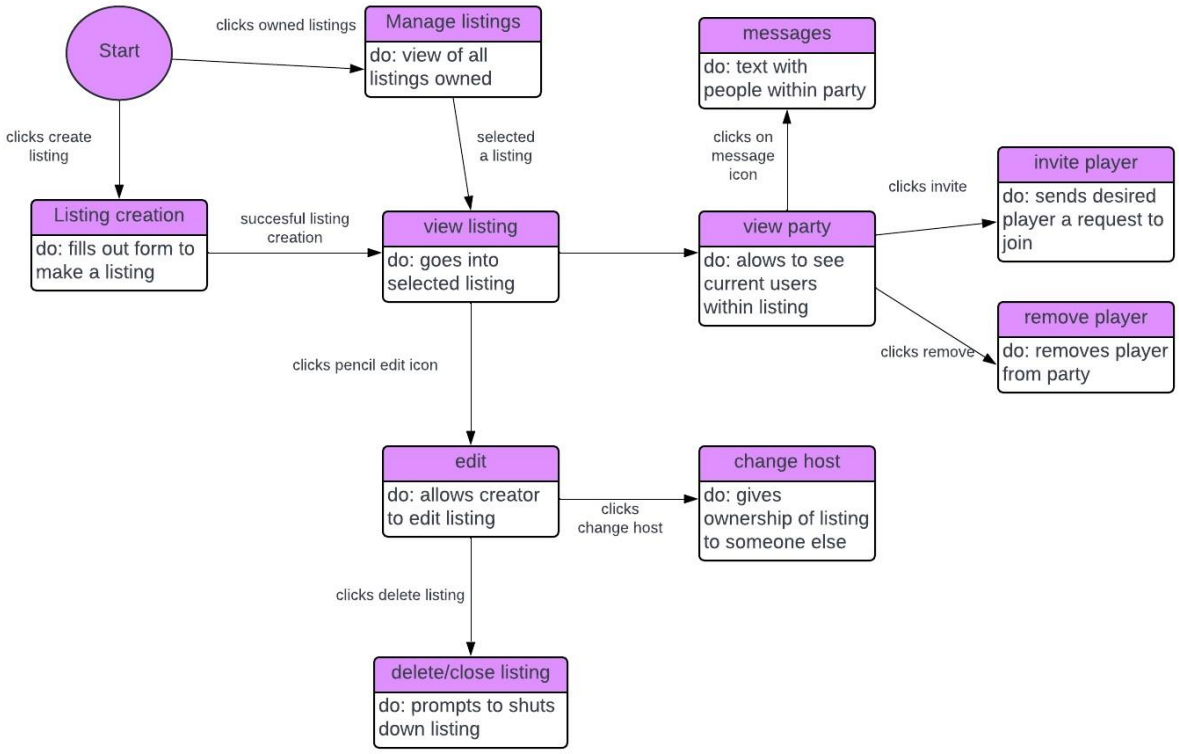


State Machine Diagrams

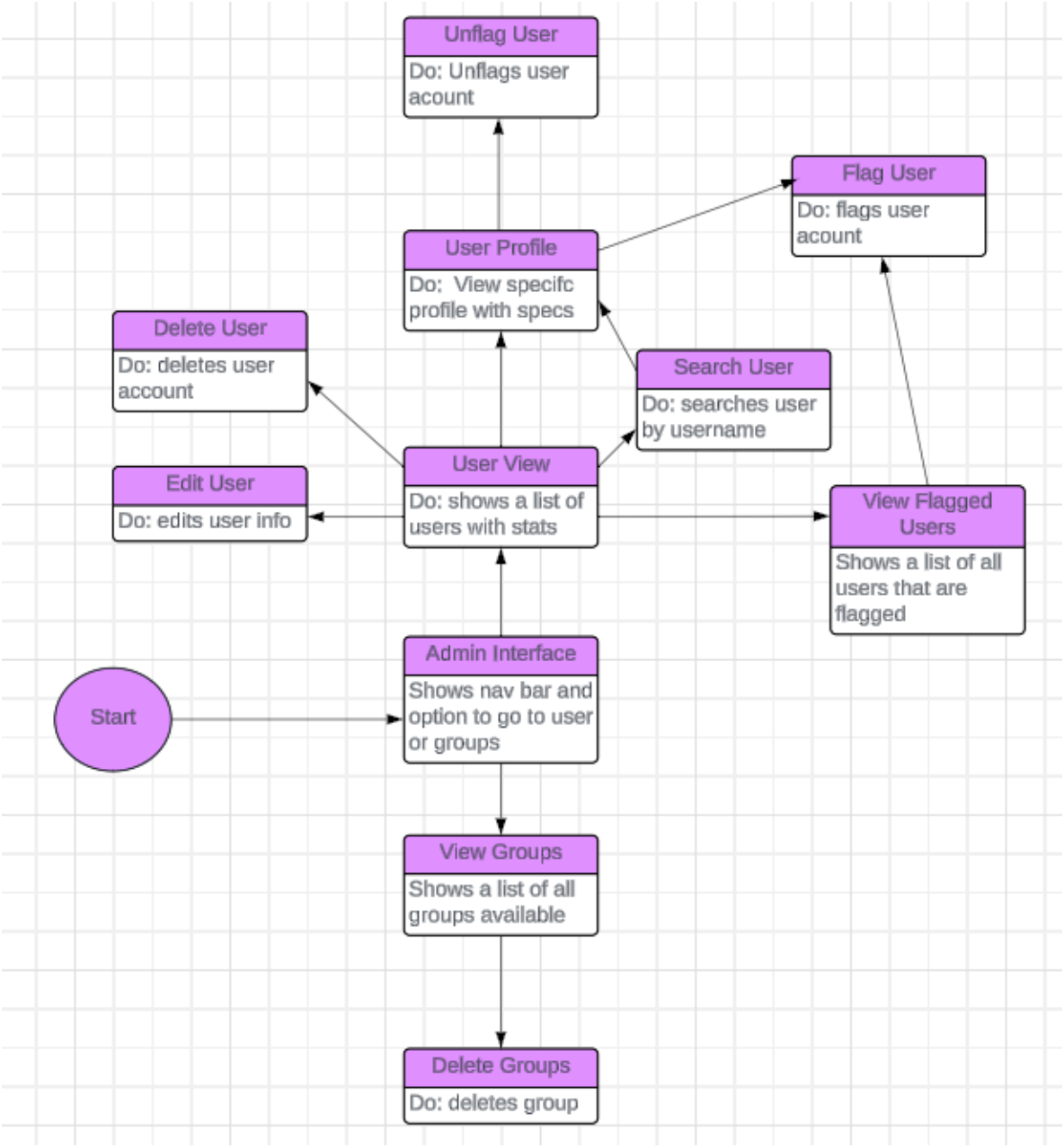
