## CHAPTER 1

**INTRODUCTION**

**1.1 OVERVIEW OF THEPROJECT**

This application framework combines powerful inventory search features for both new and used car configuration with pricing information, separate photo galleries and side-by-side comparison tools. For supporting dealers, this system connects dealers with private-party sellers to buy new or used cars over the web.

Dealers can get register to the system's web site and can promote their dealership with enhanced listings, featuring photos and interactive links to their website. The registered dealers can receive responses from the site-surfers or sellers instantaneously by the information flowing in the system and can respond accordingly. In order to support sellers, this system facilitates sellers to place ads online for their used vehicle.

## CHAPTER 2

**SYSTEM ANALYSIS**

* 1. **EXISTINGSYSTEM**

In the existing scenario, if someone has to buy a car or even interested in buying any car, then the person has to go to the showroom. Not only once, but every time, whenever he/she is looking for some fact, the only option left is to go to the showroom. This is very time consuming and very hectic as well. Sometimes, the buyer even lost interest because of this practice.

**DISADVANTAGES**

* Can’t view the exact details in other sites
* Some fake sites are created in some different portal
* Cheating are may be happened.

## PROPOSEDSYSTEM

The proposed automobile system is very effective. If someone is interested in buying any car, then he/she can check all the information related to the car in the given portal. He/she can even book the test drive within the system. The proposed system also helps the buyer to check which cars and companies are good for them, by showing them the past reviews about the car/companies. The proposed system is so helpful and effective. Passenger.

**ADVANTAGES**

* Can Book a vehicle in a minute.
* Collect the all the information about the customer
* Very user-friendly to handle this

## FEASIBILITYSTUDY

The feasibility of the system is analyzed in this phase and business proposal is put for the with general plan for the project and cost estimates. During the system analysis of the project, the feasibility study of proposed system is to be carried out. For feasibility analysis, some understanding of the major requirements for the system is essential. Three key considerations involved in feasibility analysis are

* + - Technical Feasibility
    - Economic Feasibility
    - Operational Feasibility

## Technical Feasibility

Technical feasibility assesses the current resources (such as hardware and software) technology, which are required to accomplish user requirements in the software within the allocated time and budget. For this, whether the certain current resources and technology can be upgraded or added in the software to accomplish specified user requirements.

The technical requirements of the application are simple and basic. Python isused for the developers of the application and the framework is largely used by many, thus there will be enough support for future enhancements. The framework is stable and the support from the developers is constantly updated. The devices which have internet connectivity are enough for the application.

## Economic Feasibility

Economic feasibility determines whether the required software is capable of generating financial gains for an organization. It involves the cost incurred on the software development team, estimated cost of hardware and software, cost of performing feasibility study, and so on.

The cost of application development is very less and the cost of implementation is also less. It can be developed with the system with minimum requirements and can also be operated with the system with some basic requirements that are available the existing systems. For this, it is essential to consider expenses made on purchases and activities required to carry out software development.

## Operational Feasibility

Operational feasibility assesses the extent to which the required software performs a series of steps to solve user requirements. This feasibility is dependent on developer and involves visualizing whether the software will operate after it has been developed and be operative.

The application is developed based on the user requirements and is developed on the priority of the user requirements such as an integrated service and reviewing platform.

## PROBLEMANALYSIS

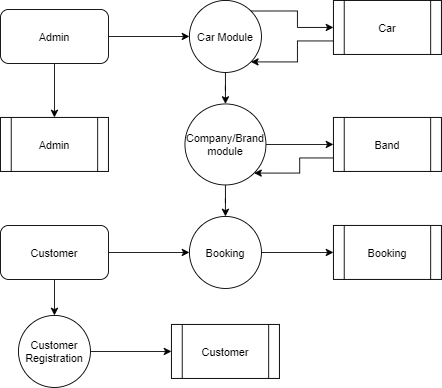
The orders from public are submitted through email, letter or form to the corresponding authority. The registered orders are forwarded to corresponding authority and report about the orders is submitted. The status of orders addressed is not recorded and maintained properly. The maps are not implemented. So, the customer does not know live location of the package. There is need for the software which receives the orders through online, forwarded to respective peoples for rectification and post the status of orders.

