CAR RENTAL APP

Software Design Document

Ionica Mihai

Petrescu Alin

Nicolae Mihai

Software Engineering laboratory

March 2021

TABLE OF CONTENTS

1. Introduction

1.1 Purpose

The Software Design Document purpose is to describe the architecture and system design of the Car Rental application. It describes the use cases detailed in the SRS document which will be implemented using the design presented in this document. The document is intended to be used by the software developers or software testers who are directly involved into the development of the application.

1.2 Scope

The application is a tool that facilitates the interaction between a car rental shop and a potential customer who is in need for a car for a limited period.

A customer can specify some criteria like size, seats, price.

An administrator can keep track of the rented cars or add/delete existing models.

1.3 Overview

-1st part: short description of the application, references, and acronyms

-2nd part: short description of the application

-3rd part: diagrams about the functionality of the application

-4th part: data description and the dictionary of the database data

-5th part: a presentation of each design component

-6th part: description of the user interaction

-7th part: requirement matrix -

1.4 Reference Material

ASP. NET:

https://dotnet.microsoft.com/apps/aspnet/mvc

https://docs.microsoft.com/en-us/aspnet/core/?view=aspnetcore-5.0

https://docs.microsoft.com/en-us/dotnet/framework/

https://docs.microsoft.com/en-us/dotnet/csharp/

Database:

https://docs.microsoft.com/en-us/sql/?view=sql-server-ver15

CSS Framework:

https://getbootstrap.com/

Other:

https://en.wikipedia.org/wiki/Car_rental

1.5 Definitions and Acronyms

SRS – software requirement specifications

User – a individual who has an interaction with the application.

Database – a collection of data that can be accessed by a user using a database management system.

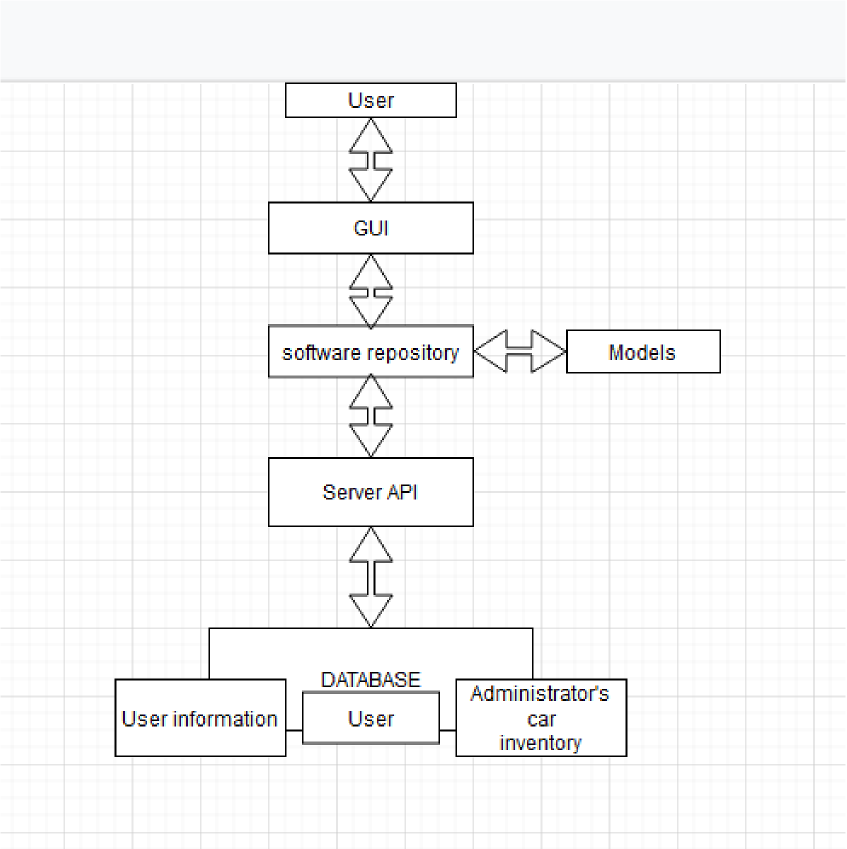
2. System Overview

The application was developed for everyone who is in need for a rental car and wants a fast and customer friendly experience by using a web browser. The application is designed to run on the most popular web browsers (Firefox, Chrome, Microsoft Edge etc.).

