RentaCar

Matthew Stauder

INFO-C451: System Implementation

4 February 2022

TABLE OF CONTENTS

SECTION 1

Customer Problem Statement

Travellers often struggle when it comes to renting a car of his or her choice. Due to the nature of the car rental industry, there may be a limited supply of cars available for rental at a given location. In addition, some locations may be experience more traffic than others, which may lead to extensive lines, or waiting periods until an individual gets their car. People want to feel comfortable and safe when travelling, which is why it is important to provide these individuals with the necessary information needed to enhance their travel experience.

This system application will focus on fulfilling individual's desires to rent a car at airports. The customer can view all available vehicles at any car location, filter vehicles based on car type (make, model, year), reserve vehicles online, set drop-off times to return vehicles, rent or reserve more than one vehicle, and view previous rental history. This application will help customers avoid long lines at rental car companies after flights. This system will minimize waiting times by allowing customers to reserve and set a pickup and drop-off time ahead of time; customers have the ability to cancel, or update reservation and drop-off times through the application. In addition, this application provides users with a detailed history log of previous car rentals. Providing users with records of previous car rentals may help individuals remember, modify, or confirm past experiences.

This system will also benefit car rental employees because it should eliminate many redundant tasks such as adding and updating customer information as well as car information. By allowing customers to enter in their own information, employees will be able to focus on more important tasks at hand such as customer service.

A username and password will be required of users in order to gain access to the system application. This means that both customers and employees will have usernames and passwords. Employee accounts will differ from customer accounts such that the employee accounts will have administrative controls. Users are responsible for creating a unique password for their account. Once a customer creates an account, he or she may begin entering vehicle information, vehicle preferences, and rental favorites.

In summary, travellers need a system application that allows them to view the availability of various types of cars at a car rental location, reserve a car, rent a car, set a drop-off time to return the car, and view previous car rental history. Customers will also have the ability to filter and favorite cars to optimize their search. This application will allow employees to focus on tasks of higher importance rather than dealing with redundant tasks/processes. Overall, this application benefit both customers and employees, and ultimately, will save both parties a lot of time.

Glossary of Terms

Pickup Time – Customers are able to reserve a time to pickup his or her vehicle from the rental car company. Pickup times may be updated, or cancelled at any time.

Drop-off Time – Customers may designate a time to drop-off his or her vehicle at the rental car company. Drop-off times may be changed at any time through the application.

Vehicle Preference – Customers have the ability to enter their personal preferences about vehicles. Vehicle preferences may include: color of vehicle, number of seats in a vehicle, the vehicle manufacturer, the vehicle model, and the year of the vehicle. Preferences about vehicles are saved to a customer's account.

Rental Favorites – Customers may save, or mark certain vehicles as favorites. Rental favorites will appear on the user dashboard and will provide customers with a quicker way to access and rent vehicles.

Username – Users will use a username to log in to the application. Usernames are required to create and access an account.

Password – Users will use a unique password to log in to the application and access their own personal information. Each user will be prompted to log in using a unique password. New customers, or first-time users will be required to create a new password to complete their account setup.

Administrative Controls – Staff members at the rental car company have the ability to access customer accounts. Customers must provide employees with consent to allow them to access their account. Administrative controls will only be used to resolve issues with customers.

System Requirements

Functional Requirements

No.	Priority Weight	Description
REQ-1: Reserve a Car	High	This will allow customers to reserve a car online for rental.
REQ-2: Update Information	High	Users will be able to update information in the application system.
REQ-3: Update Pickup/Drop-off Time	High	This will allow customers to modify the pickup and drop-off times of rental cars.
REQ-4: Cancel Reservation	High	This will allow users, both customers and employees, to cancel car rental reservations.
REQ-5: Access History Log	High	Users should have the ability to view previous rental history.
REQ-6: Vehicle Search	High	Users will be able to search for vehicles at

		various car rental locations.
REQ-7: Branch Location Search	High	This will allow users to search for car rental companies at a particular location.
REQ-8: Customer Search	High	Employees will be able to look customers up in the system.
REQ-9: Log In	High	Users will utilize their username and password to access the system application.
REQ-10: Vehicle Preference	Medium	This will allow customers to enter their vehicle preferences.
REQ-11: Rental Favorites	Medium	This will allow customers to save certain vehicles for future use, or reference.

Nonfunctional Requirements

1. **Functionality –** the system application will be able to run and store all functions of the system application.

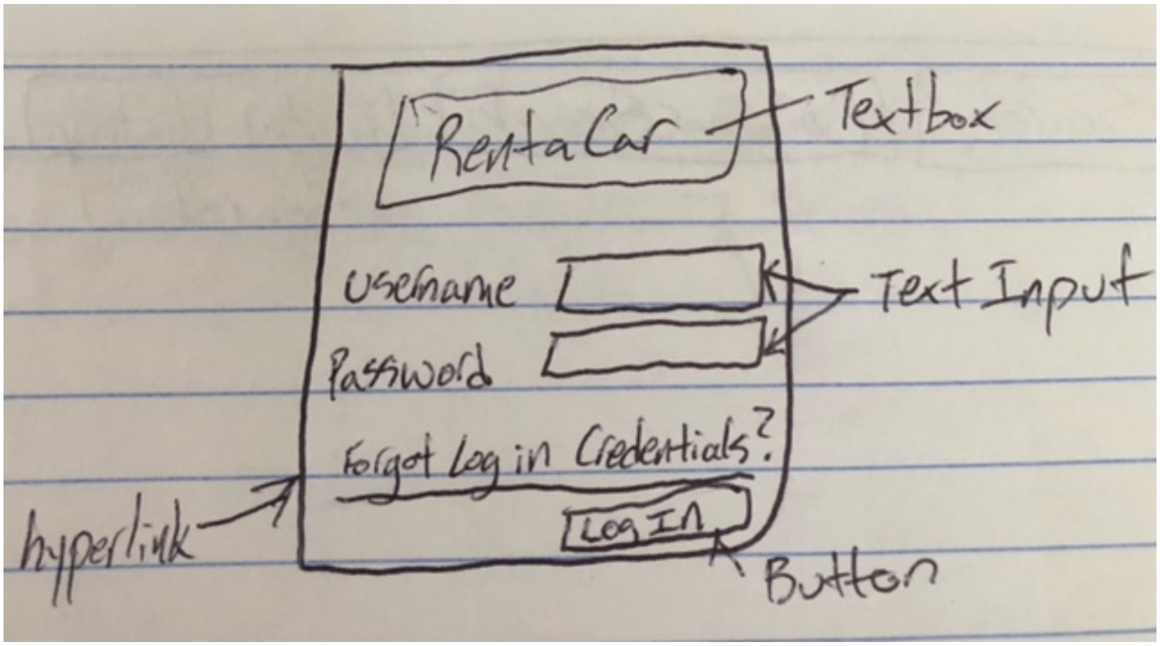
2. **Supportability –** customer support is provided to users 24/7. In-person customer support is available to users during regular office hours; however, users may receive support via phone, email, or through the system application.
3. **Reliability –** the system will be available and function properly for users regardless of the time of day, or the amount of traffic in the system.
4. **Usability –** users will be able to complete tasks efficiently and navigate the application without difficulty.
5. **Performance –** the speed in which the system will process instructions and functions of the program.

User Interface Requirements

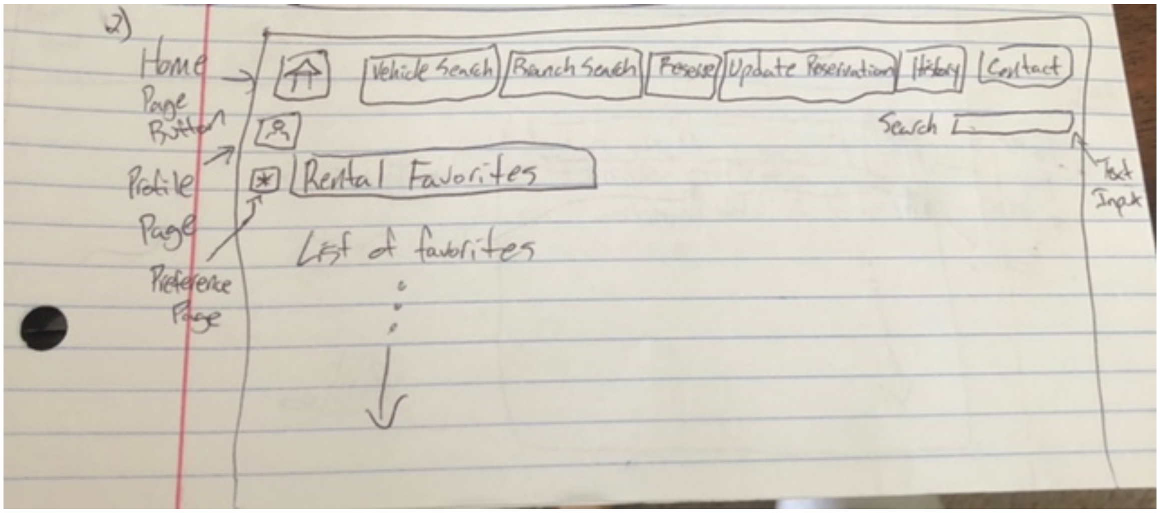
Users will use their username and password to log in to the application. Once logged in, users will see a home page that will include a menu bar at the top of the screen. In addition, rental favorites may appear on a customer's home page. A search bar will also be present to allow users to search for particular information. Users will utilize the menu and search bars to navigate the application. (NOTE: Customer Menu Bar is short for the menu bar tabs in the Customer view, and similarly, Employee Menu Bar is short for the menu bar tabs in the Employee view.)

1. **Log In Screen –** This screen will appear for users attempting to access the system application. Users will be asked to enter their username and password to enter the system. A "Forgot Log In Credentials" button will be

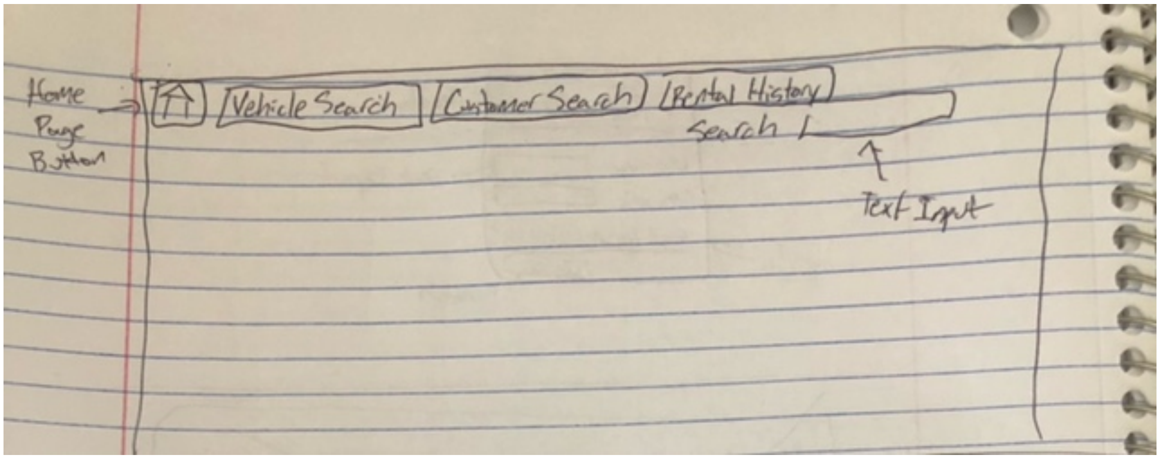
present below for users who are unable to remember either their username, or password.



