

# *Dan's Car Barn*

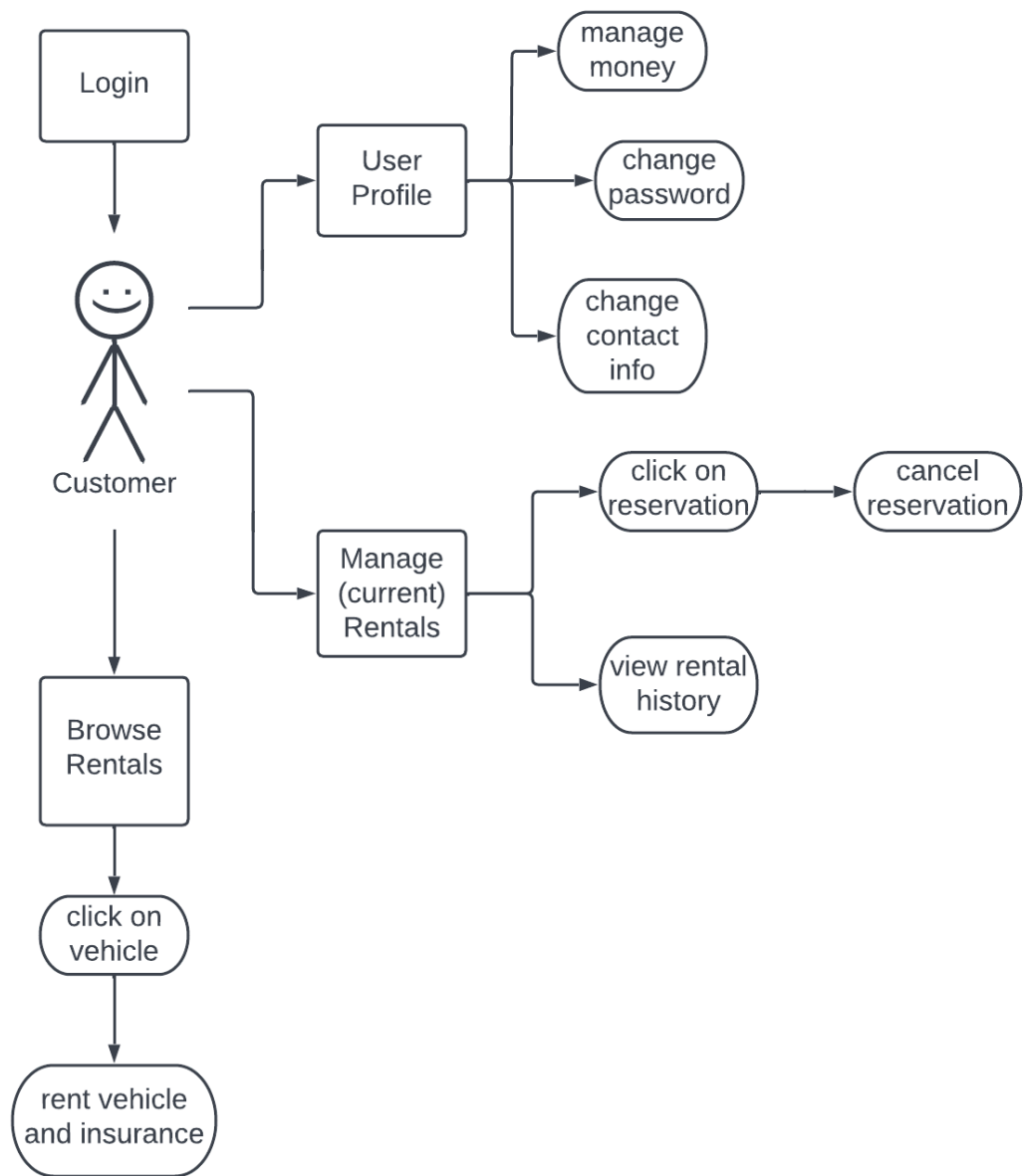
## **1. Introduction and Context**

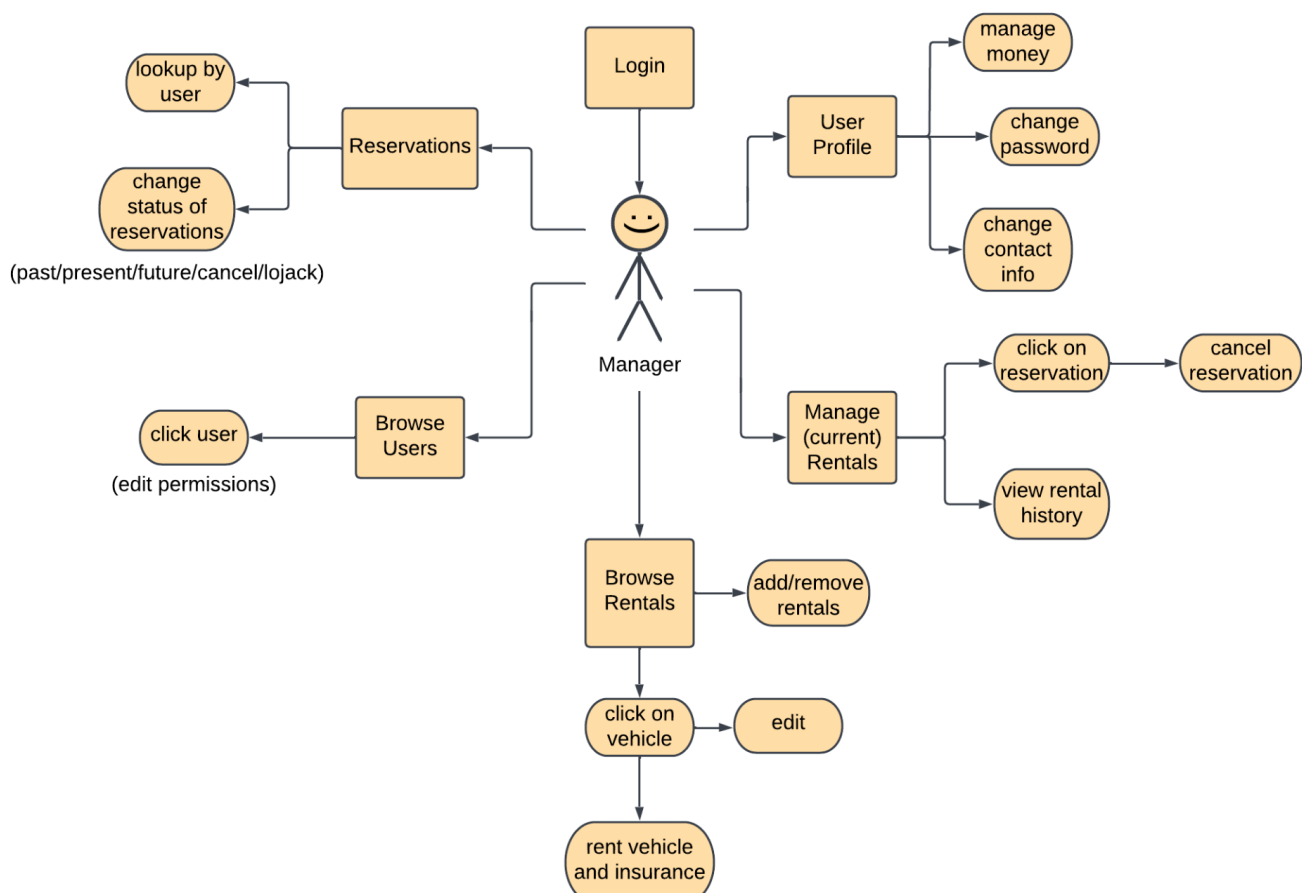
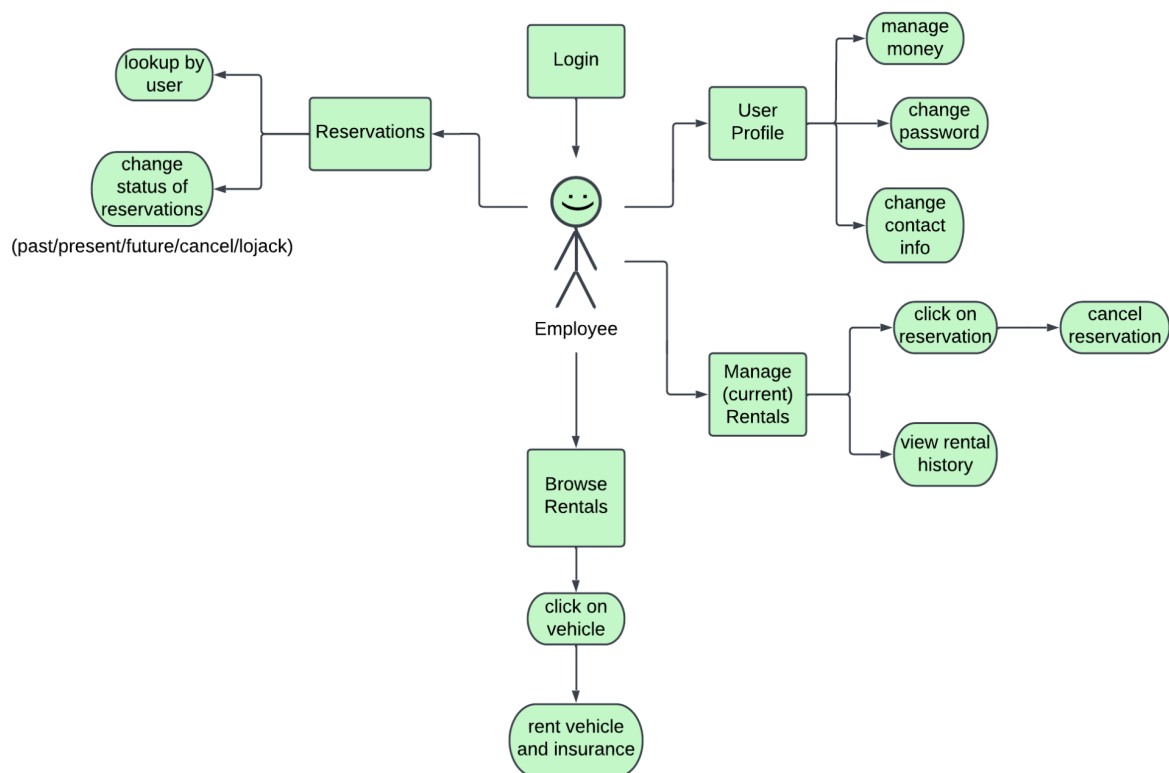
Renting a car can be a confusing process. Customers have specific requirements that need to be met and it can oftentimes be difficult to find a vehicle that meets the necessary requirements, and if you are going to a car rental place, you probably don't have a vehicle with which to drive to another car rental place. In addition to this, it is difficult to know how much your rental is actually going to cost until you are actually at the kiosk about to swipe your card. That's when all of the hidden fees and the actual rental price comes out into the open. It's important to customers to have that kind of transparency and the ability to plan ahead for their rentals, and it's important for rental companies to have all of the information necessary to be able to prepare for the rental.