3. System Architecture

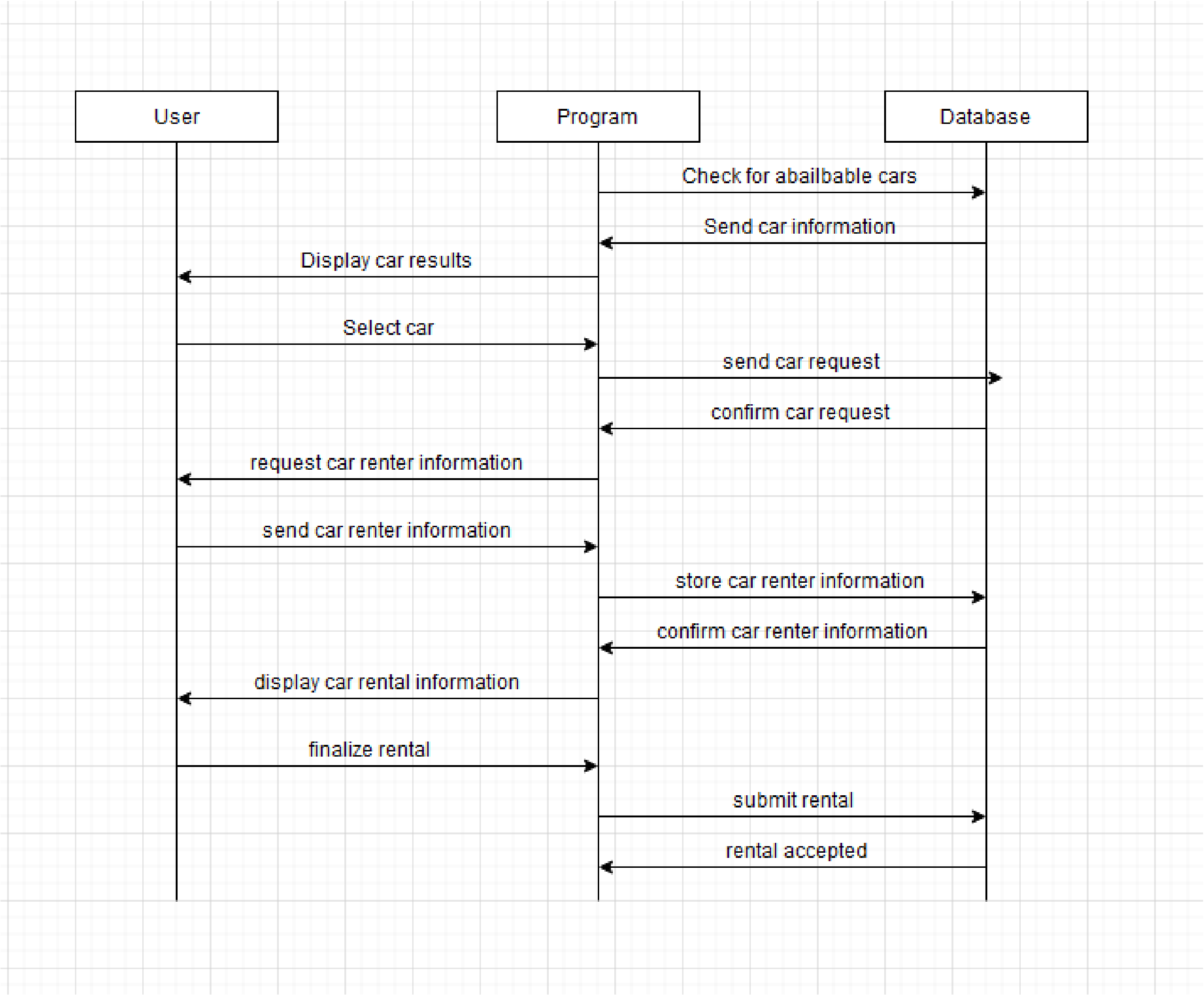
3.1 Architectural Design

The interaction between different parts of the system:

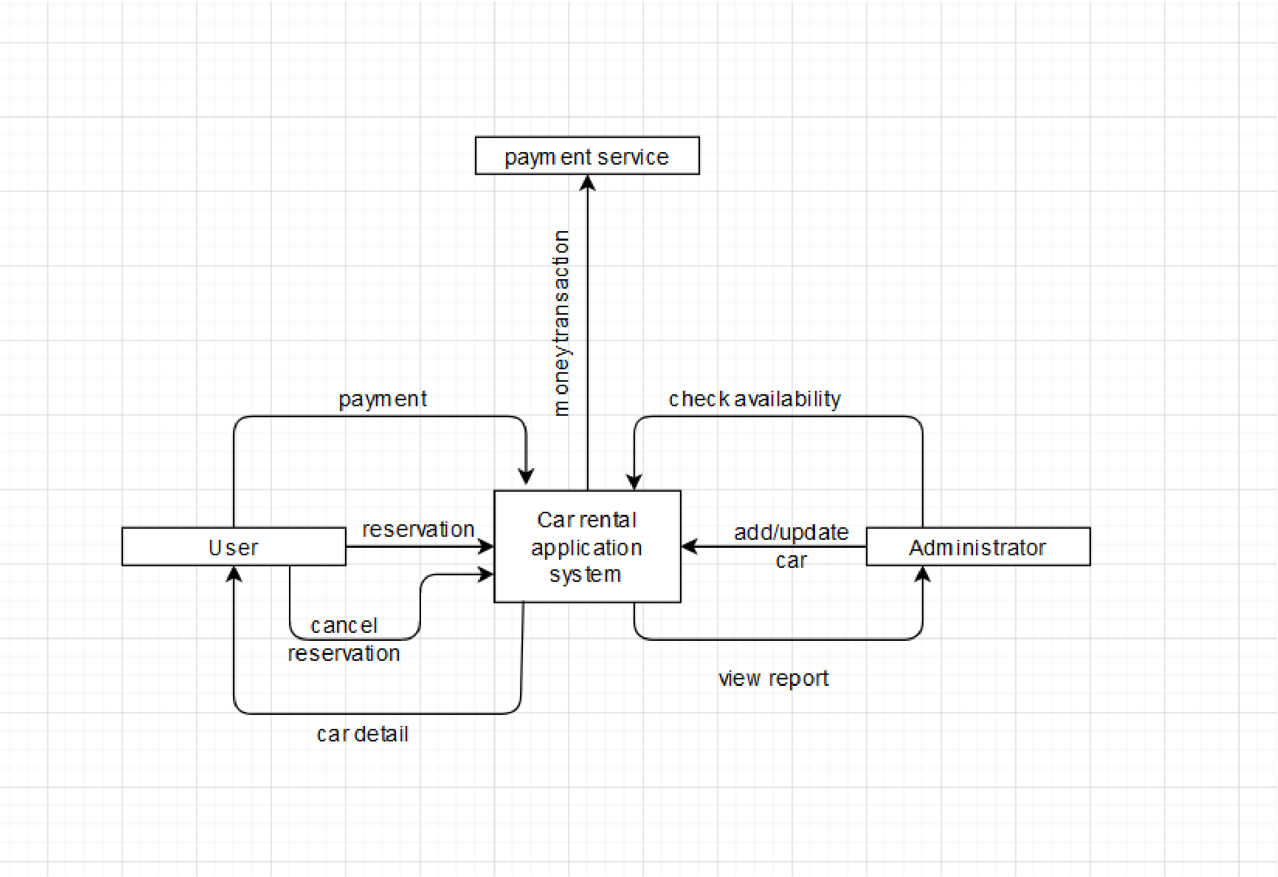


3.2 Decomposition Description

The Sequence Diagram (data flow) presents the interaction that is made possible by the car rental system between the user and database:


User
Program
Database
Check for abailbable cars
Send car information
Display car results
Select car
send car request
confirm car request
request car renter information
send car renter information
store car renter information
confirm car renter information
display car rental information
finalize rental
submit rental
rental accepted

This diagram shows the overall view of the system, the data that goes in, and how it flows to other entities.



3.3 Design Rationale

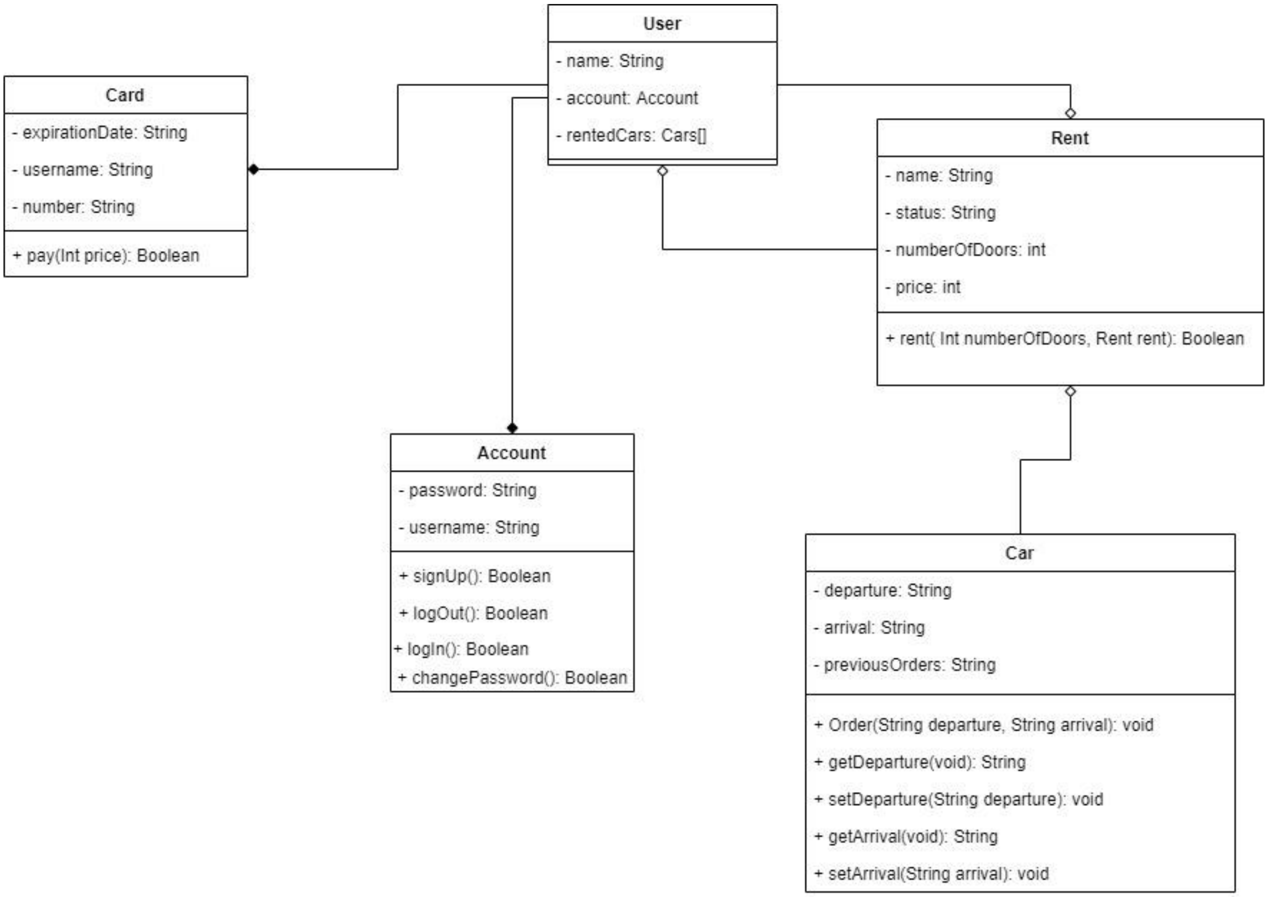
We choose a web architecture because it allows decentralized usage and offers an easier maintenance over time. Having a single repository which chooses to get the data (as API calls to the server or as local models) is also one of the reasons for this choice.