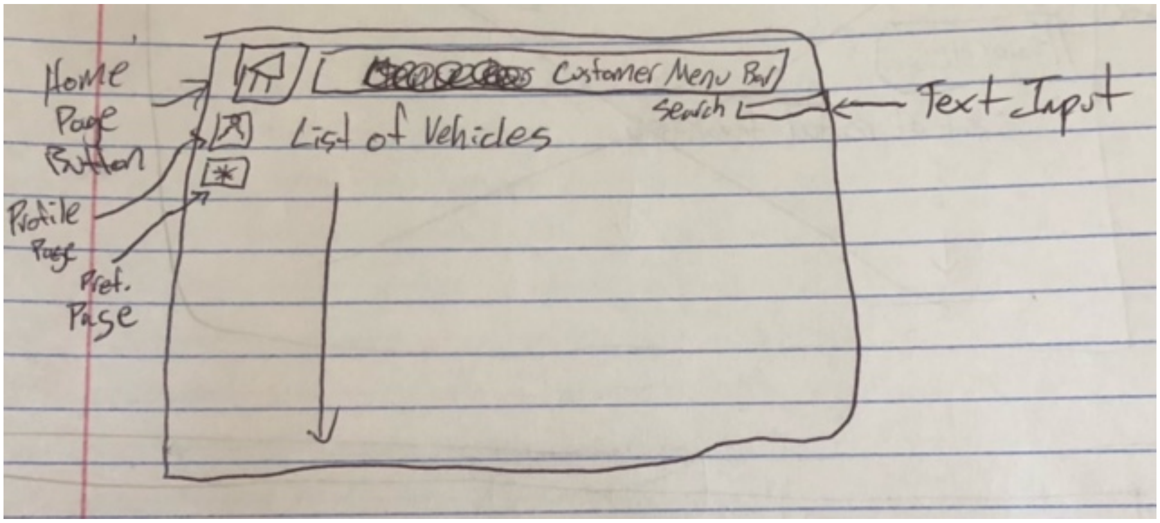
2. **Customer Home Page –** This will be the home page view for customers. Customers will see their rental favorites as well as a menu bar at the top of their screen. The menu bar contains tabs such as: Vehicle Search, Branch Search, Reserve, Update Reservation, Rental History, and Contact.



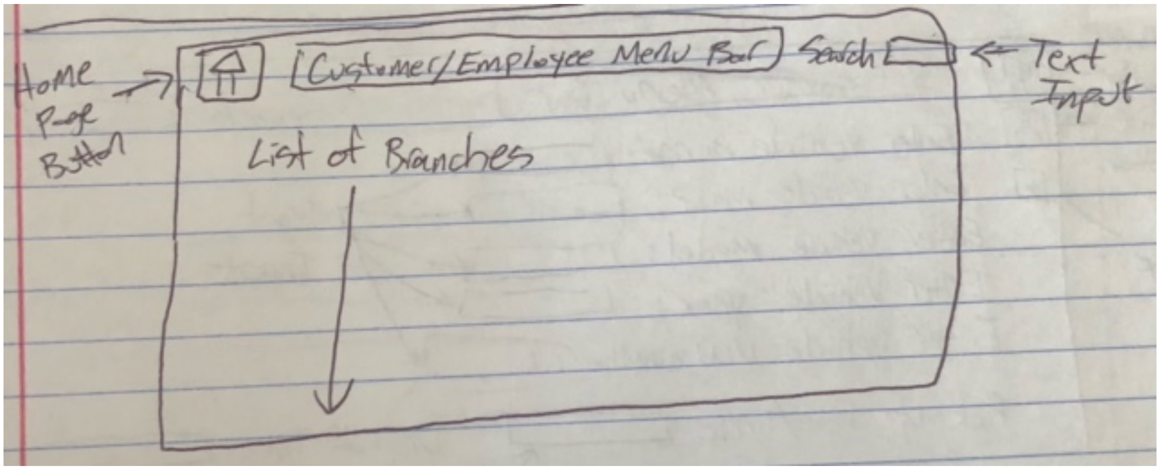
3. **Employee Home Page –** This will be the home page view for staff members. Employees will be able to search for customers on this home page and view previous rental records. When an employee searches a customer, he or she will be redirected to the *Customer Page.* When an employee clicks on the "Rental History" tab in the menu bar, he or she will be redirected to the *Rental History Page.*



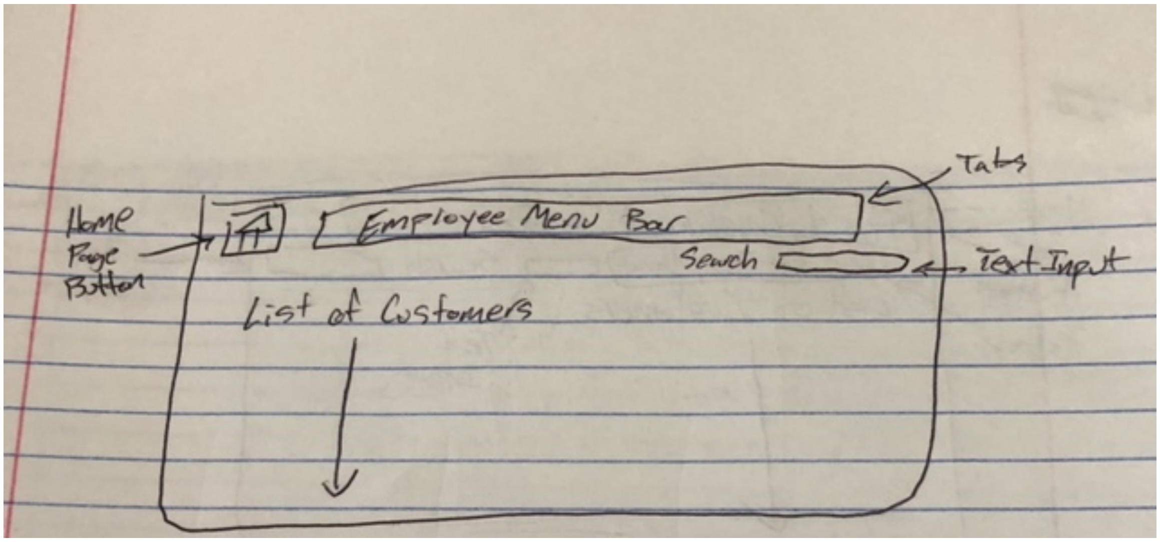
4. **Vehicle Page –** This page will appear after a user conducts a *Vehicle Search*. A list of available vehicles will appear on this screen. If a user clicks on a vehicle, he or she will be redirected to the *Vehicle Information Page.*



5. **Branch Page –** This page will appear after a user conducts a *Branch Search*. A list of operating branches will be shown.

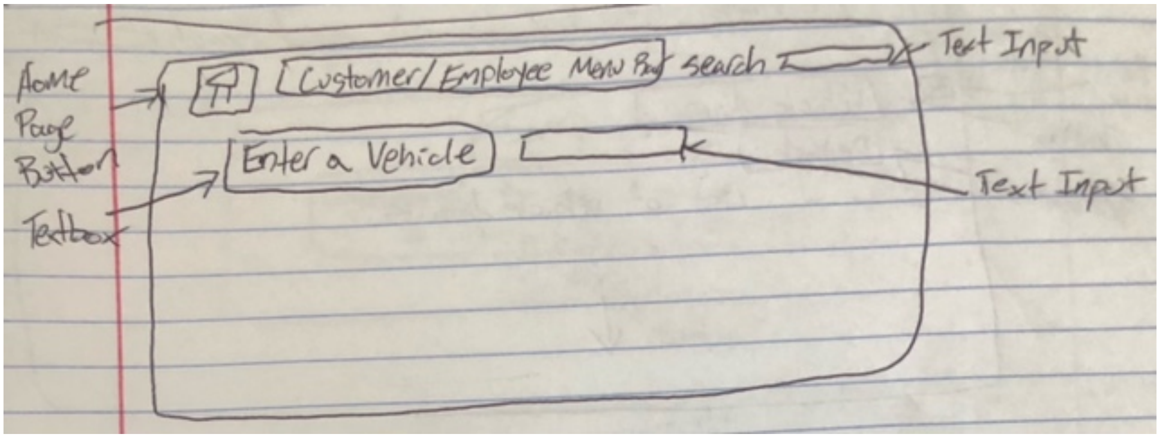


6. **Customer Page –** This page will appear after an employee performs a *Customer Search.* A list of customers will be displayed on the screen. When an employee clicks on a customer, he or she will be redirected to the *Customer Information Page.*

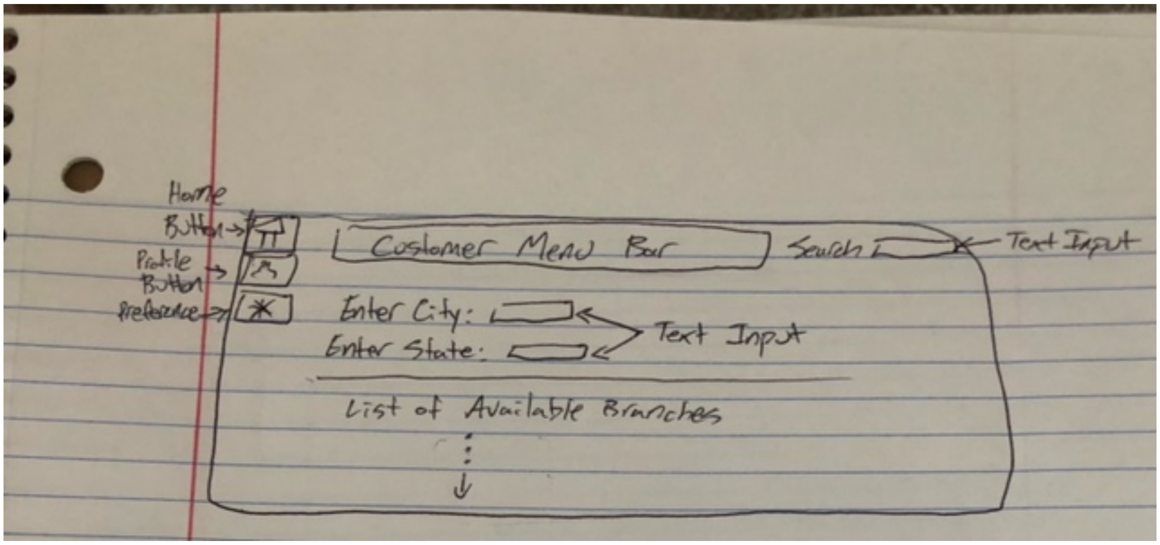


7. **Vehicle Search –** This will be the page that users will see when they click on the "Vehicle Search" tab. This screen will allow users to search for a vehicle

using the search bar. Users will be redirected to the *Vehicle Page*, where a list of vehicles will be displayed based on the search.

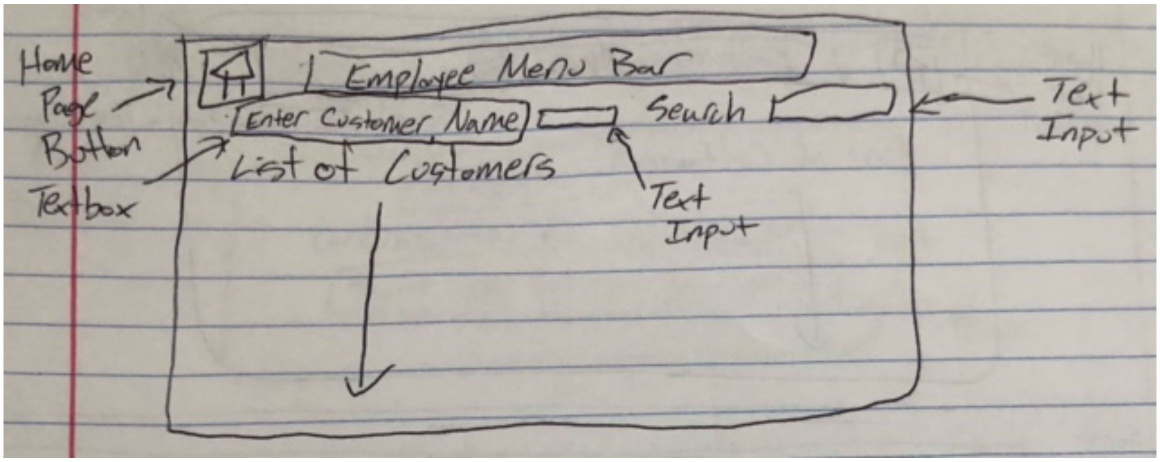


8. **Branch Search –** This will be the page that users will see when they click on the “Branch Search” tab. This screen will provide users with the opportunity to search branch locations by location (city, or state). Users will be redirected to the *Branch Page*, where a list of branches will be displayed based on the searched location.