Dan's car barn is a web application aimed at streamlining the process of renting cars. The process of renting generally requires one to go in person and work with the person there. Dan's car barn is a fully online rental service, with rental planning arrangements and easy management tools for current rental arrangements for employees, managers, and users. The general use case will be to browse the website for available rentals, select a time for the rental to take place, and reserve the rental. All that's left is to go in person to the rental service and pick up your car, at which point the rental service will have an interface allowing the manager or employee to change the status of the rental to indicate that it is currently in use. Once the rental period is over, the customer may return the car and the employee can mark the arrangements as finished.

This document details out the use cases, user stories, and functional and nonfunctional requirements. This document will be used as a roadmap for development cycles and will be important in keeping the Dan's car barn development team on track.

## **2. Users and their Goals**





| Users    | Goals  |
|----------|--|
| User     | n/a (placeholder for base user)  |
| Customer | Rent a car; Cancel a reservation for a car; Browse vehicles available to rent; pay for rentals   |
| Employee | Browse current, previous, and future reservations; Change status of reservations, this includes low-jacking; Lookup reservations by username/other info; able to rent out car; browse vehicles available to rent; manage and view their own reservations, past, present, and future; |
| Manager  | Same as employee with additions: being able to manage who can do what (permissions of a given user); change what vehicles can be rented; edit details about a vehicle to be rented, such as price, description, etc;   |

#### User Stories:

**General Use Case:** User is in need of a vehicle. They log on to the website, put in the date range for which they need the vehicle, and several available ones come up on the screen. They select one, view its details, go back and select another. They like the second one and click the reserve button found on the page. They are then brought to a checkout page with all of the fees associated with reserving the rental. They click the checkout button, and the rental is marked as reserved in this user's name. On the appointed day, they go to the rental place and the manager marks the reservation as active. At the end of the period, the user returns the car to the rental place and the manager marks the reservation fulfilled.

**Customer renting a vehicle but doesn't have enough money in account:** The customer logs into the application by inputting their username and password. They are then redirected to the browse rentals page. They click on a vehicle which pops up a modal with information on the vehicle and buttons to rent the vehicle. The user clicks the "rent vehicle" button and is redirected to a checkout page where they can select to add insurance, view their total, and a button that says checkout. Upon clicking checkout the user is blocked and notified that they have insufficient funds. There is a button to redirect the user back to the home page. The user clicks it and is redirected to the home page. They then navigate to their user profile where they select to manage account balance. This takes them to a page where they can choose to add or remove money from their account. They select to add money and input the amount they want to put in. Upon completion they navigate back to the page where they select the vehicle they want again and are directed down the same pages. From there when they select checkout they are greeted with a success page and details about their reservation. Details for their reservation are added and can be viewed by them from their "My reservations" page. The user proceeds to logout as they are satisfied with their experience.

