**CAR RENTAL APP**

# Software Design Document

Ionica Mihai

Petrescu Alin

Nicolae Mihai

Software Engineering laboratory

March 2021

**TABLE OF CONTENTS**

1. [INTRODUCTION 2](#_bookmark0)
   1. [Purpose 2](#_bookmark1)
   2. [Scope 2](#_bookmark2)
   3. [Overview 2](#_bookmark3)
   4. [Reference Material 2](#_bookmark4)
   5. [Definitions and Acronyms 2](#_bookmark5)
2. [SYSTEM OVERVIEW 2](#_bookmark6)
3. [SYSTEM ARCHITECTURE 2](#_bookmark7)
   1. [Architectural Design 2](#_bookmark8)
   2. [Decomposition Description 3](#_bookmark9)
   3. [Design Rationale 3](#_bookmark10)
4. [DATA DESIGN 3](#_bookmark11)
   1. [Data Description 3](#_bookmark12)
   2. [Data Dictionary 3](#_bookmark13)
5. [COMPONENT DESIGN 3](#_bookmark14)
6. [HUMAN INTERFACE DESIGN 4](#_bookmark15)
   1. [Overview of User Interface 4](#_bookmark16)
   2. [Screen Images 4](#_bookmark17)
   3. [Screen Objects and Actions 4](#_bookmark18)
7. [REQUIREMENTS MATRIX 4](#_bookmark19)
8. [APPENDICES 4](#_bookmark20)

### INTRODUCTION

## 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.

## 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 of time.

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

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

## 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 -

## 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>

## 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

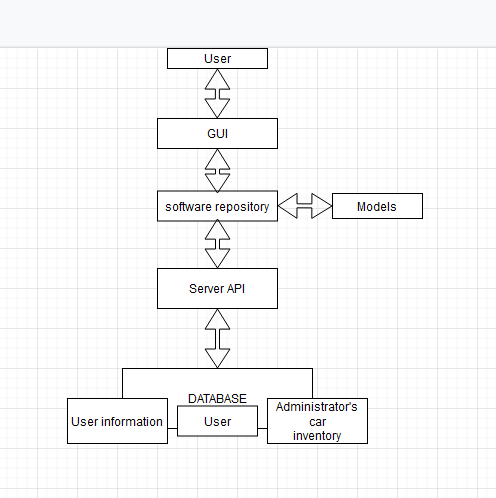
### SYSTEM OVERVIEW

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).

### SYSTEM ARCHITECTURE

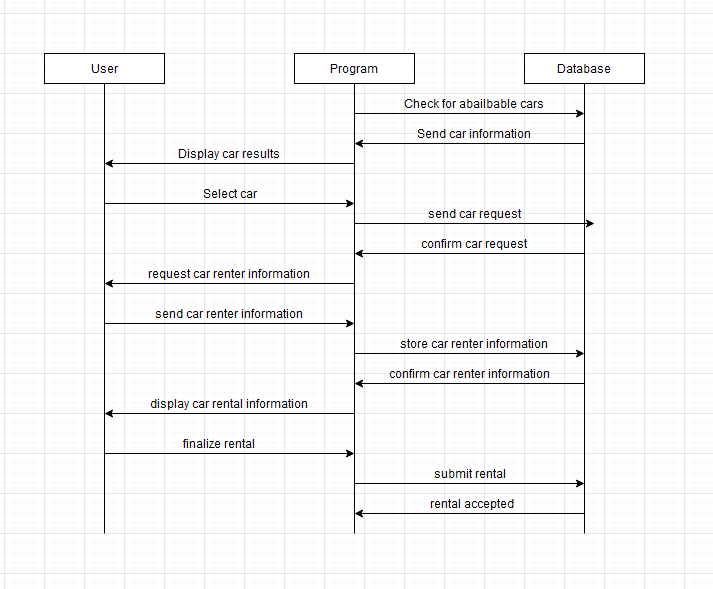
## Architectural Design

The interaction between different parts of the system:

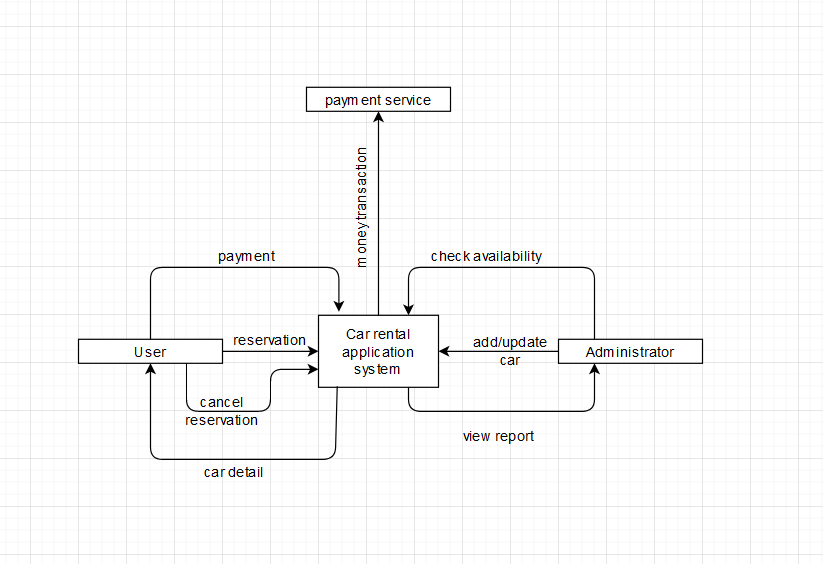


## Decomposition Description

Sequence diagram (Data flow) it presents the interaction that is made possible by the car rental system between the user and database:



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



## Design Rationale

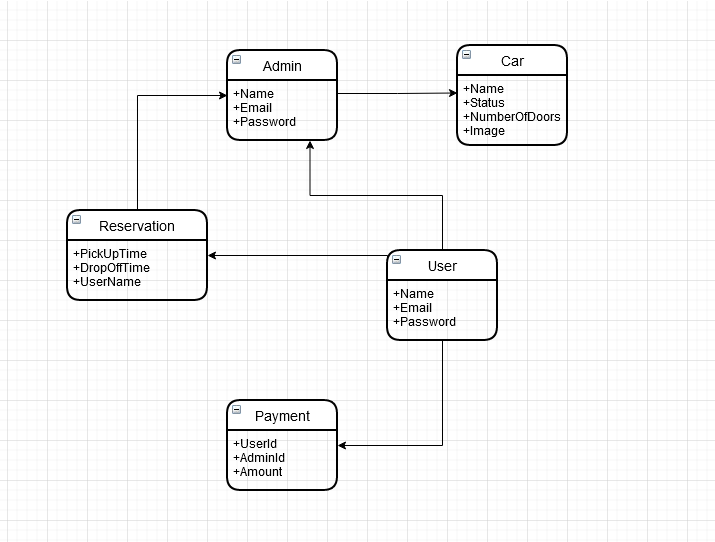
We choose a web architecture because it allows decentralized usage, and offers a easier maintenance over time. Having a single repository which chooses to get the data (as API calls to the server or as local models).

