Software Engineering

Software Requirements Specification

(SRS) Document

CarMate

[9/22/23]

[Version 1]

By: Nick Brandsma, Priyankaben Shah,

Saniyah Khan

[UNCG Academic Integrity/ Honor Code : My words and actions will reflect Academic Integrity.

I will not cheat or lie or steal in academic matters.

I will promote integrity in the UNCG community.]

Table of Contents

1. Introduction

1.1. Purpose

The goal of the CarMate's application is to provide a safe and engaging platform for drag racing enthusiasts. It offers a variety of features including user registration, event hosting by drag strips, admin oversight, safety agreements, location-based services, and real-time communication. This application will help to keep street racers off the street and give many drag strips a greater chance at staying open.

1.2. Document Conventions

The purpose of this Software Requirements Document (SRD) is to describe the client-view and developer-view requirements. For the time period of this project this will serve as a reference point in the many versions by providing details on the project's functions, data, the scope and expectations. In the document, the team will also provide users served by the system and limits to their profile base. Developer-oriented requirements outline software requirements, and applications in detail for both client and developer team to stay updated.

1.3. Definitions, Acronyms, and Abbreviations

Java	A programming language originally developed by James Gosling at Sun
Github	Cloud-based service for software development and version control to store and manage our code.
VS Code	Visual Studio Code. Source-code editor.
Bootstrap	Open-source CSS framework. This will help with front-end web development in using our application across different mobile devices.
.HTML	Hypertext Markup Language. This is the code that will be used to structure and design the web application and its content.
CSS	Cascading Style Sheets. A sheet language that is used for presenting documentation in HTML or XML.
API	Application Programming Interface. Where two or more computer programs are able to communicate with each other.
MongoDB	Easier for developers to store structured or unstructured data.

JavaScript	Programming language is going to be used for both client-side and server-side to make the webpage interactive.

1.4. Intended Audience

Stakeholders

1. Stakeholder
    a. Client: Drag Strip Venues
    b. Project Manager: Priyankaben Shah
    c. Drag-Race Consultant: Nickolas Brandsma
    d. Developers: Priyankaben Shah, Nickolas Brandsma, Saniyah Khan
    e. Users: car enthusiasts using the application
2. The primary stakeholders that are the driving force behind this project. The stakeholders are drag strip venues. Key sections of interest for the client include:
    a. 1.1 - Purpose: Summarizes the project's goal and expected benefits for clients.
    b. 1.2 - Document Conventions: Description of the main objectives of the Software Requirement Document (SRD) in context of our application, CarMates.
    c. 1.5 - Project Scope: Specifies the software goals with the overall business goals and the benefits of the project for the client and project manager to stay on track.
    d. 1.6 - Technology Challenges: Technological constraints and performance expectations, for clients and developers to

Developers

1. The development team is responsible for creating the software system. Sections relevant to developers include:
    a. 2.3-User Class and Characteristics: Categorizing the users of the software is intended for

b. 2.4-Operating Environment: The environment of the software that its being designed to operate in
c. 2.6-Assumptions and Dependencies- A list of assumptions regarding the software product and the environment along with any dependencies that can affect the project
d. 3.1-Primary: All the requirements within the system or subsystem to determine the output that the software is expected to give in relation to input
e. 3.2-Secondary: Functions that are used to support the primary requirements
f. 4.2-Interface Requirements
g. 5.1- Performance Requirements: performance requirements for all the functional requirements

Project Managers

1. Project Managers oversee the entire project including planning, execution, and monitoring. Sections relevant to project managers include:
    a. 2.1-Product Perspective: The origin of the product
    b. 2.2-Product Features: An in-depth summary of the functions of the software that would perform and features included.
    c. 2.5-Constraints: Limiting factors that pose a challenge to the development of the software
    d. 4.1-Operating System and Compatibility: environments that will be needed to operate the system
    e. 5.2-Safety Requirements: safeguards that need to be incorporated as a measure against the harm of the use of the software application may cause.
    f. 5.3-Security Requirements: privacy and data protection regulations that is neede while designing the product

g. 5.4-Software Quality Attributes: Additional qualities that need to be incorporated within the software
h. 5.5-Process Requirements
i. 5.6-Other Requirements

Users

1. System operators interact directly with the software system. They can view the system's features and functionalities from a user-centric view. Sections relevant to users include:
    a.