**Manager Adds Car:** The business has a new vehicle that they want to display on Dan's car barn available for rental. They open up the admin panel and select the new vehicle button. A page opens up in which the manager then inputs the information and attributes associated with the new vehicle, and submits the vehicle. The vehicle now displays on the users' end as available for rental.

### 3. Functional Requirements

Functional requirements for this project include:

1. User Authentication
  - 1.1. User should enter user information: Username and Password
    - 1.1.1. If the credentials are invalid, the user should not be allowed in the protected part of the application, and should be informed that the credentials are invalid
    - 1.1.2. If the credentials are valid then a login session should be started and the user should be redirected to the home page.
2. Browse Available Rentals
  - 2.1. The system will show all active rental vehicles on the browse rental page.
  - 2.2. The user will be able to view a page with detailed information on each vehicle (see requirement 3)
  - 2.3. The user will be able to filter vehicles by their availability on this page.
    - 2.3.1. The user will be able to select a range of dates for which they desire to rent a vehicle
    - 2.3.2. The vehicle's price will be displayed based on how long the user desires to rent them for. It will display without insurance fees..
3. View individual Rentals
  - 3.1. The user will be able to see all days for which the rental vehicle is available.
    - 3.1.1. If the user tries to select a time period for which the vehicle is already reserved, they will not be able to do so.
    - 3.1.2. The user will be able to change their desired rental period to any period during with the rental is not already reserved.
    - 3.1.3. The price will change based on the number of days the rental is desired.
  - 3.2. The user will be able to view more details and a description of the car on this page
    - 3.2.1. If the vehicle is no longer being used it will display a message on the page letting the user know that the vehicle is currently no longer being used
4. Account management
  - 4.1. The user will be able to view previous rentals
    - 4.1.1. The user may click on the vehicle and be taken to the page listed in requirement 3.
  - 4.2. The user will be able to change their password
  - 4.3. The user will be able to add money to their account balance.
    - 4.3.1. The user will be prevented from putting in values that allow the account balance to go negative.
5. Manage vehicles
  - 5.1. Add or remove vehicles
    - 5.1.1. A vehicle which is removed will not actually be removed, but rather marked as inactive.

- 5.2. Change vehicle status
  - 5.2.1. Employee will be able to select vehicle status from: in use, reserved, low-jacked, able to be reserved.
- 5.3. Change vehicle information
  - 5.3.1. Manager will be able to change the information associated with a vehicle such as price, description, etc.
  - 5.3.2. Manager will be able to change which vehicles can and cannot be rented.
- 6. Manage Users/User Permissions
  - 6.1. Manager will be able to change user permissions; able to select from Customer, Employee, and Manager permissions
  - 6.2. The manager will be able to create a new account for a user.
- 7. Reservations
  - 7.1. Users are able to view their past, current, and future reservations.
    - 7.1.1. Employees are able to view all past, present, and current reservations
      - 7.1.1.1. Employees can search/filter by username
    - 7.1.2. Users are able to cancel any future reservations, leading to a refund on their account balance.
      - 7.1.2.1. Current and past reservation statuses cannot be changed by the user.

#### **4. Non-functional Requirements**

Non-functional requirements for this project include:

- All code that is submitted to the repository must be reviewed by at least one other member of the group before it can be officially merged with the master branch.
- Weekly in-person meetings will be held at a minimum of once per week. Meeting frequency will be increased as needed.
- When regular class time is not being held, we will have an in-person meeting in its place.

#### **5. Future Features**

Future features for this project include:

- The ability for customers to leave ratings and reviews on the cars they've rented.
- The option for customers to apply filters as they browse the site's available rentals.
- A "reschedule" option that allows customers to quickly change their reservation dates.
- The ability for employees to reserve cars for another customer.
- The ability for employees to change user data other than permissions.

#### **6. Glossary**

Important terms for this project include:

- *Low-jacking*: Employees purposely causing a car to break down if the customer who rents it doesn't purchase insurance.
- *Reservation*: The period of time (chosen in advance) in which a customer is allowed to rent a car.
- *Rent*: The action of temporarily being allowed to drive a car, provided that the customer pays a certain fee per day.