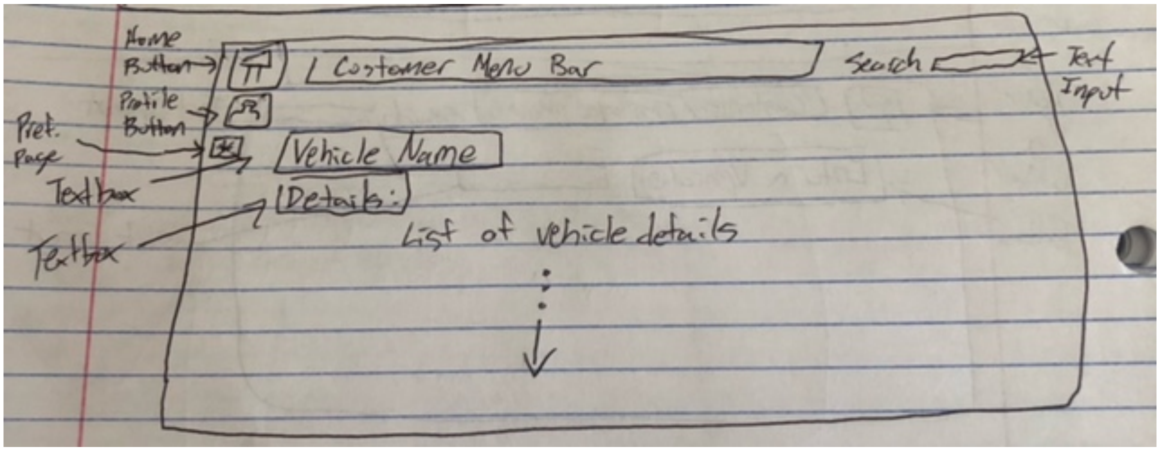


9. **Customer Search –** This will be the page that employees will see when they click on the “Customer Search” tab. This screen will allow employees to

search for customers in the system's database. Upon conducting the search, employees will be redirected to the *Customer Information Page,* where a list of customers will be shown based on the search.

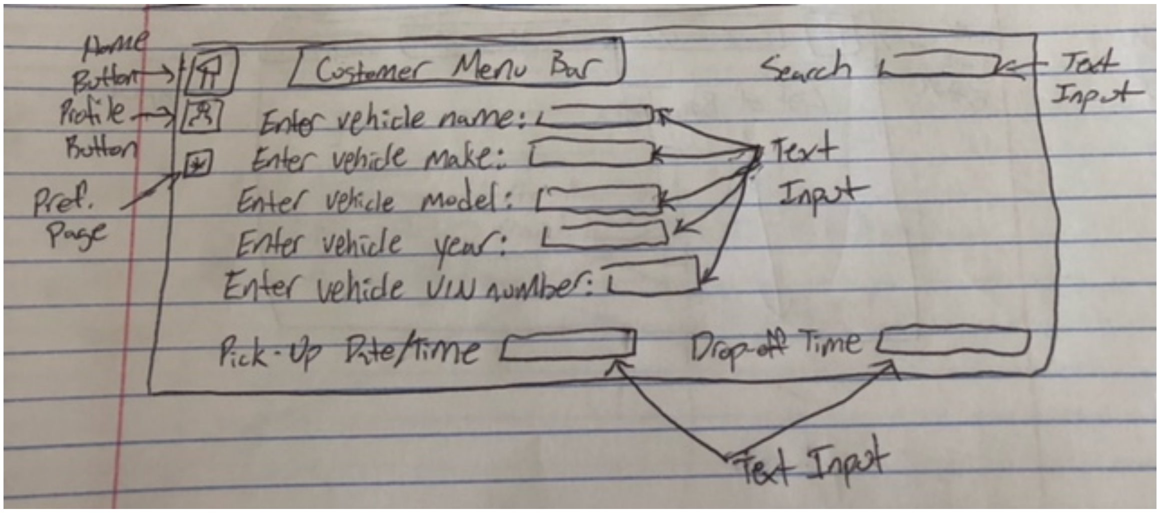


10. **Vehicle Information Page –** This page will provide details about individual vehicles such as the color, number of seats, make, model, and year of the car.

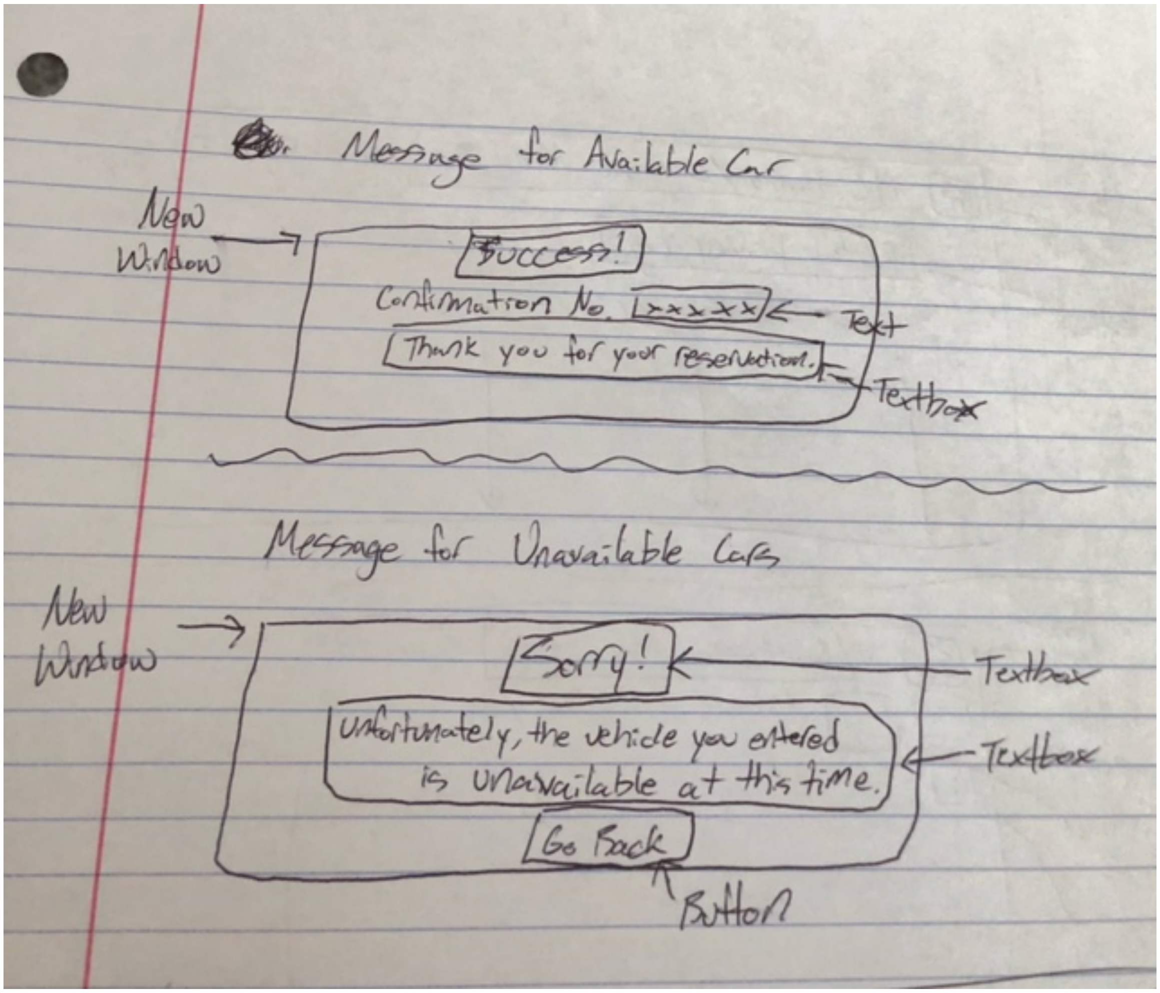


11. **Reservation Page –** This page will allow customers to reserve a rental car. Customers will set pickup and drop-off times on this page as well. In addition, customers will manually enter vehicle information such as the color, make, model, and year of the car. If the car is available, a message will appear

confirming the customer's reservation. If the car is unavailable, a message will appear stating, "Vehicle is currently unavailable".



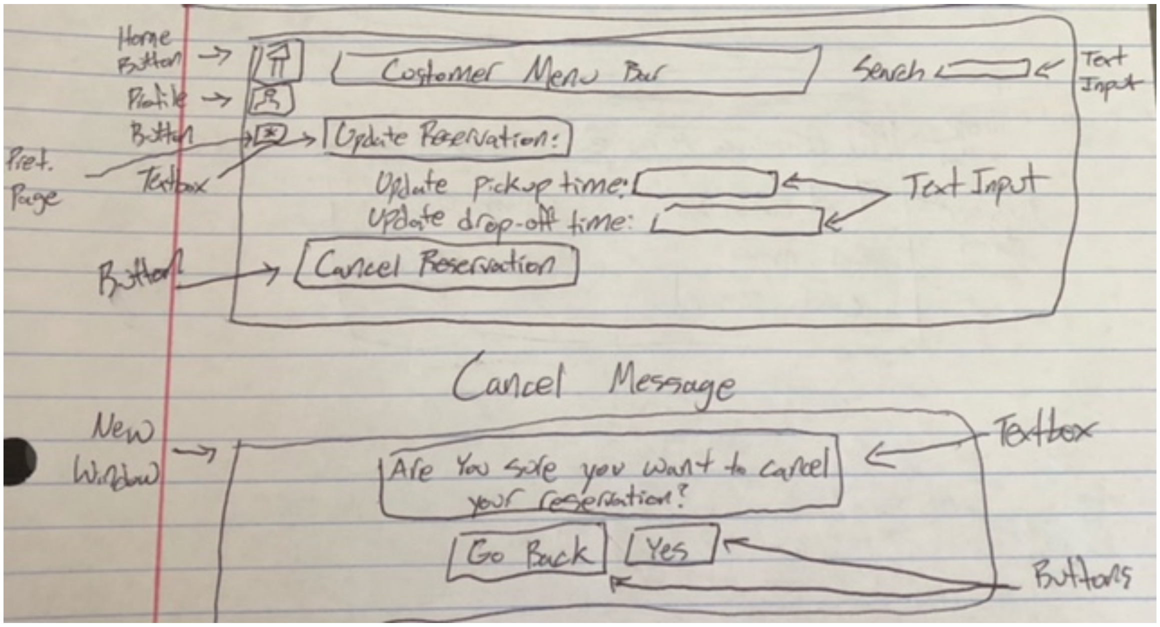
RESERVATION MESSAGES:



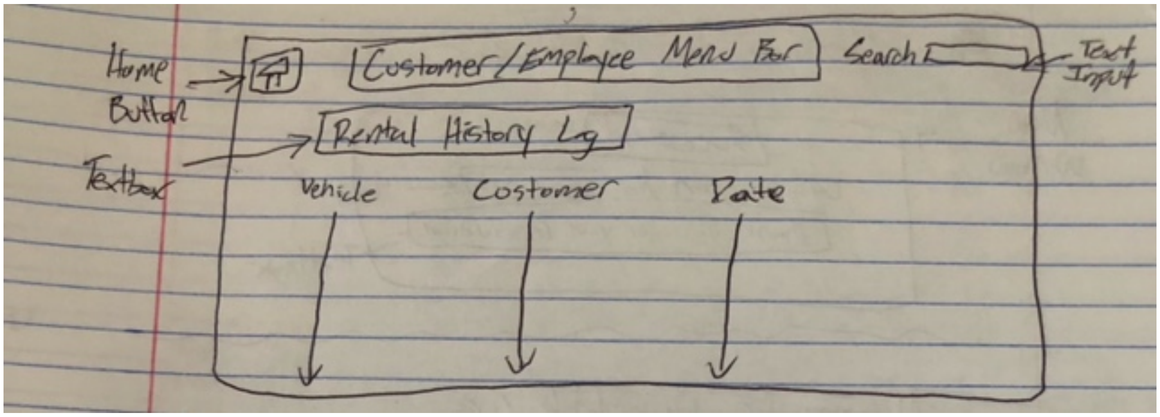
Thank you for your reservation.

Unfortunately, the vehicle you entered is unavailable at this time.