1.5. Project Scope

The goal of CarMate is to provide an easy-to-use interface for all users including, admins, drag strip accounts and general users to interact and possibly attend events hosted by drag strips and monitored by admins. This software application aligns with car/drag race enthusiasts for their hobbies, and fostering a community, as well as allow drag strips to use their locations for enthusiasts that are not necessarily racing professionally.

The overall software goal in lieu of business goals is to create a bridge for the drag racing community with each other while promoting safe and legal racing practices. This aligns with the overall client's business goals, as drag strip venues can promote events, gain sponsorship, create awareness of their location to increase visibility of location and participation; to align with monetization objectives. This application will also focus on allowing a general user-friendly experience with features such as live chat, preferred locations of users, and event notifications.

The successful development and implementation of CarMate will result in creating an application for the drag racing community to communicate with and increase participation in drag racing events whether it is as a racer or audience. For drag strip venues the app can make promotional events based on data received by the app's user preferences to ensure an increase in revenue by reaching a targeted audience. Most importantly, CarMate will be able to promote responsible racing culture to represent a positive image on the racing community.

1.6. Technology Challenges

Some technology challenges throughout the project would be the live chat, as a possible language is javascript and react/redux for backend and frontend. Though our team has some experience, it will be a learning experience. We have learned about API in class and implementation techniques. There are still other factors that we will look through when starting our implementation that we have yet to consider including finding an open API for car models when a profile is created.

1.7. References

This section is subjected to changes per team's criteria.

Car API: https://api-ninjas.com/api/cars

Javascript program npm: https://www.npmjs.com/

W3schools bootstrap: https://www.w3schools.com/bootstrap/

Drag race directory: http://speedwaysonline.com/tracks/category/dragstrips/

2. General Description

2.1. Product Perspective

Illegal street racing and takeovers is a persistent problem that leads to accidents, injuries, and destruction of property. This application provides a safer alternative for members of the car community to race. It promotes legal drag racing at verified drag stips. It will allow users to easily connect with one another and facilitate racing at a drag strip as opposed to the street.

2.2. Product Features

This application in early phases will have admins to only allow valid users to prevent bot accounts, the admins will also be able to delete any user that does not meet guidelines. Another tier of accounts is drag strip venues for them to host and post event notifications to either server or targeted audience who have preferred locations. These accounts will also have to be approved

by admins. Finally there are the majority of accounts, the users who can interact in the app via creating a profile with the details of their vehicle and allowing preferred locations to be active to interact with nearby car enthusiasts through a live chat feature. The team will also continue to discuss with clients on the posting feature but that will be updated in the early stages.

2.3. User Class and Characteristics

Our application will be able to be used by anybody but for those who want to get the most out of it, the users will be ones who are a part of the local car community and are interested in drag racing their car. Other users who aren't interested in racing are still welcome and could use the app for notifications about events they would like to spectate. The users will not need to have any kind of special skill set to operate the app.

2.4. Operating Environment

The application is designed to operate on the web across many different devices.

2.5. Constraints

Limiting factors will be updated as the project progresses.

2.6. Assumptions and Dependencies

The software will be developed using VS Code and its go live feature to stay updated on the look of the webpage. The API used will be at this stage Car API [https://api-ninjas.com/api/cars] for a user to add their car type to profile. The live chat could create issues as we plan to use react and redux and have moderate knowledge of it but are excited to use and learn. Some other features regarding the preferred location have yet to be finalized in whether using live location via geo location or generally through an API with cities and states listed.

3. Functional Requirements

3.1. Primary

- FR0: The system will allow the user to pick which car they drive from a drop down menu.

- FR1: The system will allow the user to enter the location they want to be associated with to find nearby dragstrips

- FR2: The system will allow the user to connect with other users to set up races at nearby dragstips

3.2. Secondary

[Some functions that are used to support the primary requirements.]

- Each users information will be password protects when logging into their account

- Only authorized drag stip accounts will be able to edit the information for drag strips

4. Technical Requirements

4.1. Operating System and Compatibility

The application will be compatible with any operating system that is able to view and to interact with traditional web pages.

4.2. Interface Requirements

4.2.1. User Interfaces

[The logic behind the interactions between the users and the software. This includes the sample screen layout, buttons and functions that would appear on every screen, messages to be displayed on each screen and the style guides to be used.]

Will be updated as the project progresses.

4.2.2. Hardware Interfaces

[All the hardware-software interactions with the list of supported devices on which the software is intended to run on, the network requirements along with the list of communication protocols to be used.]