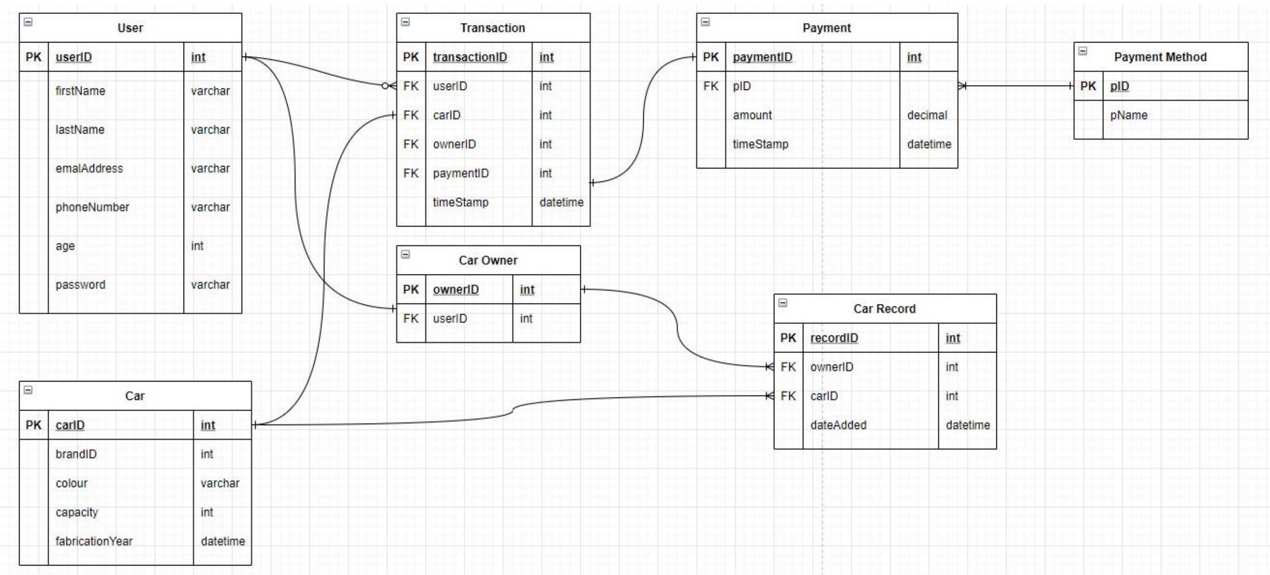
4. Data Design

4.1 Data Description

The application uses a single database which consists of 5 components:

- Admin: the admin personal details.
- User: the user personal details.
- Reservation: details about the reservation such as the pickup and drop off time and the username of the person who did rented the car.
- Payment: details about the payment after the transaction was done, the id of the individuals that made the transaction and the amount of money.
- Car: the car specification and its status (rented or not).





4.2 Data Dictionary

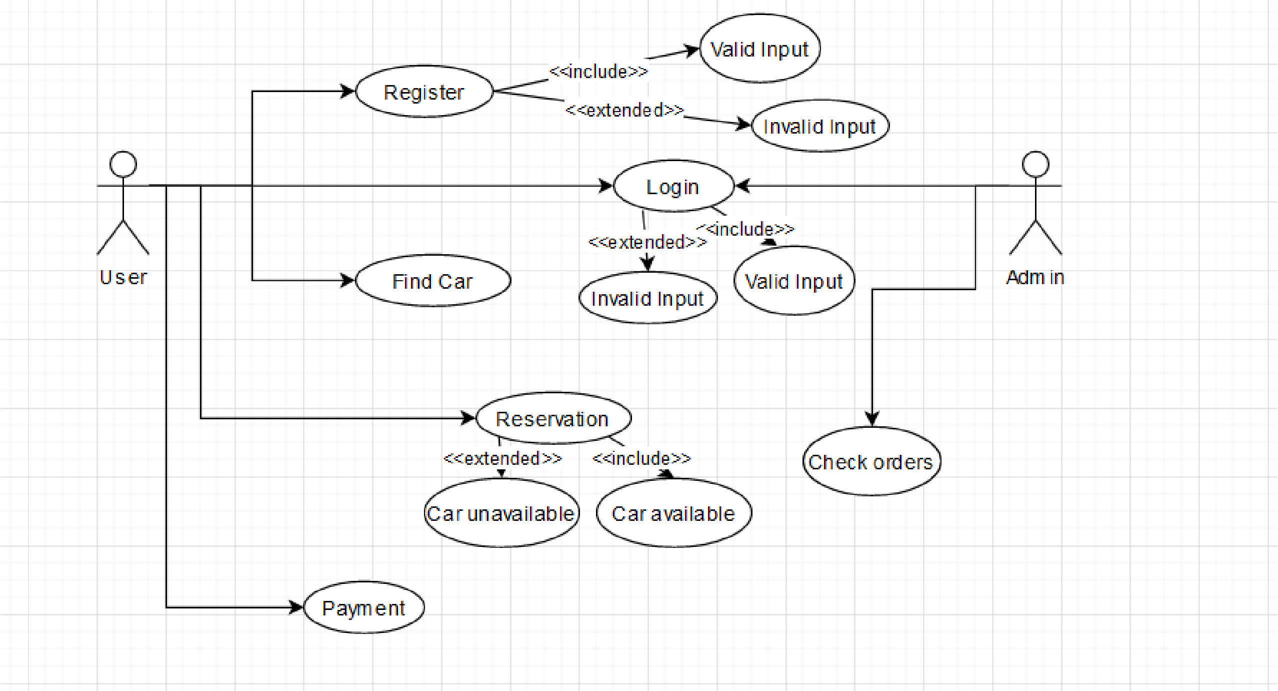
	Field	Type
User	firstName	varchar (32)
	lastName	varchar(32)
	password	varchar (32)
	email	varchar (64)
	age	int