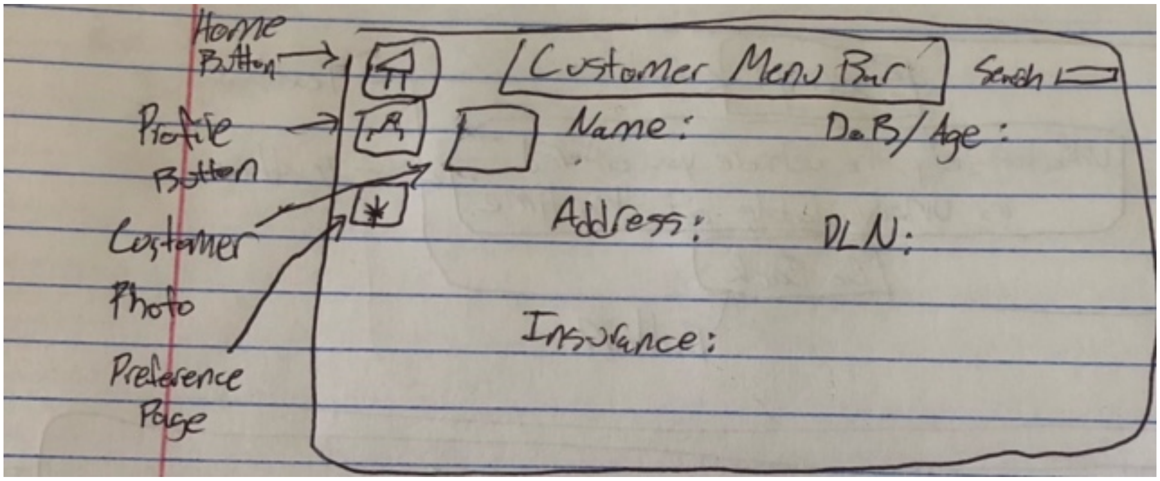
12. **Update Page –** This page will allow customers to update their reservation. Customers will be able to modify the both the pickup and drop-off times of the rental car. In addition, customers may cancel their reservations on this page. When a customer cancels his or her reservation, he or she will receive a message confirming whether or not the individual wants to proceed with cancelling their reservation.



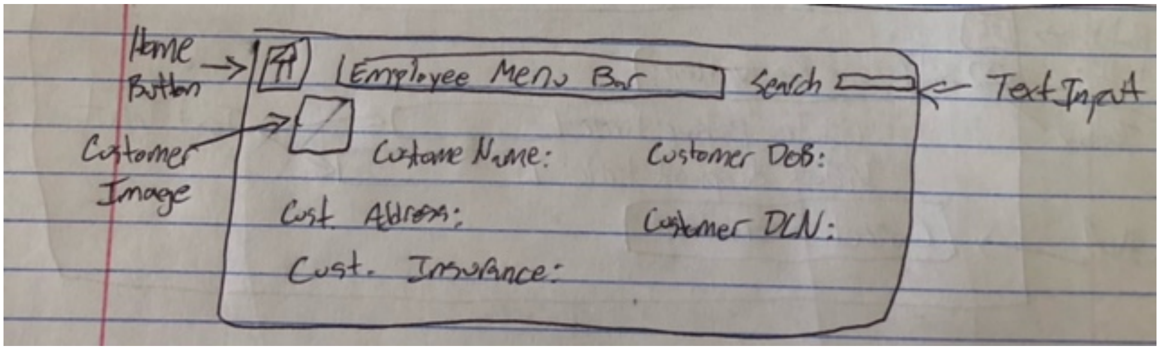
13. **Rental History Page –** This page will be available for all users. Both customers and employees will have the ability to access previous rental records. The rental history page will provide details relevant to particular customers and/or vehicles.



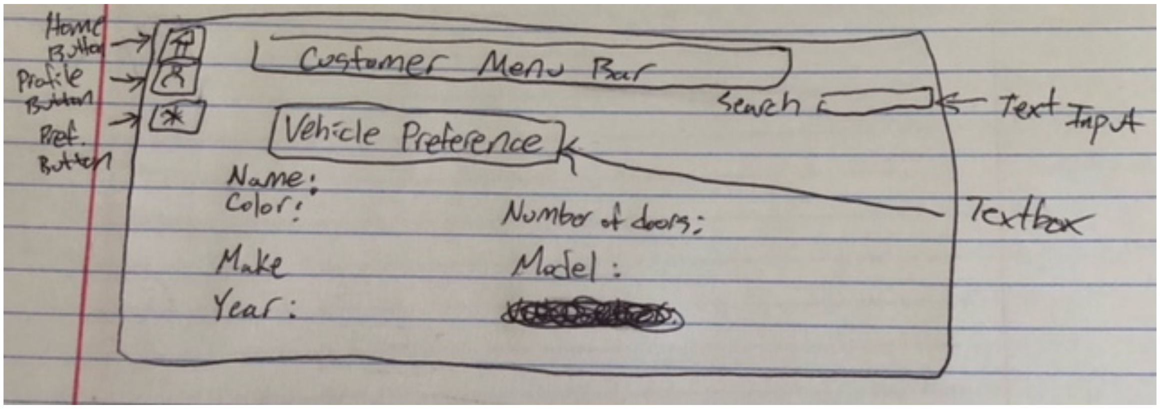
14. **Customer Profile Page –** This is the page where customers are able to manually enter personal information such as their full name, date of birth/age, address, driver's license number, and car insurance.



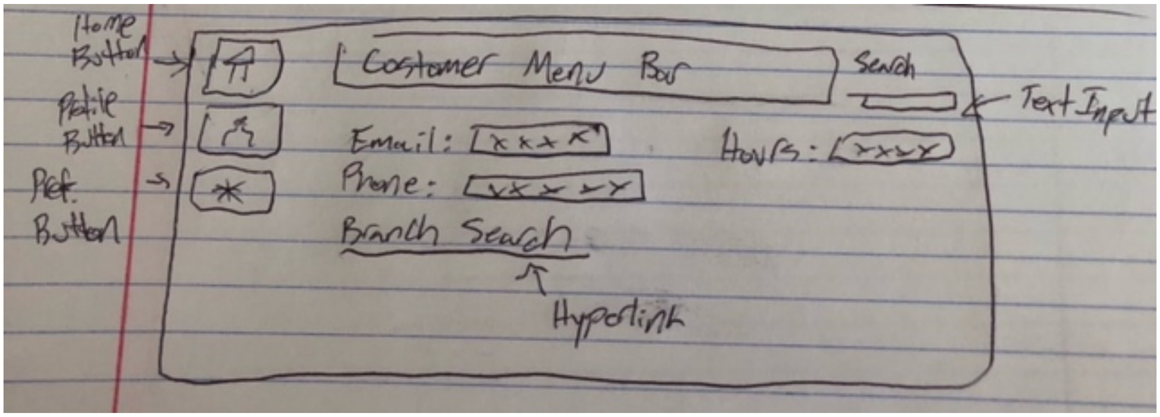
15. **Customer Information Page** - This page will consist of personal information about customers such as their full name, date of birth/age, and address. Employees will be the only ones able to access the Customer Information Page. (The information on this page should match the information on the *Customer Profile Page.)*



16. **Vehicle Preference Page –** This page will store a customer's preference(s) for vehicles. Preferences may include: color, number of doors, make, model, and year of the car. Vehicle preferences are optional, but help the system to filter best results based on the specified criteria.



17. **Contact Page –** This page will display methods of contact such as an email address, phone number, and a hyperlink to the *Branch Search.* The purpose of this page is to provide additional information and support to users.



SECTION 2

Functional Requirement Specification

Stakeholders: A list of stakeholders for this system application include:

- Matthew Stauder (Owner of RentaCar)
- Customers (Travellers)
- Sales Staff
- Car Fleet Manager
- Customer Support Staff
- Rental Car Company Executives

Actors and Goals: The chart below provides details regarding the roles of the various entities that will directly interact with this system application.

Actors	Goals
Staff (Administrator)	• Log into system • Log out of system • Look up Log in credentials • Add a new customer • Search for customer(s) • View customer information • Update customer information

	• View car information • Update car information • Update branch information • Update contact information • Add a new car • View car rental history • View available cars • View unavailable cars • View reserved cars • View customer reservations • Schedule a customer reservation • Update a customer reservation • Cancel a customer reservation • Reschedule customer pickup time • Reschedule customer drop-off time
Customer (User)	• Log into system • Log out of system • Look up Log in credentials • Enter personal information • View personal information • Update personal information • Search for cars • Search for branch locations • View available cars • Reserve car(s) • View reservation • Update reservation • Cancel reservation • Set a pickup time • Set a drop-off time • View personal rental history • Filter vehicle searches
Database	• Store customer information • Store staff information • Store car information • Store branch information • Store reservation information

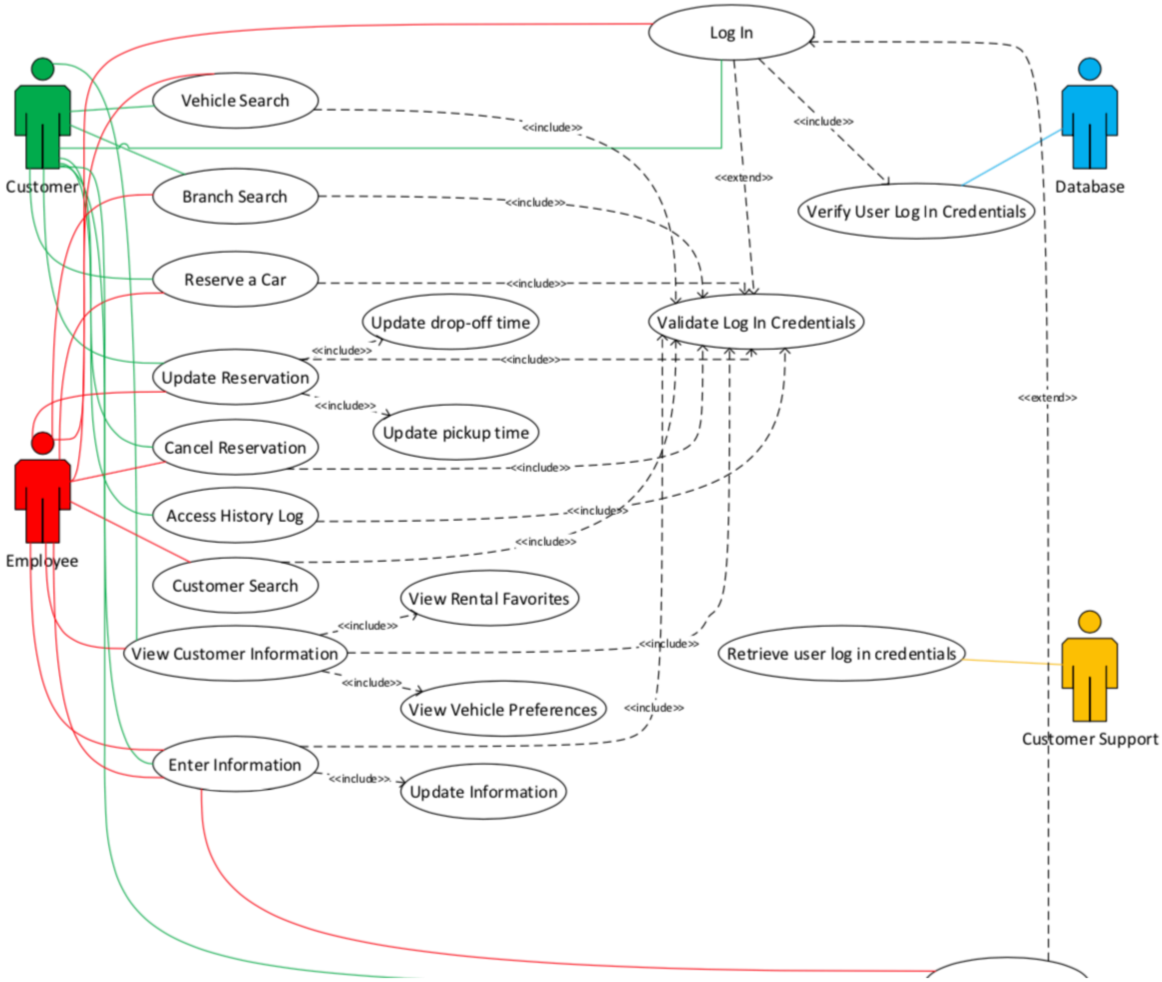
Use Cases:

Casual Description

System Requirement	Description
REQ-1: Log In	Users will utilize their username and password to access the system application.
REQ-2: Update Information	Users will be able to update information in the application system.
REQ-3: Vehicle Search	Users will be able to search for vehicles at various car rental locations.
REQ-4: Reserve a Car	This will allow users to reserve a car online for rental.
REQ-5: Update Reservation	Users will be able to modify the pickup and drop-off times of rental cars.
REQ-6: Cancel Reservation	This will allow users, both customers and employees, to cancel car rental reservations.
REQ-7: Access History Log	Users will have the ability to view previous rental history.
REQ-8: Customer Search	Staff members will be able to look customers up in the system.
REQ-9: Branch Location Search	This will allow users to search for car

	rental companies at a particular location.
REQ-10: Vehicle Preference	This will allow customers to enter their vehicle preferences.
REQ-11: Rental Favorites	This will allow customers to save certain vehicles for future use, or reference.
REQ-12: Log Out	This will allow users to log out, or exit the system application at any time.
REQ-13: Look Up Credentials	This will allow users to retrieve their log in credentials so that they are able to access the system application.