This application will run on any hardware device that has access to the internet, the ability to display webpages, and the ability to interact with web pages. This includes, but is not limited to smartphones, tablets, desktop computers, and laptops.

4.2.3. Communications Interfaces

[Determination of all the communication standards to be utilized by the software as a part of the project]

It must be able to connect to the internet as well as the local database on phpMyAdmin.

The communication protocol, HTTP, must be able to connect to the World Time API and return the current date and time.

4.2.4. Software Interfaces

[The interaction of the software to be developed with other software components such as frontend and the backend framework to the used, the database management system and libraries describing the need and the purpose behind each of them.]

We will use React and Spring Boot ThymeLeaf to help build the frontend, as well as JPA for the backend database functionality. We will also use Spring Boot with Java to connect the frontend to the backend.

5. Non-Functional Requirements

[Constraints on the services or functions offered by the system (e.g., timing constraints, constraints on the development process, standards, etc.). Often apply to the system as a whole rather than individual features or services.]

5.1. Performance Requirements

[The performance requirements need to be specified for all the functional requirements.]

- NFR0(R): The local copy of the vehicle violation database will consume less than 20 MB of memory
- NFR1(R): The system (including the local copy of the vehicle violation database) will consume less than 50MB of memory
- NFR2(R): The novice user will be able to create and print a ticket in less than 5 minutes.
- NFR3(R): The expert user will be able to create and print a ticket in less than 1 minute.

Will be updated as the project progresses.

5.2. Safety Requirements

We will have a legal notice pop up for all users that will have them accept liability with the dangers that come with drag racing as well as stating that they will use this application for its intended purpose and not use it for street racing.

5.3. Security Requirements

[Privacy and data protection regulations that need to be adhered to while designing the product.]

- NFR4(R): The system will only be usable by authorized users.

5.4. Software Quality Attributes

[Detailing on the additional qualities that need to be incorporated within the software like maintainability, adaptability, flexibility, usability, reliability, portability etc.]

5.4.1. Availability

The application will be available to use 24/7.

5.4.2. Correctness

[Details] - TBD

5.4.3. Maintainability

The application will need the admins to maintain it through fixing bugs and adding new features and dragstrips that apply to be accepted.

5.4.4. Reusability

This application can be used in different communities that share similar interests and altered to their needs as admins and users will be recyclable and accounts such as drag strip venues to be altered per the new project and client interests.

5.4.5. Portability

This application will be able to be used on a downloadable application on any device.

5.5. Process Requirements

This section will be updated as building starts.

5.5.1. Development Process Used

[Software Process Model]