## CONTEXT AND DATA FLOWDIAGRAM

A data-flow diagram (DFD)is a way of representing a flow of a data of a process or system. The DFD also provides information about the outputs and inputs of each entity and process itself. A data-flow diagram is a part of structured-analysis modelling tools.



**LEVEL 1:**



**SYSTEMCONFIGURATION**

### Hardware Requirements

Processor : P 4 700 GHz

RAM Capacity : 4GB

Hard Disk : 180GB

### Software Requirements

Operating System : Windows 8,10

Front End : JAVA

Back End : SQL

### Software Descriptions

### Java

Java is a high-level programming language developed by Sun Microsystems. It was originally designed for developing programs for set-top boxes and handheld devices, but later became a popular choice for creating web applications.

The Java syntax is similar to C++, but is strictly an object-oriented programming language. For example, most Java programs contain classes, which are used to define objects, and methods, which are assigned to individual classes. Java is also known for being stricter than C++, meaning variables and functions must be explicitly defined. This means Java source code may produce errors or "exceptions" more easily than other languages, but it also limits other types of errors that may be caused by undefined variables or unassigned types.

Unlike Windows executables (.EXE files) or Macintosh applications (.APP files), Java programs are not run directly by the operating system. Instead, Java programs are interpreted by the Java Virtual Machine, or JVM, which runs on multiple platforms. This means all Java programs are multiplatform and can run on different platforms, including Macintosh, Windows, and Unix computers. However, the JVM must be installed for Java applications or applets to run at all. Fortunately, the JVM is included as part of the Java Runtime Environment (JRE),

### MySQL

MySQL is the popular Open-Source Relational SQL Database Management System. MySQL is being used for developing various web-based software applications. The MySQL development project has made its [source code](https://en.wikipedia.org/wiki/Source_code) available under the terms of the [General Public](https://en.wikipedia.org/wiki/GNU_General_Public_License) [License.](https://en.wikipedia.org/wiki/GNU_General_Public_License) It is used to store the information.

MySQL was owned and sponsored by the single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation. For proprietary use, several paid editions are available, and offer additional functionality.

## CHAPTER 3

**SYSTEM DESIGN**

**3.1 INPUT DESIGN**

Input Design is the process of converting a user-oriented description of the input into a computer-based system. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. Input Design is the first phase in the system design. Input designing is to converting the user-oriented information to the computer- oriented form. The input data items are grouped and analyzed to find out whether the proposed system can be developed from the user input. The system is developed using various processes screens formats.

The main objective of input design is to

* + - It should serve specific purpose effectively such as storing, recording, and retrieving the information.
    - It ensures proper completion with accuracy.
    - It should be easy to fill and straightforward.
    - It should focus on user’s attention, consistency and simplicity.

### DATABASE DESIGN

The most important consideration in designing the database is how the information will be used. The main objective of designing a database is Data Integration, Data Integrity and Data Independence.

### Data Integration

In a database, information from several files is coordinated, accessed and operated upon as through it is in a single file. Logically, the information is centralized, physically, the data may be located on different devices, connected through data communication facilities.

### Data Integrity

Data integrity means storing all data in one place only and how each application access it. This approach results in more consistent information, one update being sufficient to achieve a new record status for all applications. This leads to less data redundancy, that is data items need not be duplicated. A reduction in the direct access storage requirement.

### Data Independence

Dataindependenceistheinsulationofapplicationprogramsfromchangingaspects of physical data organization. This objective seeks to allow changes in the content and organizationofphysicaldatawithoutreprogrammingofapplicationandallowmodifications to application programs without reorganizing the physical data.

### 3.2.1 TABLE DESIGN

The table needed for each module were designed and the specification of each and every column was given based on the records and details collected during record specification of the system study.

**TABLE NAME: ADMIN**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD** | **DATA TYPE** | **SIZE** | **CONSTRAINT** |
| Admin id | Int | 10 | Primary key |
| Username | Varchar | 30 | Not null |
| password | Varchar | 30 | Not null |

**TABLE NAME: BRAND**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD** | **DATA TYPE** | **SIZE** | **CONSTRAINT** |
| Brand id | Int | 10 | Primary key |
| Brand name | Varchar | 20 | Not null |
| Country | Varchar | 10 | Not null |
| Model | Varchar | 15 | Not null |
| Publishing year | Int | 4 | Not null |