Use Case Diagram



Traceability Matrix

REQ-ID #	Requirement	Priority Level	Test Case ID	Test Case Description
1	Log In	HIGH	T01	Log in without a username
			T02	Log in without a password
			T03	Log in without a username and password
			T04	Log in as a customer
			T05	Log in as an employee
2	Update Information	HIGH	T06	Customer updates information
			T07	Employee

				updates information
3	Vehicle Search	HIGH	T08	Customer searches for vehicle
			T09	Employee searches for vehicle
4	Reserve a Car	HIGH	T10	Customer reserves a car
			T11	Employee reserves a car for a customer
5	Update Reservation	HIGH	T12	Customer updates pickup/drop-off time for a car reservation
			T13	Employee

				updates pickup/drop-off time for a customer’s car reservation
6	Cancel Reservation	HIGH	T14	Customer cancels a reservation
			T15	Employee cancels a reservation for a customer
7	Access Rental History Log	HIGH	T16	Customer accesses personal rental history log
			T17	Employee accesses a customer’s

				rental history log
8	Customer Search	HIGH	T18	Employee searches for a customer
9	Branch Location Search	HIGH	T19	Customer searches for car rental branches in a particular location
10	Vehicle Preference	MEDIUM	T20	Customer enters vehicle preferences
11	Rental Favorites	MEDIUM	T21	Customer selects cars as favorites
12	Log Out	HIGH	T22	Customer clicks Log Out button
			T23	Employee

				clicks Log Out button
13	Look Up Log In Credentials	HIGH	T24	Customer retrieves Log In Credentials
			T25	Employee retrieves Log In Credentials

Fully-Dressed Description

REQ-1 Log In: This requirement allows users to access the system application using their username and password. The system application will be able to differentiate customers and employees based off of the username entered at log in. It is important to understand that the views of customers and employees differ, which is why it is important that the correct user is logged into the system application. Users who enter their log in credentials correctly will receive access to the application, whereas users with invalid log in credentials will receive an error message that states, “Sorry, invalid username and/or password. Please try again.” If a user attempts to enter the system application without entering a username or password, the user will receiver a message that states, “Please enter missing information.”

REQ-3 Vehicle Search: In order for users to reserve a car, the user must search for a vehicle by location on the *Vehicle Search Page*. If the vehicle entered is available, it will appear as a car that is available to rent/reserve at the specific location. If the vehicle entered is unavailable, the user will be presented with a list of all vehicles currently available to rent/reserve at the specified branch location. Users may also conduct a generic search of vehicles to see the availability of a certain vehicle at various branch locations; users may conduct this search on the main Search bar that is present on every page.

REQ-8 Customer Search: All employees have the ability to look up customers in the system application. Employees may search for customers in the system’s database using the *Customer Search Page.* Under this page, employees will enter a customer’s first name and last name to be searched for in the system. All customers with matching names will be displayed in a list below the search bar on the *Customer Search Page.* If a customer was not found in the system, a message will appear stating, “Customer was not found.”

REQ-4 Reserve a Car: When a user clicks on the *Reservation* Page, the user will be required to enter the vehicle name as well as the make, model, year, and VIN number to search the availability of the car within the system. If the car is available, the user will be required to enter a pickup and drop-off time for the rental car. The pickup time serves as a designated time slot to pick up the car from the rental car company, and the drop-off time serves as a designated time slot to drop off the car at the rental car company. (Pickup times are spaced out in 15-minute increments.) Users will confirm rental car reservations by clicking the “CONFIRM” button. All confirmed reservations will be added to the *Rental History Page*.

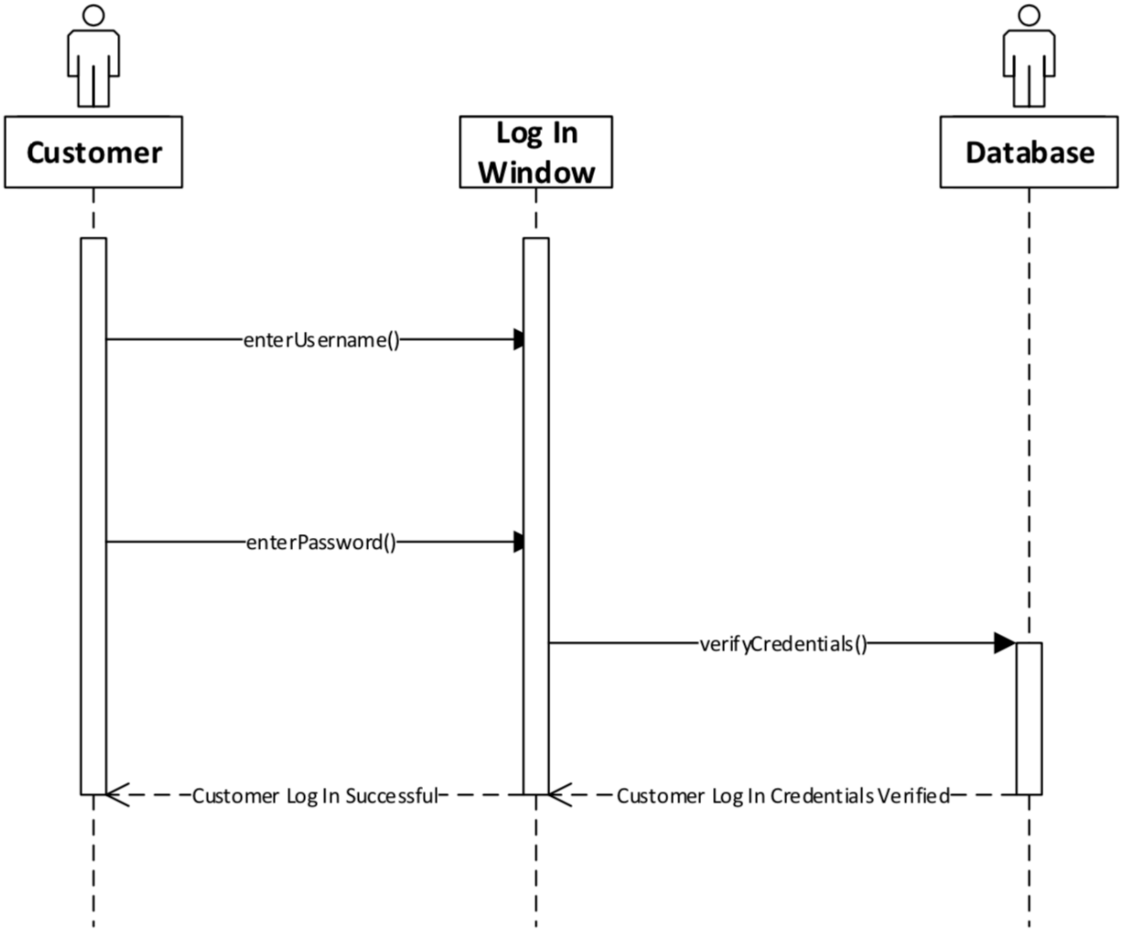
REQ-5 Update Reservation: If a user wants, or needs to update the pickup or drop-off time for their rental car, he or she may do so on the *Update Page.* Users will be asked to provide a reservation number to access their car reservation. Once a customer’s reservation is pulled up, users will be asked to enter a new pickup and drop off time under the “Update Pickup Time” and “Update Drop-Off Time” sections. Users will be required to click a “UPDATE” button, which will make the necessary changes to the reservation. Users will receive a message stating, “Reservation has been successfully updated.”

REQ-6 Cancel Reservation: In addition to updating pickup and drop-off times, users may also cancel reservations on the *Update Page.* Similar to updating a reservation, users must provide a reservation number to access a particular car reservation. Once a customer’s reservation is located, the user is able to select a button that says, “CANCEL”. The “CANCEL” button will cancel the reservation for the designated car and respective customer. A message will appear to verify the user’s intended action. If a reservation is successfully cancelled, a message will be displayed stating, “Reservation has been cancelled.” Cancelled reservations will be removed and not appear in the *Rental History Log*.

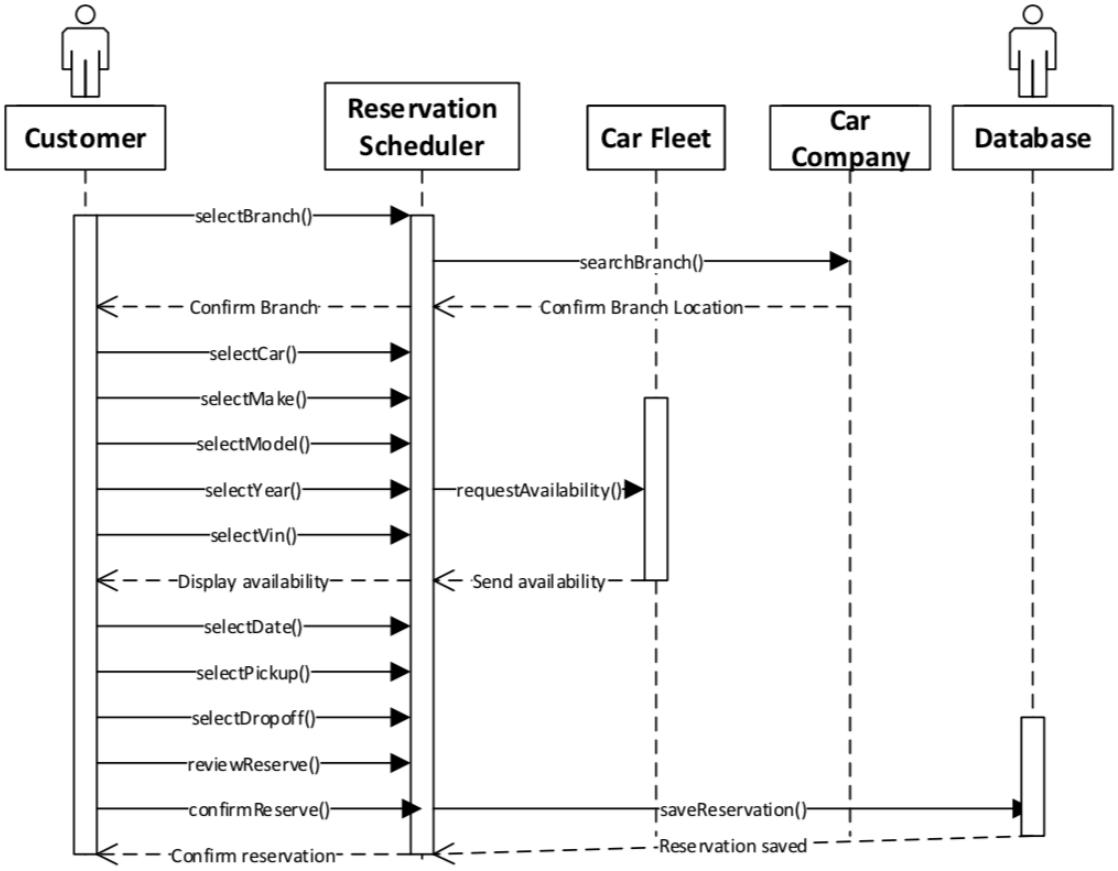
REQ-7 Access History Log: All users may view previous rental history on the *Rental History Page*. Customers are able to view information regarding all of their previous car rentals. Similarly, staff members are able to view a customer’s previous rental history as well. However, employees will enter a customer’s first and last name to search for in the database. A list of matching customers will appear for the employee to choose from. Once an employee selects a customer, a list of previous reservations for that respective customer will be displayed. Users may view the details regarding a particular reservation by clicking on a reservation.