5.5.2. Time Constraints – This section will be updated as building starts.

5.5.3. Cost and Delivery Date

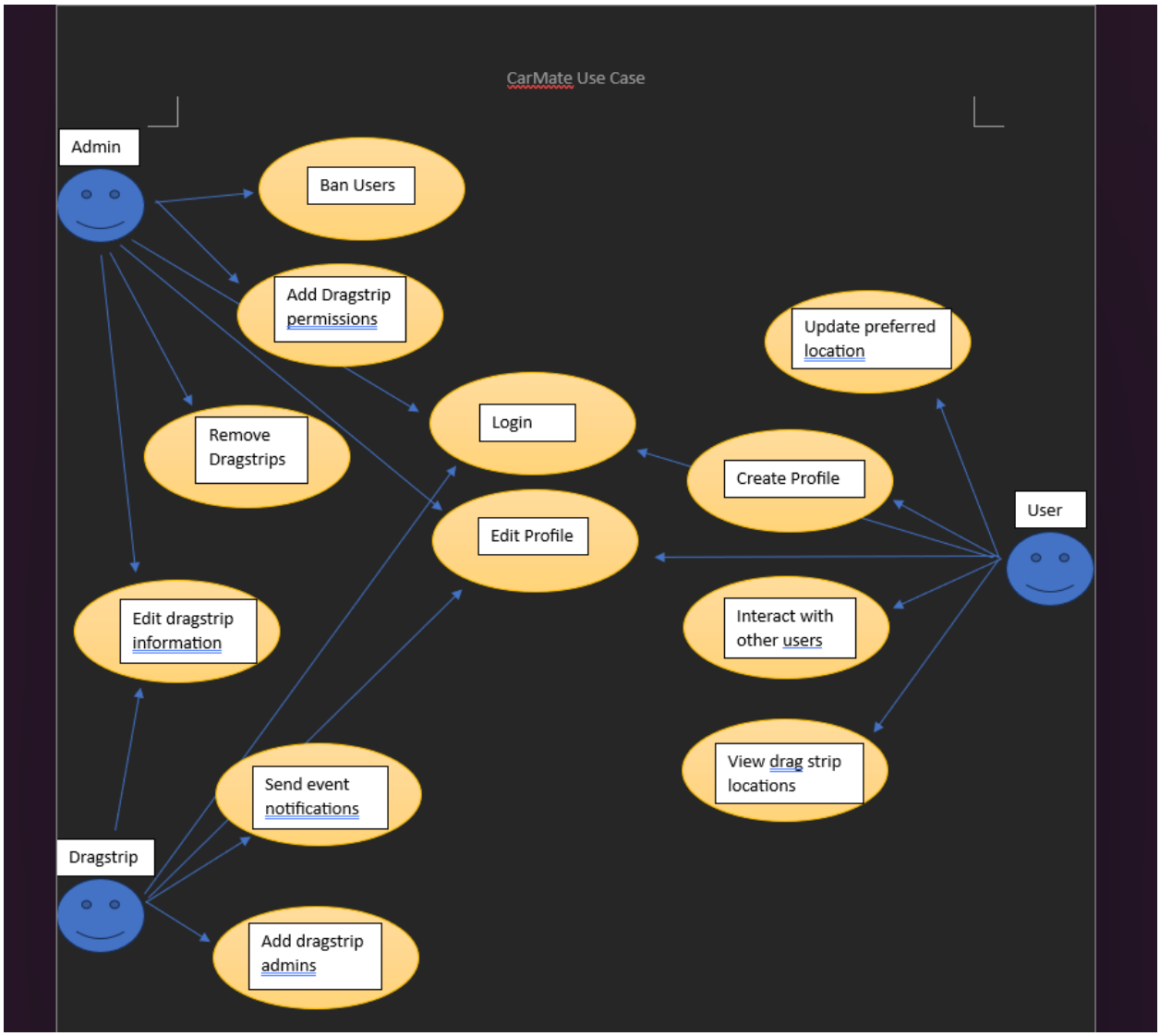
Delivery Date: December 6, 2023

Cost: To be negotiated

5.6. Other Requirements

This section will be updated as building progresses.

5.7. Use-Case Model Diagram



5.8. Use-Case Model Descriptions

5.8.1. Actor: Sarah (Saniyah Khan) = Admin

- **Use-Case Name**: Ban Users

The ability of authorized administrators of the web application to impose restrictions on specific users who violate any of the terms and conditions set by the web application. Examples of violating these terms can include, sending inappropriate messages, posting explicit content, etc. They will either not be able to use the web application for a duration of time or not be able to come back to the web application as a user.

- **Use-Case Name**: Add Dragstrip Permissions

The ability of authorized administrators of the web application to grant read and write permissions for the Dragstrip user to access and update information relating to the dragstrip. Assign administrative permissions to specific users of the Drapstrip venue to manage the drapstrip's event control, special events, event time, event security, etc.

-User-Case Name: Remove Dragstrips

The ability of authorized administrators of the web application to remove the dragstrip from the web application. The admin can remove the dragstrip. The user is met with a confirmation dialog displaying a message that the dragstrip is about to be removed from the application and data associated with it will not be accessible afterwards. The user can either select YES or NO.

-User-Case Name: Edit Dragstrip Information

The user with administrative rights will be able to modify or update information regarding the dragstrip. After they are done making the changes needed, they will need to confirm if they want to save the changes or not.

-User-Case Name: Login

The administrator will enter in a username and password to login. They will need to have administrative level credentials to successfully log in to the system.

-User-Case Name: Edit Profile

The administrator will need to have the correct credentials to edit profiles of the user in the system.

5.8.2. Actor: Actor Name (Priyanka) - Users

- **Use-Case Name:** Update preferred location

When creating a profile, the user can choose their preferred locations to see nearby users to connect with. This can be enabled or disabled per preference but when enabled this data can be shared with drag strip venues for advertisement purposes to reach a targeted audience.

- **Use-Case Name:** Create Profile

Any user is able to create a profile with a nickname or username, also required for a birthdate to ensure age limitations of 18+ in this software. The user is then able to add their car to their profile to start connecting with others, this can include from just model of car to specifications of their car.