Car	carID	int
	brandID	int
	colour	varchar
	capacity	int
	fabricationYear	datetime
Payment	paymentID	int
	pID	int
	amount	decimal
	timeStamp	datetime
Transaction	transactionID	int
	userID	int
	carID	int
	ownerID	int
	paymentID	int
	timeStamp	int

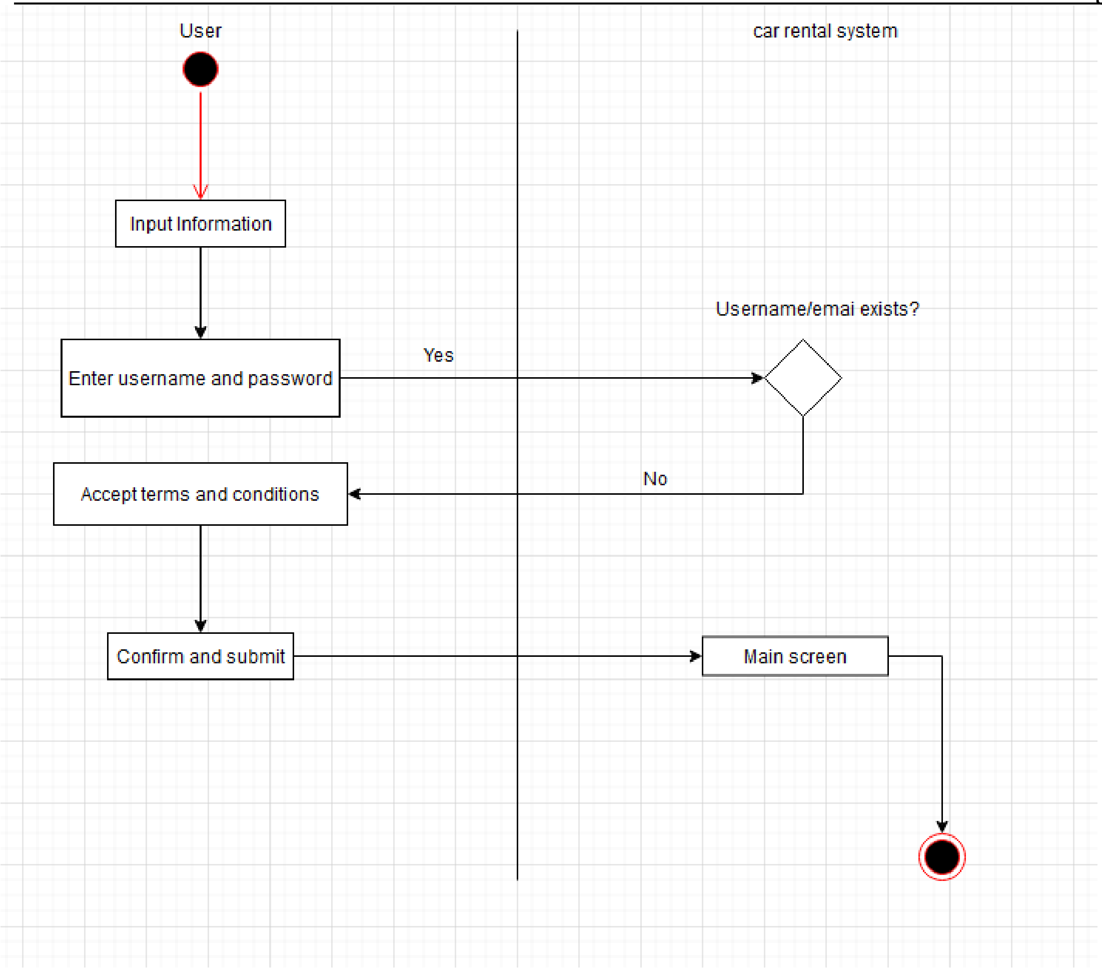
Car Owner	ownerID	int
	userID	int
Car Record	recordID	int
	ownerID	int
	carID	int
	dateAdded	datetime
Payment Method	pID	int
	pName	varchar(32)