System Sequence Diagram

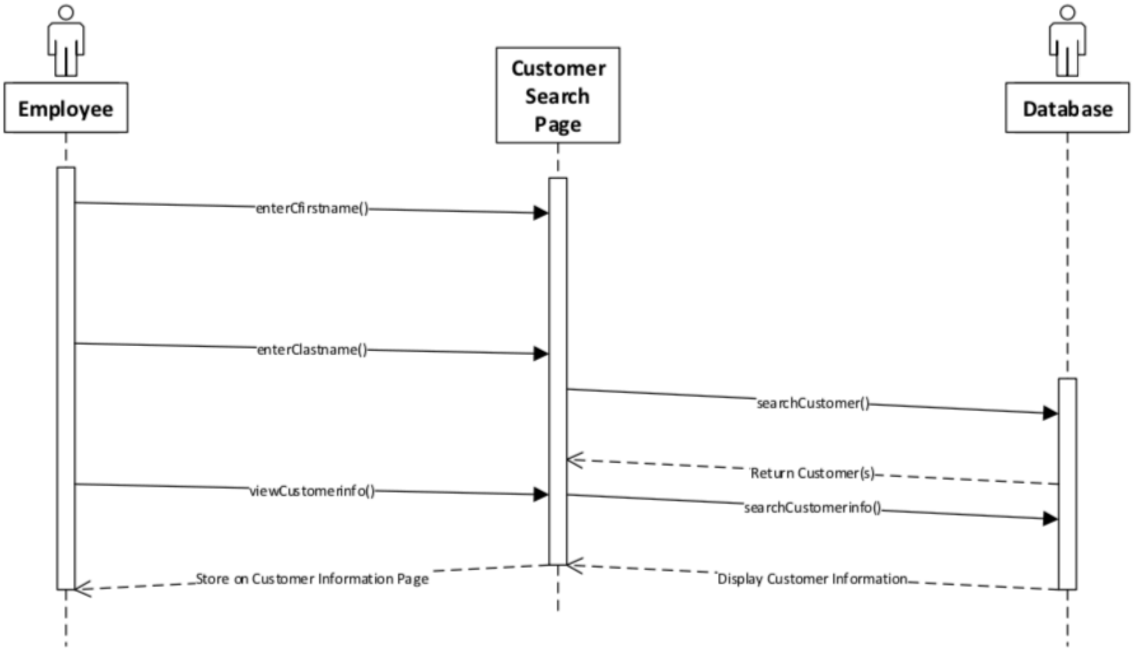
Customer Log In



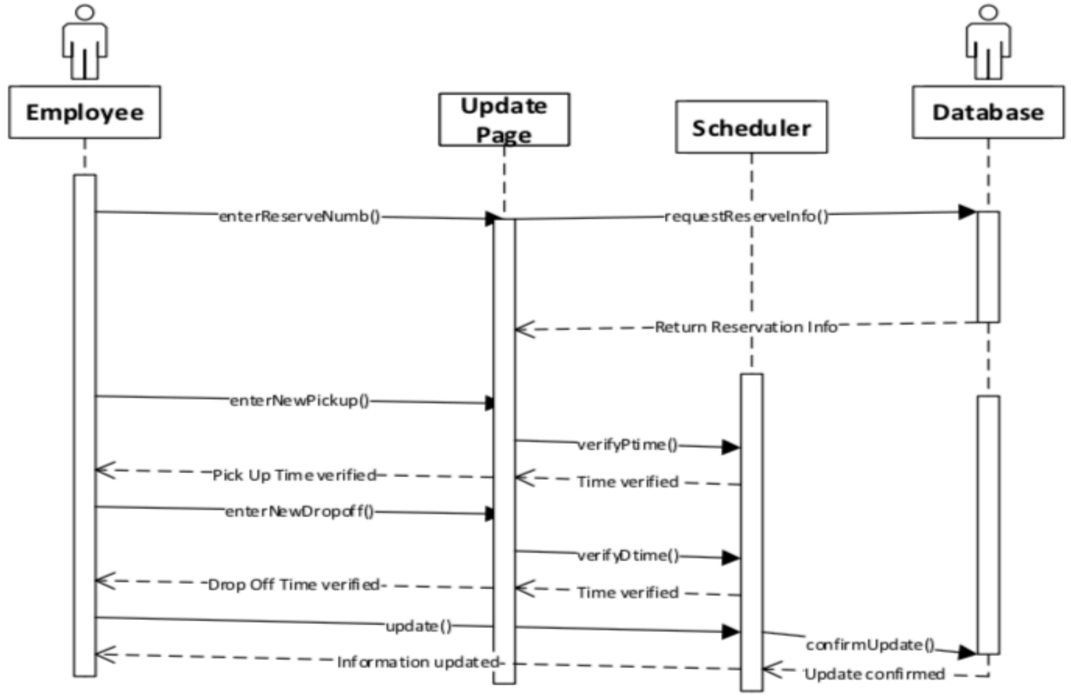
Customer reserving a car



Employee searching for a customer



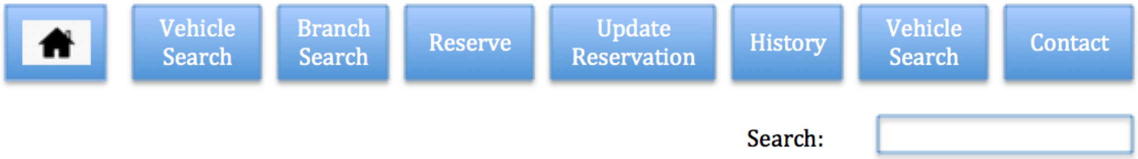
Employee updating a customer reservation



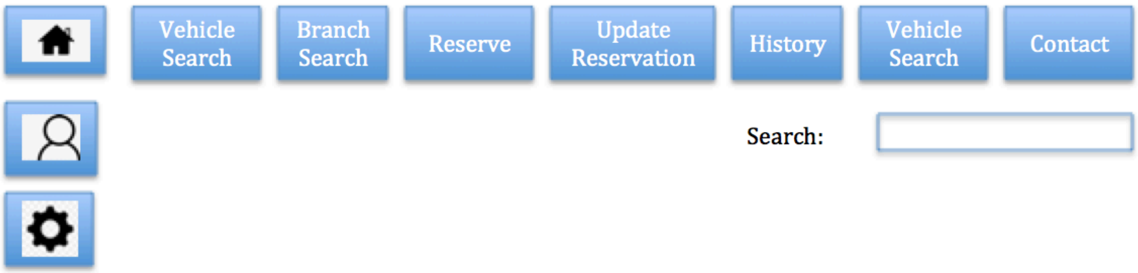
User Interface Specification

Preliminary Design

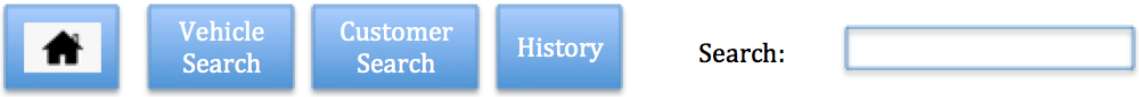
Upon logging into the system application, users will experience one of two views. Employees will view the home page from an administrator's view, whereas customers will view the home page from the home page view.



Generic View

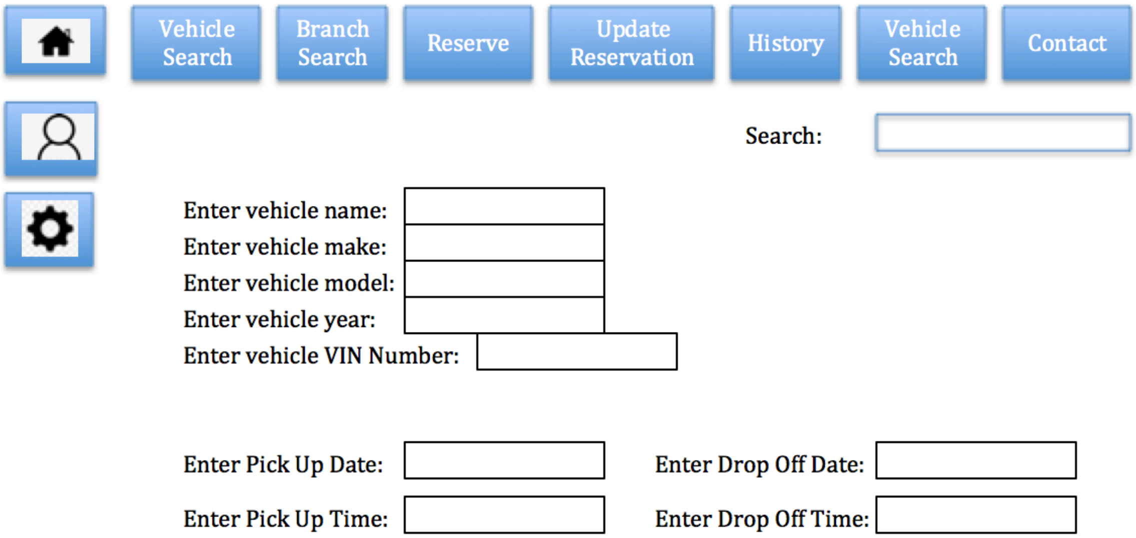


Customer View



Employee View

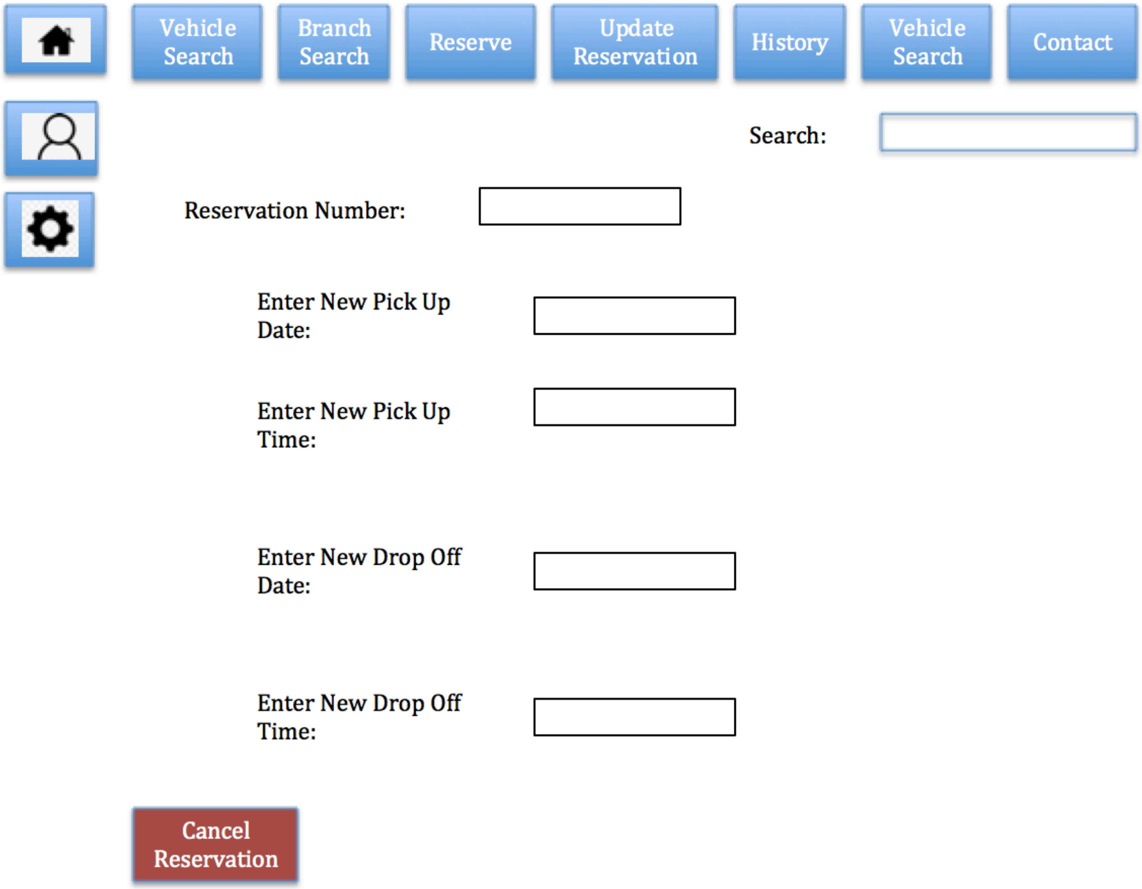
When a customer wants to rent a car, he or she will click on the "Reserve" tab. The *Reservation Page* prompts user to enter vehicle information to search for the vehicle's availability. In addition, users are asked to enter a date and time designated to pick up and drop-off the vehicle.