- **Use-Case Name:** Interact with users

This application was based upon fostering a community in the car community, so after setting up profile users are able to use live chat to talk to like minded people and share highlights of their vehicles.

- **Use-Case Name:** View dragstrip locations

Users can check for nearby drag strip venues for events or meetups to promote the venues and its events and allow for a safe space to essentially race legally.

-User-Case Name: Login

The administrator will enter in a username and password to login. They will need to have administrative level credentials to successfully log in to the system.

-User-Case Name: Edit Profile

The administrator will need to have the correct credentials to edit profiles of the user in the system.

5.8.3. Actor: Piedmont Dragway (Nick Brandsma) - Drag strips

-User-Case Name: Login

The administrator will enter in a username and password to login. They will need to have administrative level credentials to successfully log in to the system.

-User-Case Name: Edit Profile/Drag Strip information

The administrator will need to have the correct credentials to edit profiles of the user in the system. When the system sees that the user is a dragstrip account, they will be able to edit the information about the given dragstrip.

- **Use-Case Name:** Send event notifications

The administrator will be able to send out a mass notification to nearby users about upcoming events at their dragstrip.

- **Use-Case Name:** Add dragstip admins

The administrator will be able to look for and add other users without the dragstrip admin privileges to be linked with their account. That user account will then have administrator privileges that the dragstrip account has.

5.9. Use-Case Model Scenarios

5.9.1. Actor: Sarah (Saniyah Khan)-Admin

- **Use-Case Name:** Removing Dragstrips

  ● Initial Assumption: Administrator is able to remove the dragstrip from the web application.

  ● Normal: The user with administrative privileges removes the dragstrip from the app.

  ● What Can Go Wrong: The user with administrator credentials accidentally removes a dragstrip that is still needed or in use. The user is met with a confirmation dialog displaying a message that the dragstrip is about to be removed from the application and data associated with it will not be accessible afterwards.

  ● Other Activities: The user can either click YES or NO on the dialogue box.

● System State on Completion: Depending on their action, the drapstrip is deleted or the dragstrip is not deleted.

- **Use-Case Name:** Add Dragstrip Permissions

● Initial Assumption: The process of granting and configuring specific access or privileges to users to edit Dragstrip related actions.

● Normal: The user with administrator privileges is able to read, write, edit, delete, etc., dragstrip related actions

● What Can Go Wrong: The user may not be of administrator level yet, which does not allow them to add dragstrip permissions

● Other Activities: The system will show a dialog box showing a warning or error message.

● System State on Completion: The user can contact the administrator if they need to correct the selection.

- **Use-Case Name:** Ban Users

● Initial Assumption:The ability of authorized administrators of the web application to impose restrictions on specific users who violate any of the terms and conditions set by the web application.

● Normal: The administrator is able to ban the user from the system.

● What Can Go Wrong: The selected user may not exist in the system.

● Other Activities: The system will provide an error message.

● System State on Completion: The administrator will need to correct their selection of the user they want to ban.

- **Use-Case Name:** Edit Drapstrip Information

● Initial Assumption: An authorized user will be able to modify and update details relating to dragstrip information.

● Normal: The user with administrative rights will be able to modify or update information regarding the dragstrip.

● What Can Go Wrong: The dragstrip that is selected may not exist in the system.

● Other Activities: The system will provide an error message stating that the dragstrip does not exist.

● System State on Completion: The user will need to correct their selection of the dragstrip information they tried to edit.

- **Use-Case Name:** Login

● Initial Assumption: The administrator has a registered account to login into the system. The account is saved in the database.

● Normal: The administrator will enter in a username and password to login.

● What Can Go Wrong: The administrator login credentials are not accepted as the username or password do not match the one saved in the database.

● Other Activities: The administrator can reset the password using the forgot password link

● System State on Completion: The administrator is logged in. They can view their dashboard.

- **Use-Case Name:** Edit Profile

● Initial Assumption: The administrator logins and is able to edit user profiles on behalf of the users.

● Normal: The administrator can edit profile information of the user such as editing their access level such as having administrator privileges for a dragstrip venue.

● What Can Go Wrong: The administrator is not able to find the user they searched for.

● Other Activities: The administrator will receive an error message from the system stating that the user cannot be found.

● System State on Completion: The administrator can search for the user again.