5. Component Design

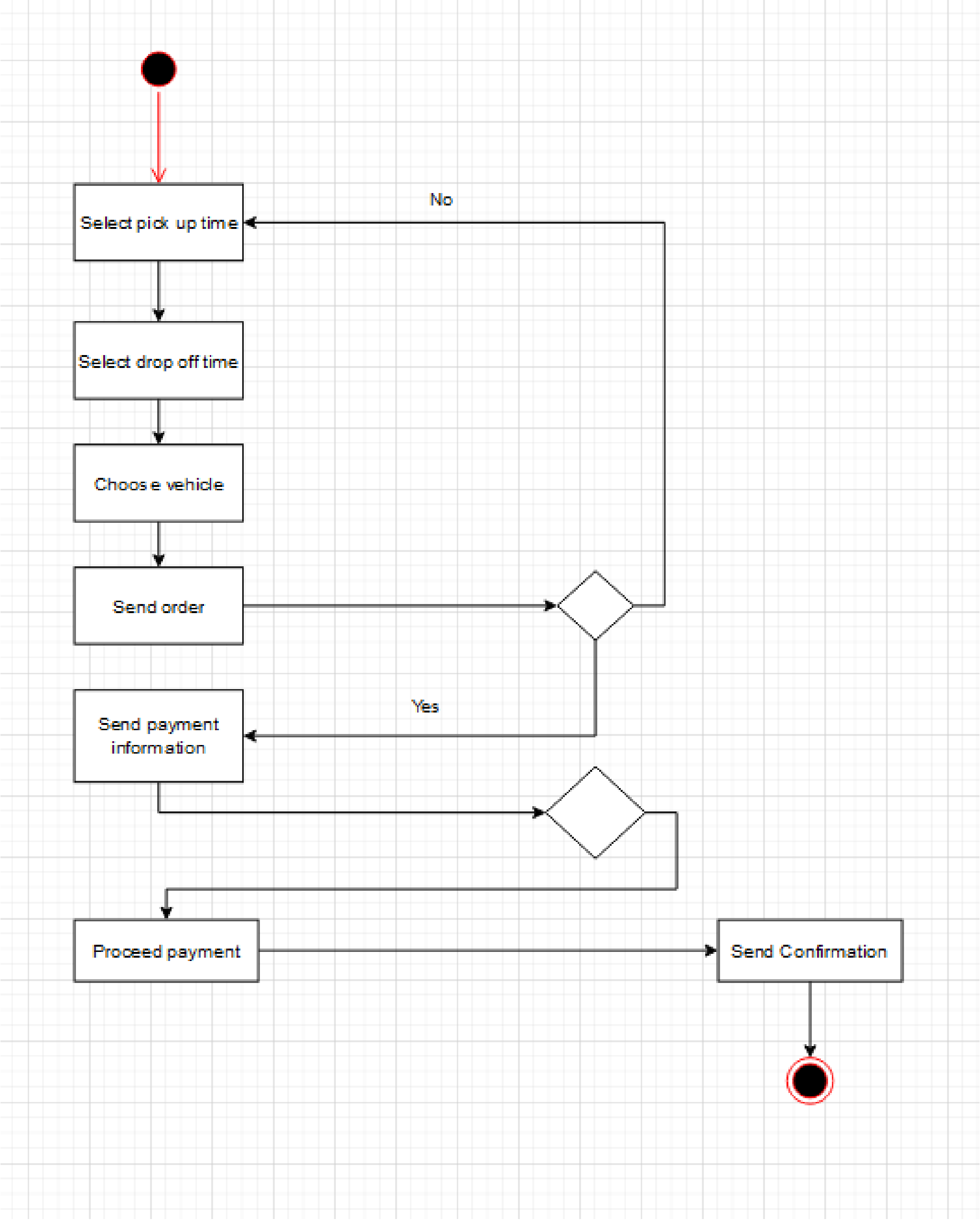
Use case:


User
Adm in
Register
<<include>>
Valid Input
<<extended>>
Invalid Input
Login
<<extended>>
<<include>>
Invalid Input
Valid Input
Find Car
Reservation
<<extended>>
<<include>>
Car unavailable
Car available
Check orders
Payment

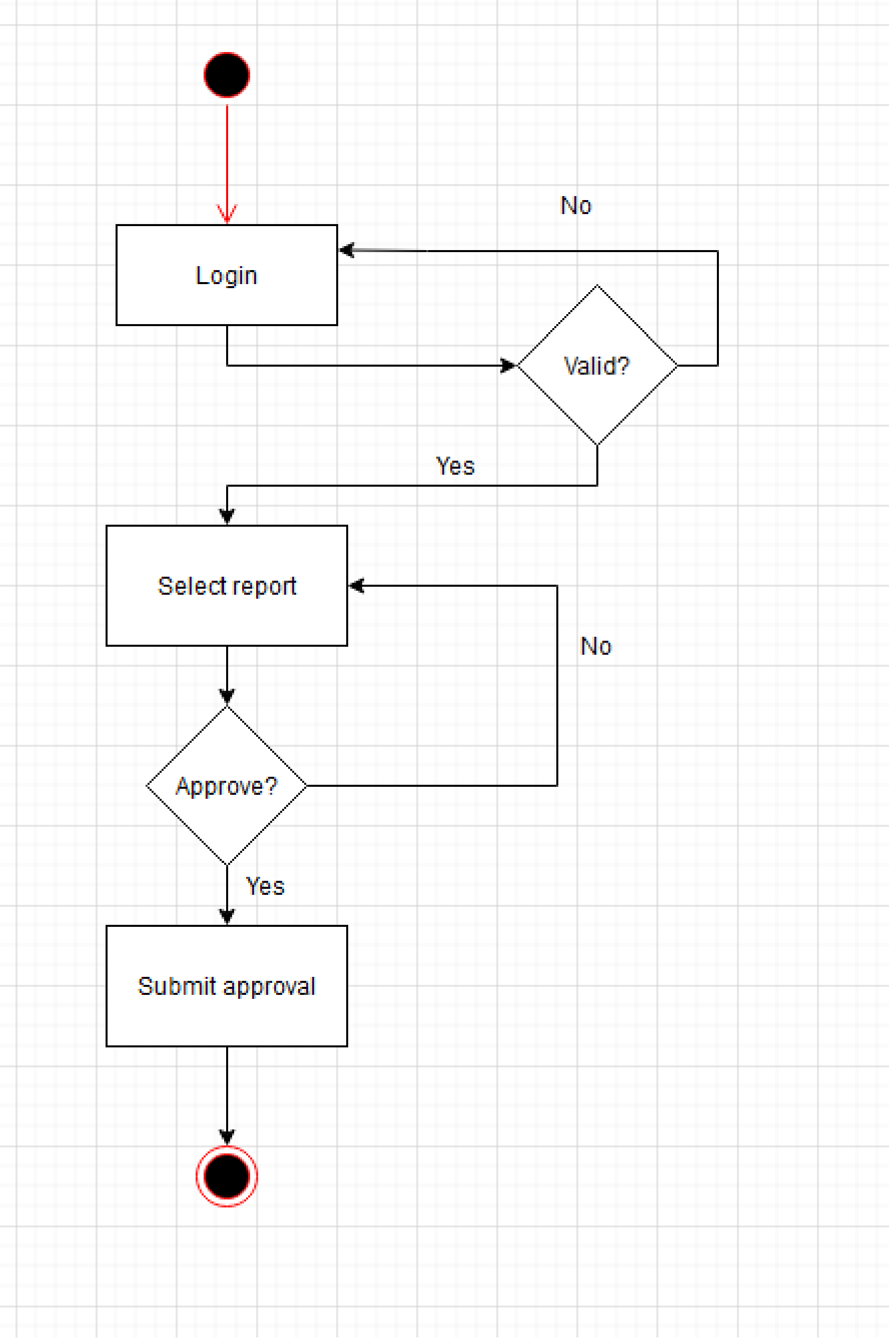
Sign up:



Car reservation:



Admin report:

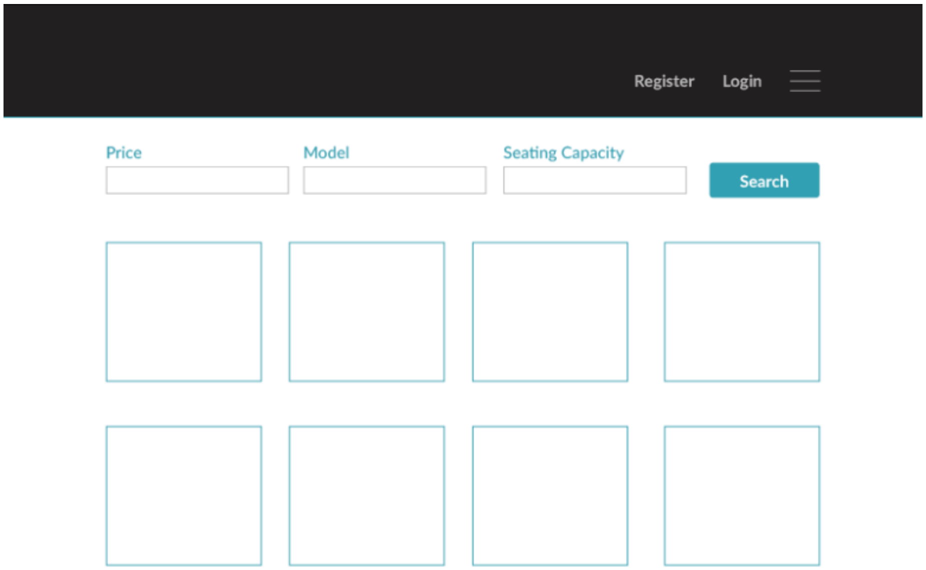

No
Login
Valid?
Yes
Select report
No
Approve?
Yes
Submit approval

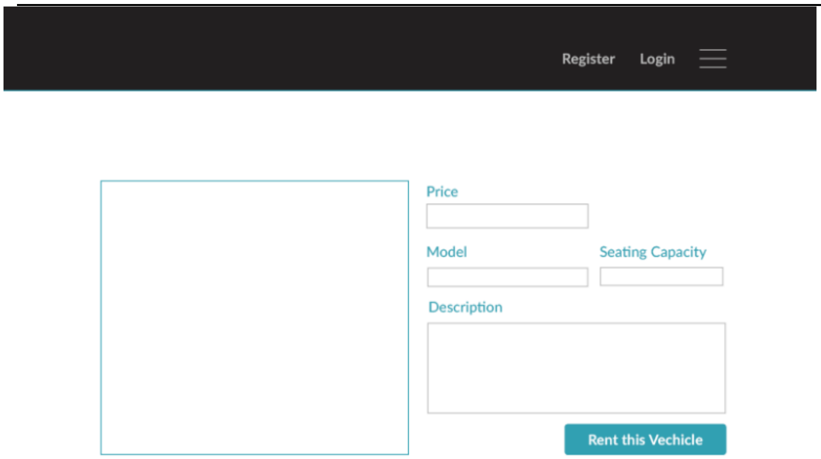
6. Human Interface Design

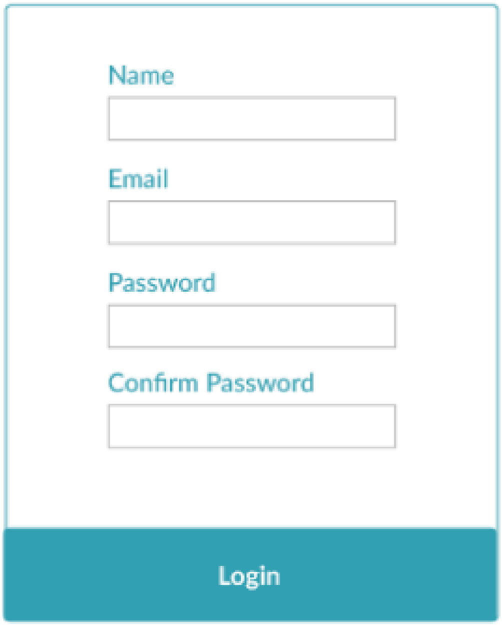
6.1 Overview of User Interface

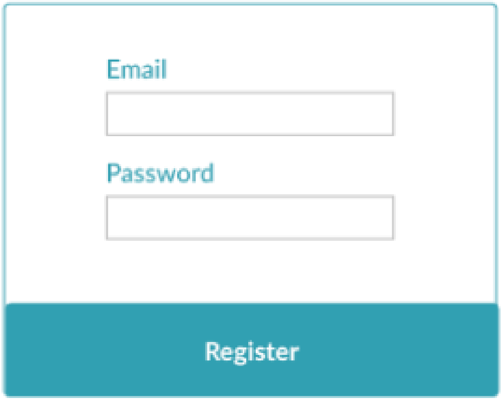
The page should have a “Home Button” concealed by a logo or a suggestive picture. A log in button should also be visible next to a button for creating an account (“Sign up”). There should also be a section of the page where images of recommended cars are shown. If the user is already logged in, then a button for the profile (“My account”) of the current user should be on the opposing side of the Home page button. The website will have a text bar for searching deals or cars. The searching process should be easy, with options for criteria and filters, some of which will be represented by checkboxes.