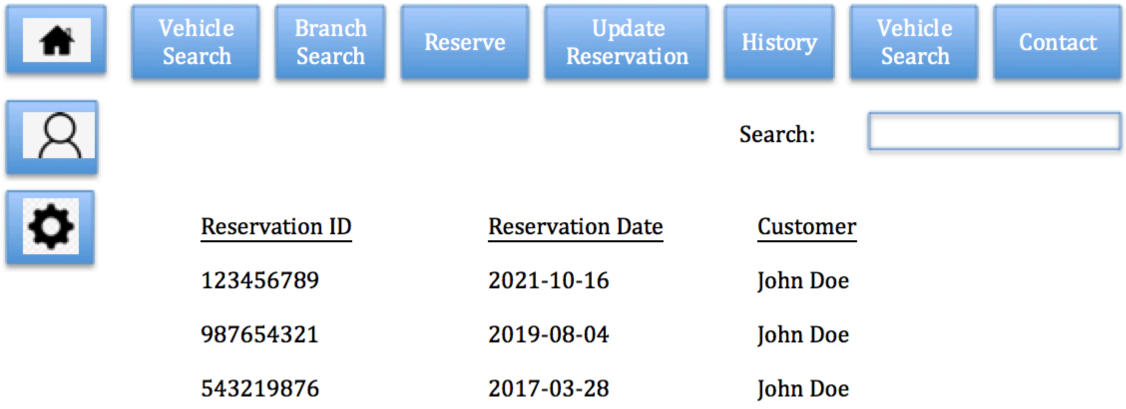
Enter vehicle name:
Enter vehicle make:
Enter vehicle model:
Enter vehicle year:
Enter vehicle VIN Number:

Enter Pick Up Date:
Enter Pick Up Time:
Enter Drop Off Date:
Enter Drop Off Time:

When a customer wants to update, or cancel a car reservation, he or she will click on the “Update Reservation” tab. The *Update Page* prompts users to enter a reservation number to search for in the database. Once the reservation has been located, users may click on the reservation and update reservation information. Users are able to update the date and time in which the pick up their vehicle, drop off their vehicle, or both. In addition, users are able to cancel their reservation from the *Update Page*. Users will receive a confirmation message stating that their reservation has been updated successfully. Similarly, users will receive a warning message before a reservation is cancelled.



Users are able to access car rental history logs through the "History" tab. By clicking on the "History" tab, users will be redirected to the *Rental History Page*. For customers, a list of their previous car rentals will appear. However, employees will have to search a customer by the customer's first and last name. Customers whose name match the one entered by an employee will appear in a list. An employee may access a customer's car rental history by clicking on the customer's name. The car rental history log displays the reservation ID number, date of the reservation, and the name of the customer on the reservation.



Reservation ID	Reservation Date	Customer
123456789	2021-10-16	John Doe
987654321	2019-08-04	John Doe
543219876	2017-03-28	John Doe

User Effort Estimation

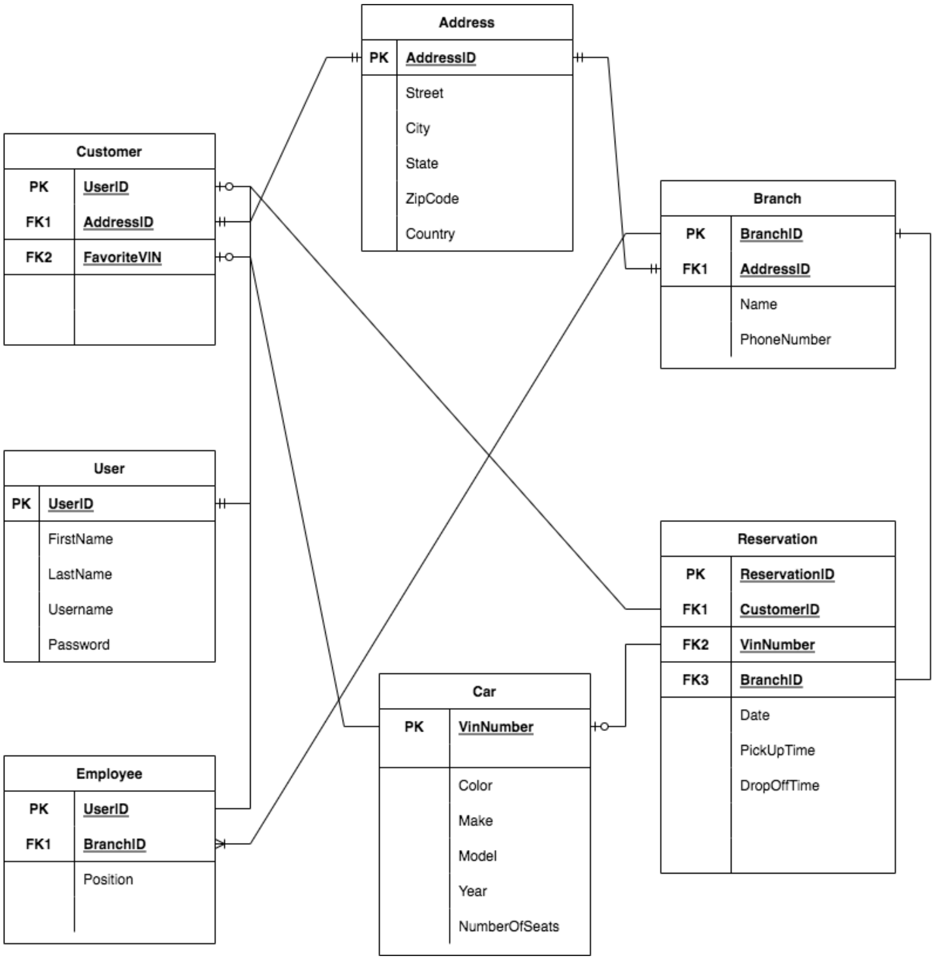
The chart below displays the least amount of clicks to complete a given activity.

Use Case	Least Amount of Clicks
Log In	3
Search for Car	2
Reserve a Car	8/9

Reschedule Pickup/Drop-off Time	4
Cancel Reservation	4
Access Rental History Log (Customer)	2
Access Rental History Log (Staff)	3
Update Personal Information (Customer)	2
Update Customer Information (Staff)	4
Branch Location Search	3
Log Out	1

SECTION 3

Domain Analysis



The diagram pictured above displays the details regarding the relationship between each entity. Each entity contains attributes that help describe, or define the entity itself. The attributes are listed in the tables underneath the entity names. Primary keys (PK) and foreign keys (FK) are labeled to the left of the corresponding attribute(s). A primary key is an attribute that is used to uniquely identify an entity. A foreign key is an attribute, or field that must match the primary key of another entity; foreign keys are essential for establishing relationships between entities, or tables. Entity relationships are displayed above through the various start and end points of the lines connecting to the tables. To get a better understanding of the diagram pictured above, please refer to the *System Diagram Glossary* below.

System Diagram Glossary

Entity Definitions:

- Customer – this entity will store personal information regarding each individual customer of the RentaCar company. Such information includes: UserID, AddressID, and FavoriteVIN.
- User – this entity will store information regarding each user of the RentaCar system. Such information includes: UserID, FirstName, LastName, Username, and Password.
- Employee – this entity will store information regarding each employee of the RentaCar company. Such information includes UserID, BranchID, and Position.

- Address – this entity will store information about the origin of a particular place. Such information includes: AddressID, Street, City, State, ZipCode, and Country.
- Branch – this entity will store information about each RentaCar branch location. Such information includes: BranchID, AddressID, Name, and PhoneNumber.
- Reservation – this entity will store information about customers' RentaCar reservations. Such information includes: ReservationID, CustomerID, VinNumber, BranchID, Date, PickUpTime, and DropOffTime.
- Car – this entity will store information regarding a particular car. Such information includes: VinNumber, Color, Make, Model, Year, and NumberOfSeats.

Attribute Definitions:

Customer:

- UserID – this attribute is used as the primary key (PK) of the Customer table. This attribute will have an integer data type and be used to uniquely identify each customer in the RentaCar system.
- AddressID – this attribute is used as a foreign key (FK1) in the Customer table. This attribute will have an integer data type and be used to store and access a customer's address in the RentaCar system.
- FavoriteVIN – this attribute is used as a foreign key (FK2) in the Customer table. This attribute will have a integer data type and be used to store

information about a particular car. Customers will have the option of whether or not to favorite vehicles based on their rental experience.

User:

- UserID – this attribute is used as the primary key (PK) of the User table. This attribute will have an integer data type and be used to uniquely identify each user of the RentaCar system. Therefore, this attribute will be used to determine whether a user is a customer, or an employee.
- FirstName – this attribute will have a string data type and store the first name of the user.
- LastName – this attribute will have a string data type and store the last name of the user.
- Username – this attribute will have a string data type and store the username of the user.
- Password – this attribute will have a string data type and store the password of the user.

Employee:

- UserID – this attribute is used as the primary key (PK) of the Employee table. This attribute will have an integer data type and be used to uniquely identify each RentaCar employee.
- BranchID – this attribute is used as a foreign key (FK1) in the Employee table. This attribute will have an integer data type and be used to store and access employees associated to a specific RentaCar branch location.

- Position – this attribute will have a string data type and store the position, or role of each RentaCar employee.

Address:

- AddressID – this attribute is used as the primary key (PK) of the Address table. This attribute will have an integer data type and be used to uniquely identify each address in the RentaCar system.
- Street – this attribute will have a string data type and store the number and name of the street address.
- City – this attribute will have a string data type and store the names of a city.
- State – this attribute will have a string data type and store names of a state.
- ZipCode – this attribute will have an integer data type and store address zip codes.
- Country – this attribute will have a string data type and store the name of a country.

Branch:

- BranchID – this attribute is used as the primary key (PK) of the Branch table. This attribute will have an integer data type and be used to uniquely identify each RentaCar branch location.
- AddressID – this attribute is used as a foreign key (FK1) in the Branch table. This attribute will have a string data type and be used to store and access the address of a RentaCar branch.
- Name – this attribute will have a string data type and store the name of the RentaCar branch location.

- PhoneNumber – this attribute will have a string data type and store the phone number of RentaCar branch locations.

Reservation:

- ReservationID – this attribute is used as the primary key (PK) of the Reservation table. This attribute will have an integer data type and be used to uniquely identify each customer reservation in the RentaCar system.
- CustomerID – this attribute is used as a foreign key (FK1) in the Reservation table. This attribute will have an integer data type and be used to store and access a customer in the RentaCar system.
- VinNumber – this attribute is used as a foreign key (FK2) in the Reservation table. This attribute will have an integer data type and be used to store and access a car in the RentaCar system.
- BranchID – this attribute is used as a foreign key (FK3) in the Reservation table. This attribute will have an integer data type and be used to store and access a branch location in the RentaCar system.
- Date – this attribute will have a date/time data type and be used to store the date of a customer reservation in the RentaCar system.
- PickUpTime – this attribute will have a date/time data type and be used to store a specific time to pick up the car associated to the customer reservation.
- DropOffTime - this attribute will have a date/time data type and be used to store a specific time to drop off the car associated to the customer reservation.