### DATA DESIGN

## 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 pick up 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).

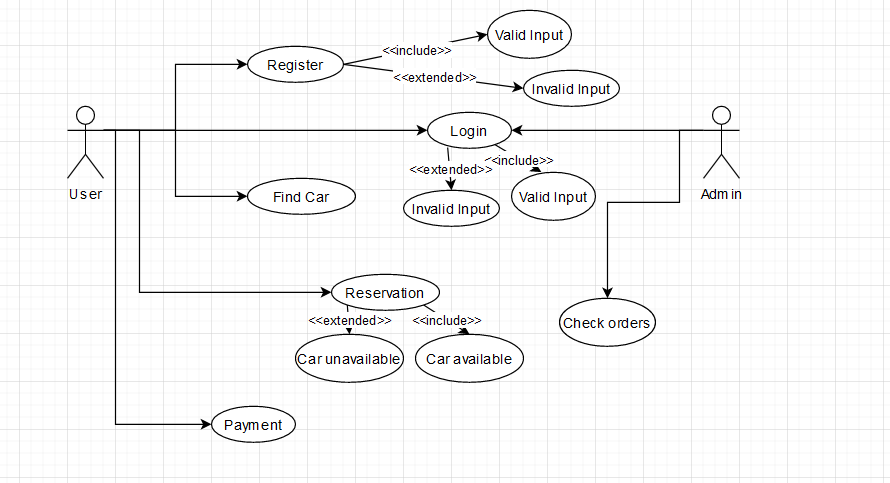


## Data Dictionary

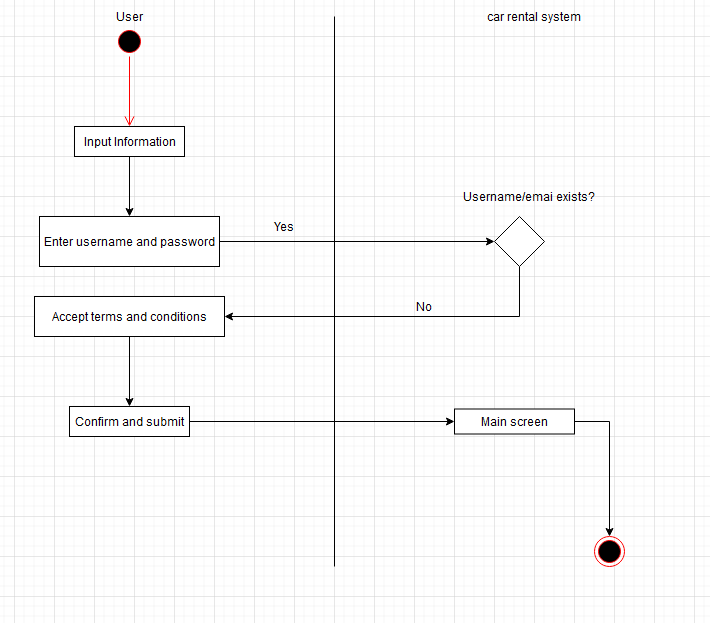
|  |  |  |
| --- | --- | --- |
|  | **Field** | **Type** |
| **User** | Name | varchar(32) |
|  | Password | varchar(32) |
|  | Email | varchar(64) |
|  |  |  |
|  |  |  |
| **Admin** | Name | varchar(32) |
|  | Email | varchar(64) |
|  | Password | varchar(32) |
|  |  |  |
| **Car** | Name | varchar(32) |
|  | Status | int |
|  | NumberOfDoors | int |
|  | Image | int |
|  |  |  |
| **Payment** | UserId | varchar(32) |
|  | AdminId | varchar(32) |
|  | Amount | Int |
|  |  |  |
| **Reservation** | PickUpTime | date |
|  | DropOffTime | date |
|  | UserName | varchar(32) |

### COMPONENT DESIGN

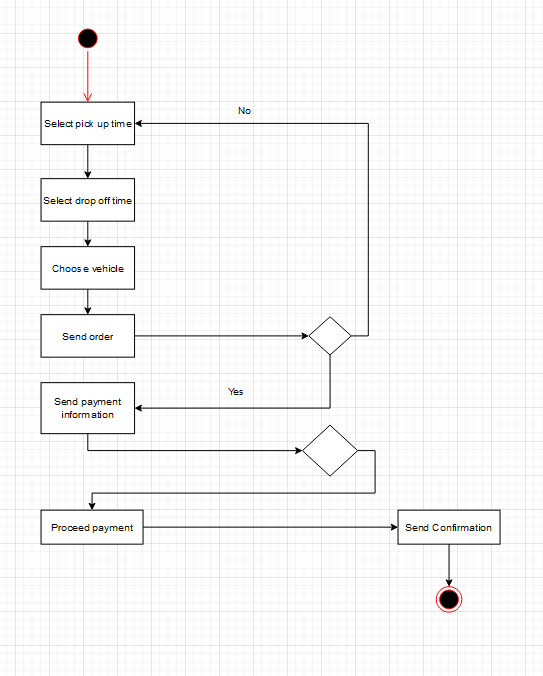
Use case:



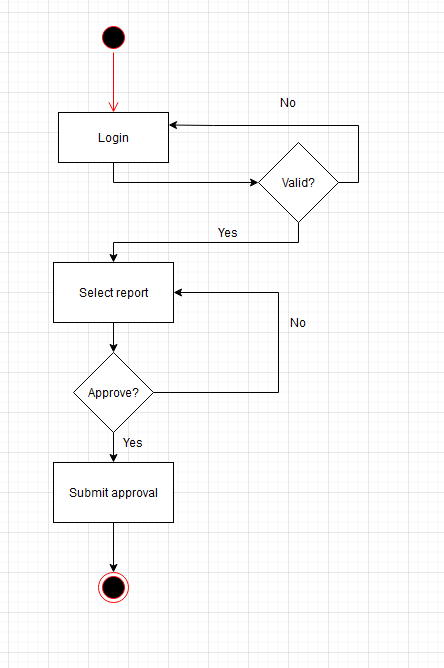
Sign up:



Car reservation:



Admin report:

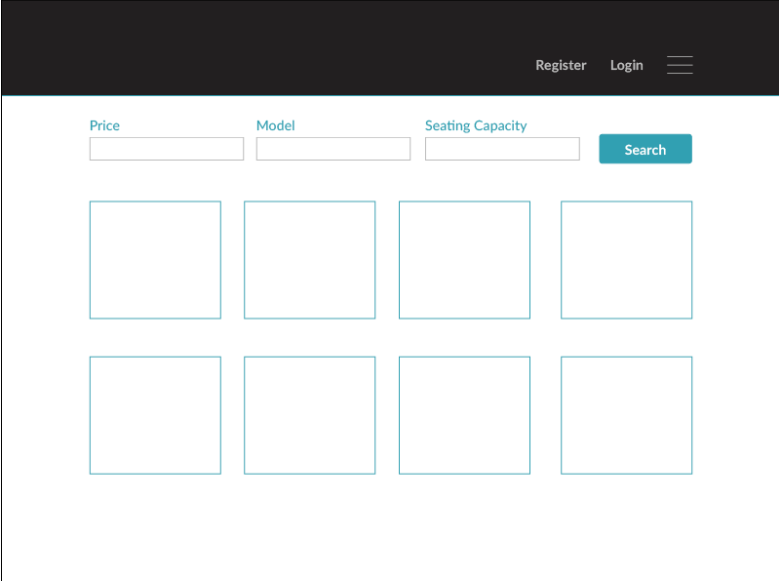


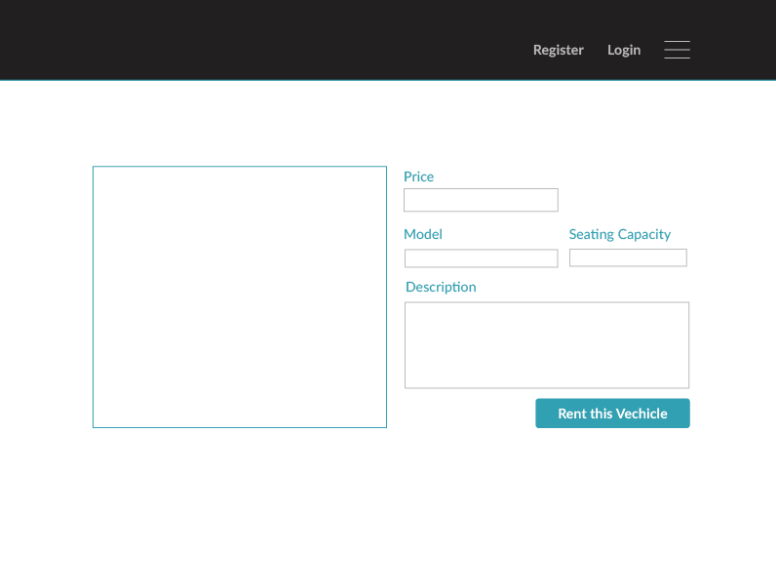
### HUMAN INTERFACE DESIGN

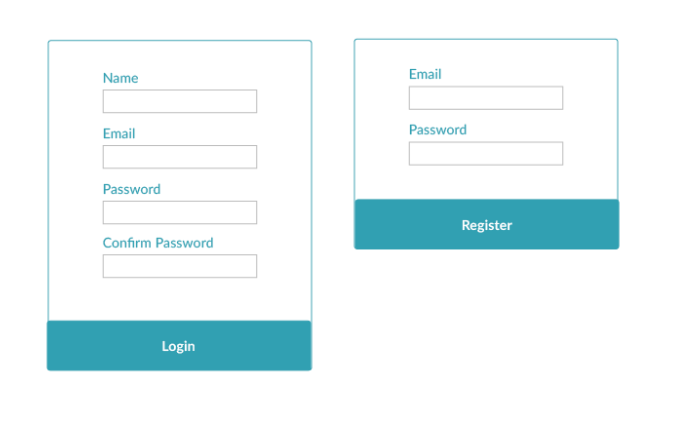
## Overview of User Interface

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.

## Screen Images







### REQUIREMENTS MATRIX