5.9.2. Drag Strip: Piedmont Dragway (Nick Brandsma)

- Use-Case Name: Log In

● Initial Assumption: The user will enter the correct information and will be allowed access to their account

● Normal: The user will gain access to their account after logging in

● What Can Go Wrong: The user enters the wrong information

● Other Activities: The user will be met with an error message

● System State on Completion: The user will be logged into their account

- Use-Case Name: Edit Dragstip information

● Initial Assumption: The user will add their location and its information and it will then be discoverable by the normal users of the application

● Normal: The drag strip is added and then is discoverable by the other users

● What Can Go Wrong: The address/location of the dragstrip cannot be found or verified

● Other Activities: The dragstrip account is met with a error that the location cannot be found

● System State on Completion: A new dragstrip location is added and is now discoverable by the users

- Use-Case Name: Send event notifications

● Initial Assumption: The dragstrip account can send a mass notification to users in a certain area range notifying them of an event

● Normal: Users within the specified range of the drag strip get a notification of the event

● What Can Go Wrong: The server fails to be able to send out the notification

● Other Activities: The dragstrip account is met with a error the the notification was not sent out and should try again later

● System State on Completion: The notification is sent out notifying nearby users of the event

- Use-Case Name: Add dragstrip admins

● Initial Assumption: The user will choose another account they wish to be linked with the dragstrip account to have access to the admin privileges

● Normal: The other account will now have admin privileges

● What Can Go Wrong: The account they are looking for is not able to be found

● Other Activities: The dragstrip account is met with a error that the account cannot be found

● System State on Completion: A new account is now added that is able to have the same privileges as the drag strip

5.9.3. Actor: Actor Name (Priyankaben Shah)

- **Use-Case Name:** Login

    ● Initial Assumption: The administrator has a registered account to login into the system. The account is saved in the database.

    ● Normal: The administrator will enter in a username and password to login.

    ● What Can Go Wrong: The administrator login credentials are not accepted as the username or password do not match the one saved in the database.

    ● Other Activities: The administrator can reset the password using the forgot password link

    ● System State on Completion: The administrator is logged in. They can view their dashboard.

- **Use-Case Name:** Edit Profile

    ● Initial Assumption: The administrator logins and is able to edit user profiles on behalf of the users.

    ● Normal: The administrator can edit profile information of the user such as editing their access level such as having administrator privileges for a dragstrip venue.

    ● What Can Go Wrong: The administrator is not able to find the user they searched for.

● Other Activities: The administrator will receive an error message from the system stating that the user cannot be found.

● System State on Completion: The administrator can search for the user again.

- **Use-Case Name:** Update Preferred Location

● Initial Assumption: In profile settings, users can choose their preferred location to connect in an area.

● Normal: Users can choose location by city and state.

● What Can Go Wrong: This data will be saved per user but showing nearby users on a map could lead to errors.

● Other Activities: When the location section does not show nearby users as it doesn't sync all users.

● System State on Completion: User can add location and look for others nearby.

- **Use-Case Name:** Create Profile

● Initial Assumption: Users can create a profile by signing up.

● Normal: User creates a profile after verifying email and creating username, password and verifying birthdate.

● What Can Go Wrong: During this time when a new profile is created, data could not be saved for next login or does not check for a valid email address.

● Other Activities: The age limit needs to be verified as this app is for 18+ users.

● System State on Completion:Users will be able to successfully create a profile and log back in.

- **Use-Case Name:** Interact with users

● Initial Assumption: Interact with other users in the application.

● Normal: Users can find other car enthusiasts via searching for other users, cars or nearby users and chat with them via live chat.

● What Can Go Wrong: The search form does not check through the database properly leading to no users found per criteria.

● Other Activities: A search form will be created on top for easy searchability.

● System State on Completion: Users will be able to find other users and chat with them.

- **Use-Case Name:** View dragstrip locations

● Initial Assumption: Dragstrip venues create their profile for users to find nearby locations.

● Normal: Dragstrip profiles also have their location tag on for users to find their location.

● What Can Go Wrong: Location does not sync and prevents nearby users to locate them.

● Other Activities: Users can check their stats/information on location/

● System State on Completion: Dragstrip accounts can show their location to nearby users.

6. Design Documents

6.1. Software Architecture

6.2. High-Level Database Schema

6.3. Software Design

6.3.1. State Machine Diagram: Actor Name (Responsible Team Member)

6.3.2. State Machine Diagram: Actor Name (Responsible Team Member)

6.3.3. State Machine Diagram: Actor Name (Responsible Team Member)

6.4. UML Class Diagram

7. Scenario