Car:

- VinNumber – this attribute is used as the primary key of the Car table. This attribute will have an integer data type and be used to uniquely identify each car in the RentaCar system.
- Color – this attribute will have a string data type and store the color of each car.
- Make – this attribute will have a string data type and store the make, or manufacturer of the car.
- Model – this attribute will have a string data type and store the model name of each car.
- Year – this attribute will have an integer data type and store the year of the car.
- NumberOfSeats – this attribute will have an integer data type and store number of seats in each car.

Relationships:

- A User may be either a Customer, or an Employee.
- A Customer must be a User, but a User does not have to be a Customer.
- A Customer has one Address, and one Address belongs to one Customer.
- A Customer may favorite a Car, but a Car does not have to be favorited by a Customer.
- An Employee must be a User, but a User does not have to be an Employee.
- An Employee is associated to one Branch, but a Branch may have many employees.

- A Branch has many Employees, but an Employee is associated to only one Branch.
- A Branch has one Address, and one Address is associated to one Branch.
- A Reservation is associated to one Customer, but a Customer may have many Reservations.
- A Reservation is associated to one Car, but a Car may be associated to many Reservations.
- A Reservation is associated to one Branch, but a Branch may be associated to many Reservations.

System Operation Contracts

Use Case: Log In

Contract C01: enterUsername	
Operation:	enterUsername(username: UserName)
Cross Reference:	Use Case: Log In
Preconditions:	A User is attempting to log into the RentaCar system.
Postconditions:	• Username is associated with a Customer. • Username is associated with a RentaCar Employee.

Contract C02: enterPassword	
Operation:	enterPassword(password: PassWord)
Cross Reference:	Use Case: Log In
Preconditions:	A User is attempting to log into the RentaCar system.
Postconditions:	• Password is associated with a Customer. • Password is associated with a RentaCar Employee.

Contract C03: clickLogin	
Operation:	clickLogin(username: UserName, password: PassWord)
Cross Reference:	Use Case: Log In
Preconditions:	A User is attempting to log into the RentaCar system.
Postconditions:	• Username and Password are associated to a Customer. • Username and Password are associated to a RentaCar Employee.

Contract C04: checkCredentials	
Operation:	checkCredentials(username: UserName, password: PassWord)
Cross Reference:	Use Case: Log In
Preconditions:	A User is attempting to log into the RentaCar system.
Postconditions:	• Username and Password are associated to a Customer. • Username and Password are associated to a RentaCar Employee.

Use Case: Vehicle Search

Contract C01: enterVehicle	
Operation:	enterVehicle(vehicleName: vname)
Cross Reference:	Use Case: Vehicle Search
Preconditions:	A User is searching for a vehicle in the RentaCar system.
Postconditions:	• Vehicle name becomes the vehicle name entered

Contract C02: enterVehicle	
Operation:	enterVehicle(vinNumber: vin, color: c, make: m, model: mod, year: y)
Cross Reference:	Use Case: Vehicle Search
Preconditions:	A User is searching for a vehicle in the RentaCar system.
Postconditions:	• Vin Number becomes the VIN number entered • Color becomes the vehicle color entered • Make becomes the vehicle make entered • Model becomes the vehicle model entered • Year becomes the year of the vehicle entered

Contract C03: clickSearch	
Operation:	clickSearch
Cross Reference:	Use Case: Vehicle Search
Preconditions:	A User is searching for a vehicle in the RentaCar system.

Postconditions:	• Vin number is associated to the VIN Number of a car in the RentaCar system. • Color is associated to a car in the RentaCar system. • Make is associated to a car in the RentaCar system. • Model is associated to a car in the RentaCar system. • Year is associated to a car in the RentaCar system.

Use Case: Reserve a Car

Contract C01: newReservation	
Operation:	newReservation()
Cross Reference:	Use Case: Reserve a Car
Preconditions:	A User is attempting to reserve a car.
Postconditions:	• newReservation instance was created.

Contract C02: enterVIN	
Operation:	enterVIN(vinNumber: vin)
Cross Reference:	Use Case: Reserve a Car
Preconditions:	A User is attempting to reserve a car.
Postconditions:	• Vin Number becomes the VIN number entered

Contract C03: searchVIN	
Operation:	searchVIN(vinNumber: vin)
Cross Reference:	Use Case: Reserve a Car
Preconditions:	A User is attempting to reserve a car.
Postconditions:	• Vin Number is associated to a Car in the RentaCar system.

Contract C04: checkAvailability	
Operation:	checkAvailability(vinNumber: vin, reservation: res)
Cross Reference:	Use Case: Reserve a Car
Preconditions:	A User is attempting to reserve a car.
Postconditions:	• Vin Number is associated to a Car in the RentaCar system. • The car is associated to a

	reservation in the RentaCar system.

Contract C05: enterDate	
Operation:	enterDate(month: m, day: d, year: y)
Cross Reference:	Use Case: Reserve a Car
Preconditions:	A User is attempting to reserve a car.
Postconditions:	• Month becomes the month entered • Day becomes the day entered • Year becomes the day entered

Contract C06: enterPickUp	
Operation:	enterPickUp(pickupTime: t)
Cross Reference:	Use Case: Reserve a Car
Preconditions:	A User is attempting to reserve a car.
Postconditions:	• Pick up time becomes the pick up time selected.

Contract C07: enterDropOff	
Operation:	enterDropOff(dropoffTime: d)
Cross Reference:	Use Case: Reserve a Car
Preconditions:	A User is attempting to reserve a car.
Postconditions:	• Drop off time becomes the drop off time selected

Contract C08: enterDate	
Operation:	enterDate(month: m, day: d, year: y)
Cross Reference:	Use Case: Reserve a Car
Preconditions:	A User is attempting to reserve a car.
Postconditions:	• Month becomes the month entered • Day becomes the day entered • Year becomes the day entered

Contract C09: saveReservation	
Operation:	saveReservation()
Cross Reference:	Use Case: Reserve a Car
Preconditions:	A User is attempting to reserve a car.
Postconditions:	• Customer reservation is associated with a reservation in the RentaCar system.

Use Case: Update Reservation

Contract C01: updatePickUp	
Operation:	updatePickUp(pickupTime: t)
Cross Reference:	Use Case: Update Reservation
Preconditions:	A User is attempting to update a reservation.
Postconditions:	• Pick up time becomes the pick up time selected.

Contract C02: updateDropOff	
Operation:	enterDropOff(dropoffTime: d)
Cross Reference:	Use Case: Update Reservation
Preconditions:	A User is attempting to update a reservation.

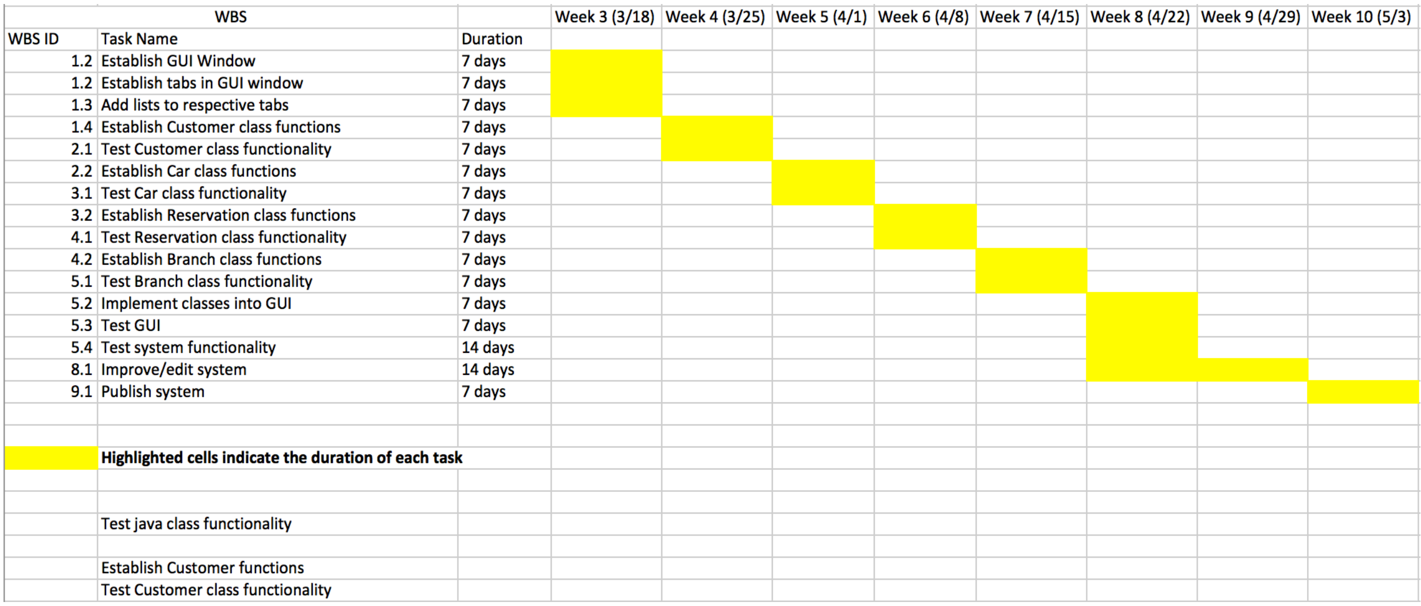
Postconditions:	• Drop off time becomes the drop off time selected.

Contract C03: saveReservation	
Operation:	saveReservation()
Cross Reference:	Use Case: Update Reservation
Preconditions:	A User is attempting to update a reservation
Postconditions:	• Customer reservation is associated with a reservation in the RentaCar system.

Project Size Estimation

Based on the use cases detailed in this report, I believe that my system application is quite large in size. There are many different colors, makes, and models of cars. In addition, a car manufacturer produces a new car each year. Therefore, this system application would need to store an extensive list of cars. A car rental company may also have branch locations in many different cities all over the world, which means this system would need to store almost all major cities, states, and countries. Due to the amount of significant information that must be stored in this system application, I would say that the size of this project is quite large. This system application is feasible and could potentially benefit a car rental company

Plan of Work



	WBS		Week 3 (3/18)	Week 4 (3/25)	Week 5 (4/1)	Week 6 (4/8)	Week 7 (4/15)	Week 8 (4/22)	Week 9 (4/29)	Week 10 (5/3)
WBS ID	Task Name	Duration								
1.2	Establish GUI Window	7 days	🟨							
1.2	Establish tabs in GUI window	7 days	🟨							
1.3	Add lists to respective tabs	7 days	🟨							
1.4	Establish Customer class functions	7 days		🟨						
2.1	Test Customer class functionality	7 days		🟨						
2.2	Establish Car class functions	7 days			🟨					
3.1	Test Car class functionality	7 days			🟨					
3.2	Establish Reservation class functions	7 days				🟨				
4.1	Test Reservation class functionality	7 days				🟨				
4.2	Establish Branch class functions	7 days					🟨			
5.1	Test Branch class functionality	7 days					🟨			
5.2	Implement classes into GUI	7 days						🟨		
5.3	Test GUI	7 days						🟨		
5.4	Test system functionality	14 days						🟨		
8.1	Improve/edit system	14 days						🟨	🟨	
9.1	Publish system	7 days								🟨
🟨	**Highlighted cells indicate the duration of each task**									
	Test java class functionality									
	Establish Customer functions									
	Test Customer class functionality									

Gantt Chart

Pictured above is an updated Gantt Chart that displays my proposed plan of work for the rest of this semester.

After experiencing some trouble with creating the GUI, I have decided to break the GUI down into smaller chunks, or tasks. I'm going to start by first getting a window to appear to users of the application. Second, I'm going to add tabs to that window. Once tabs are added to window, I can begin focusing on implementing lists from the Access database into my system; the lists should appear under the relevant tab in the GUI window. Once the GUI has been constructed, I then plan on focusing and sharpening the functionality of Java classes in the system application. I will dedicate the remaining time I have to debugging and improving the RentaCar application.

REFERENCES

"Lecture 6." *Larman: Chapter 13*, http://csis.pace.edu/~marchese/CS616/Lec6/se_l6.htm.

*This reference provided me with an understanding of how system operation contracts are constructed.

"Home." *Rental Cars at Low, Affordable Rates | Enterprise Rent-A-Car*, https://www.enterprise.com/en/home.html.

*I used Enterprise's website as a general reference, or guide on how I want the layout of my system application to appear.