User – Ethan Moore



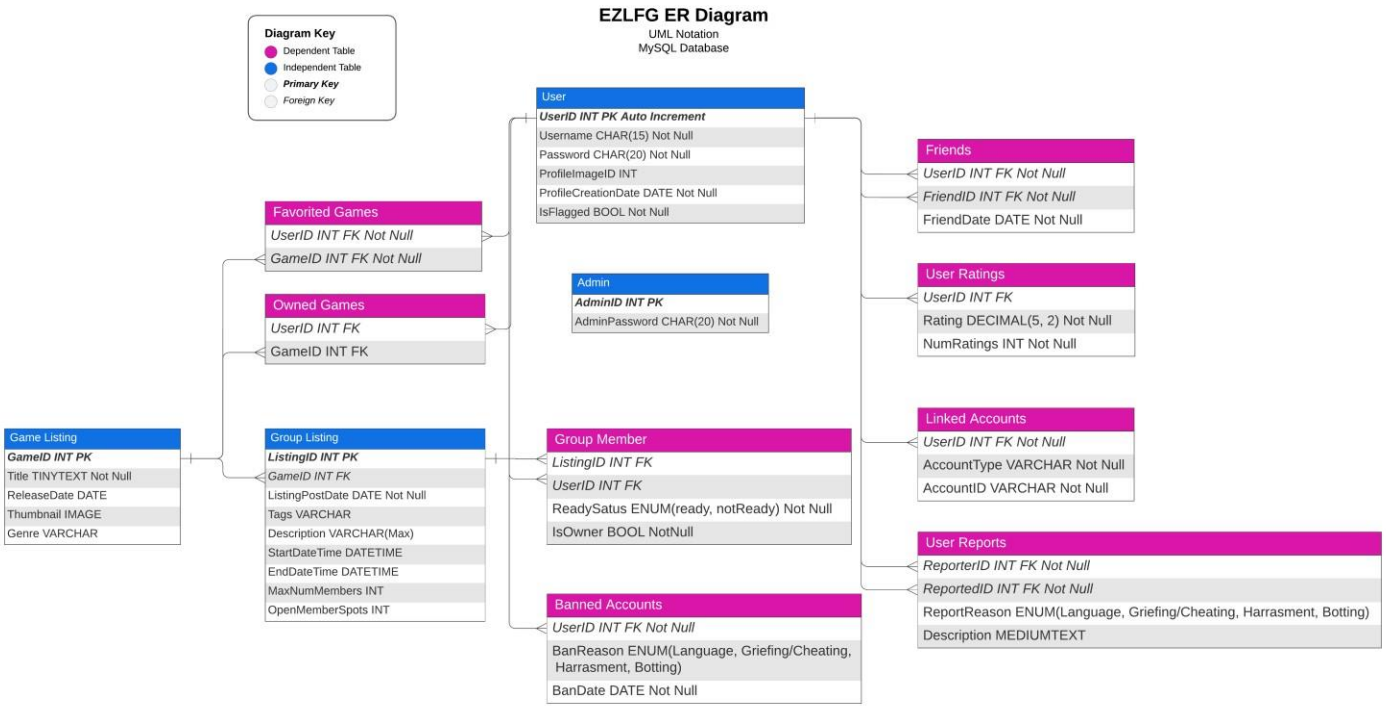
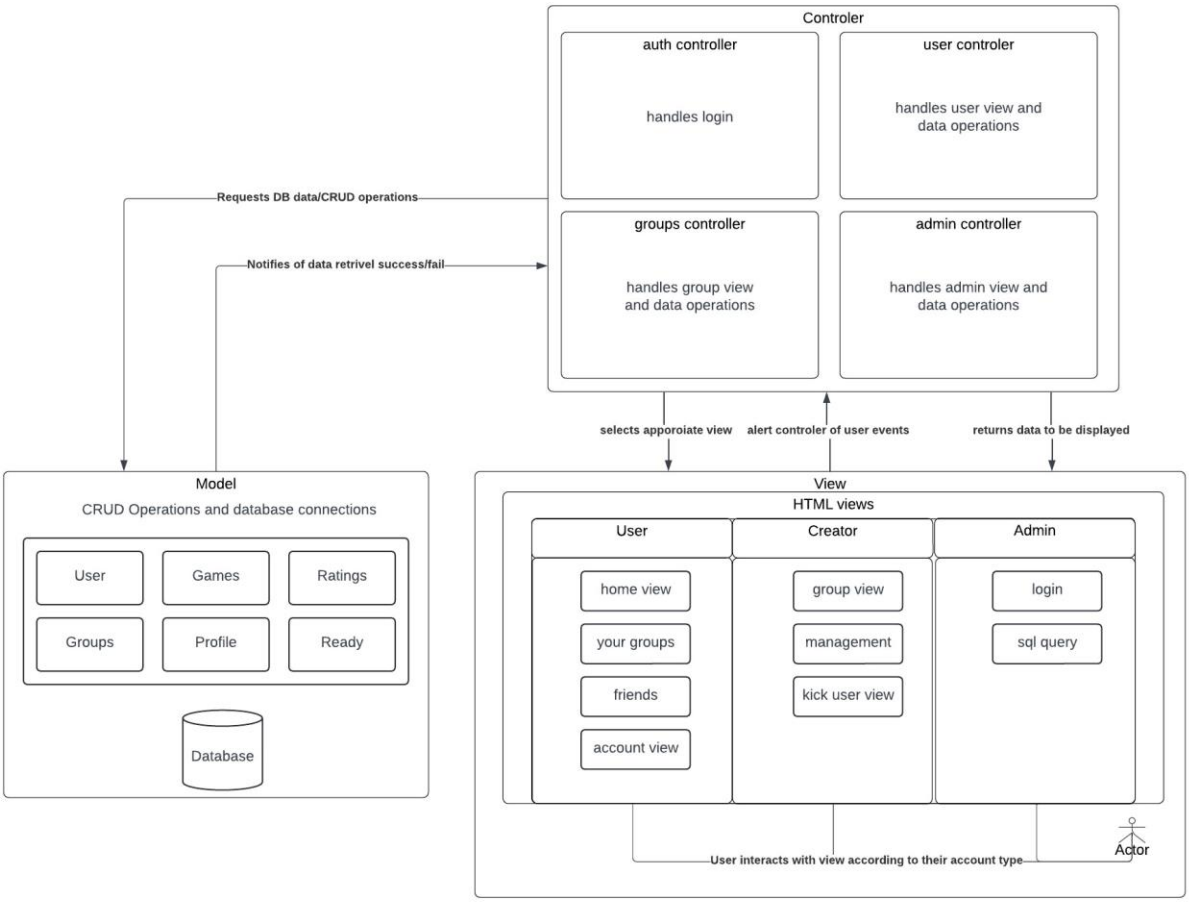
Content Creator – Thomas MacGuire



Admin – Collin Rodgers



Database Schema



Software Architecture- MVC