6.2 Screen Images

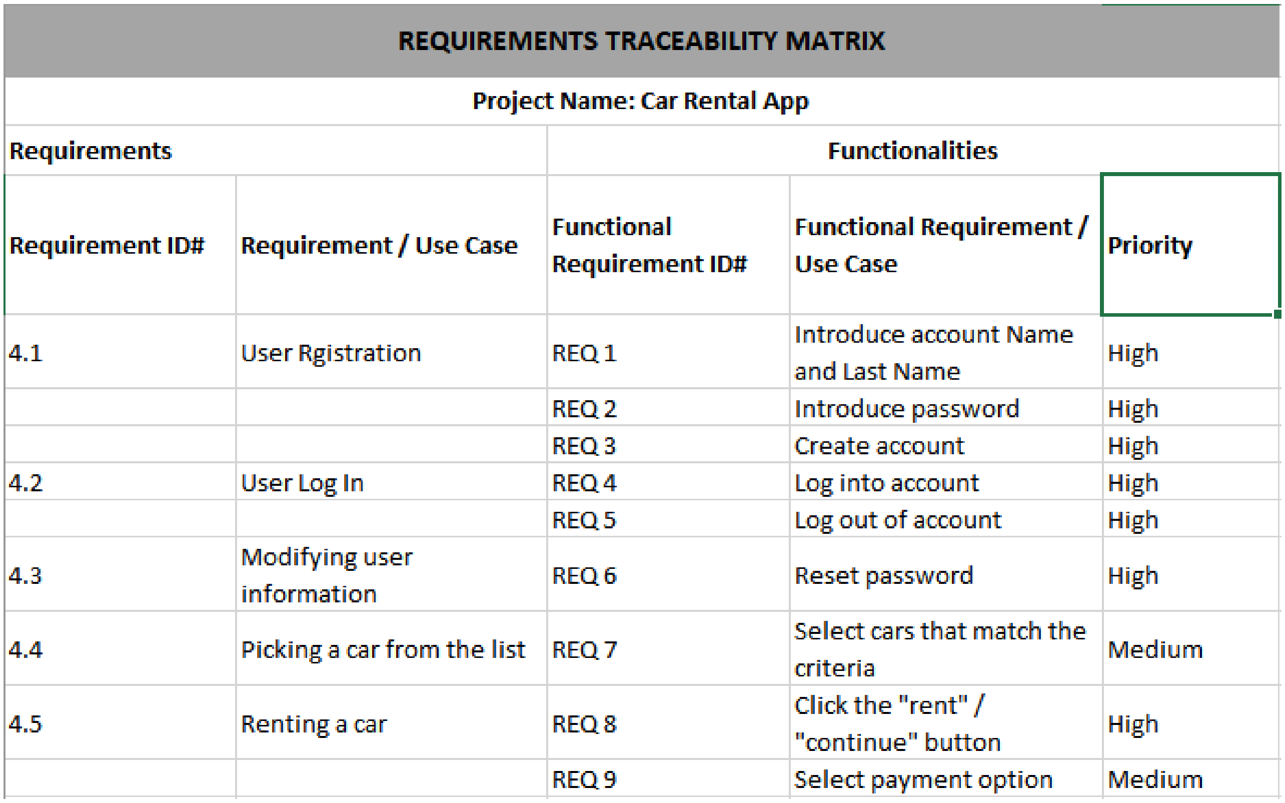



Register
Login
Price
Model
Seating Capacity
Description
Rent this Vechicle


Name
Email
Password
Confirm Password
Login


Email
Password
Register

7. REQUIREMENTS MATRIX



REQUIREMENTS TRACEABILITY MATRIX				
Project Name: Car Rental App				
Requirements		Functionalities		
Requirement ID#	Requirement / Use Case	Functional Requirement ID#	Functional Requirement / Use Case	Priority
4.1	User Rgistration	REQ 1	Introduce account Name and Last Name	High
		REQ 2	Introduce password	High
		REQ 3	Create account	High
4.2	User Log In	REQ 4	Log into account	High
		REQ 5	Log out of account	High
4.3	Modifying user information	REQ 6	Reset password	High
4.4	Picking a car from the list	REQ 7	Select cars that match the criteria	Medium
4.5	Renting a car	REQ 8	Click the "rent" / "continue" button	High
		REQ 9	Select payment option	Medium