**TABLE NAME: CAR**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD** | **DATA TYPE** | **SIZE** | **CONSTRAINT** |
| Carid | Int | 10 | Primary key |
| Brand id | Int | 10 | Foreign key |
| Car name | Varchar | 10 | Not null |
| Model | Varchar | 10 | Not null |
| Engine type | Varchar | 15 | Not null |
| Width | Varchar | 10 | Not null |
| Height | Varchar | 10 | Not null |
| Price | Int | 5 | Not null |
| Published year | Int | 4 | Not null |

**TABLE NAME: CUSTOMER**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD** | **DATA TYPE** | **SIZE** | **CONSTRAINT** |
| customer id | Int | 10 | Primary key |
| First name | Varchar | 20 | Not null |
| Last name | Varchar | 20 | Not null |
| Email id | Varchar | 20 | Not null |
| Password | Varchar | 20 | Not null |
| Mobile number | Int | 10 | Not null |

**TABLE NAME: BOOKING**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD** | **DATA TYPE** | **SIZE** | **CONSTRAINT** |
| booking id | Int | 10 | Primary key |
| Customer id | Int | 10 | Foreign key |
| Model id | Int | 10 | Foreign key |
| Car id | Int | 10 | Foreign key |
| Price | Int | 5 | Not null |
| Expected price | Int | 5 | Not null |
| suggestions | Varchar | 30 | Not null |

**MODULEDESCRIPTION**

The main module in this project are listed below

* + - CAR MODULE
    - COMPANY MODULE
    - BOOKING MODULE
    - CUSTOMER MODULE

**1.CAR MODULE**

All the facts and figures related to a car of each model is present in this module & the viewer can view this module after login into the system. New car details and existing car details are added into this module. Lists of all the cars are available inside this module.

**2.** **COMPANY MODULE**

All the details of the company are added to the module. Along with the existing details, new details are also added into this module about the company and various new companies are being updated by the admin of the system, frequently.

**3. BOOKING MODULE**

Via this module, the user or customer of this system can book the cars for them. If they are not willing to book the car, then they can at least book a test drive for themselves. All the previous booking details are available through this module. Admin can update the details as per the need.

**4. CUSTOMER MODULE**

This module is designed specifically for the customers. They can log in into the system after getting their credentials. This module keeps track of each and every customer and all the information related to each customer is being added into this module. This module is updated timely by the admin of this system.

## CHAPTER 4 SYSTEMTESTING

Testingisanintegralpartofanysystemdevelopmentlifecycle.Insufficient and untested applications may tend to crash and the result is loss of economic and manpower investment besides user's dissatisfaction and downfall of reputation. Software testing can be looked upon as one among many processes, an organization performs, and that provides the lost opportunity to correct any flaws in the developed system. Software testing includes selecting test data that have more probability of giving errors.

The first step in system testing is to develop a plan that tests all aspects of the system. Completeness, correctness, reliability and maintainability of the software aretobetestedforthebestqualityassurancethatthesystemmeetsthespecificationand requirements for its intended use and performance. System testing is the most useful practical process of executing a program with the implicit intention of finding errors that make the program fails. System testing is done in three phases.

* + - * Unit Testing
      * Integration Testing
      * Validation Testing

### UNIT TESTING

Unit testing focuses verification effort on the smallest unit of software the module. Using the detailed design and the process specification testing is done to registration by the user with in the boundary of the Login module. The login form receives the username and password details and validates the value with the database. If valid, the home page is displayed.

### INTEGRATIONTESTING

Integration Testing is the process of this activity can be considered as testing the design and hence module interaction. The primary objective of integration testing is to discover errors in the interfaces between the components. Login form and registration form are integrated and tested together. If the user is newly registered, the received details will be stored in the registration table. While logging in, the application will check for valid user name and password in the registration table and if valid the user is prompted for submitting complaints.

### VALIDATIONTESTING

Validation are independent procedures that are used together for checking that a product, service, or system meets [requirements](https://en.wikipedia.org/wiki/Requirement) and [specifications](https://en.wikipedia.org/wiki/Specification_(technical_standard)) and that it fulfills its in purpose the actual result from the expected result for the complaint process. Select the complaint category of the complaint by user. The input given to various forms fields are validated effectively. Each module is tested independently. It is tested that the complaint module fields receive the correct input for the necessary details such as complaint category, complaint id, reference name, complaint description, email for further process.

## CHAPTER 5

* 1. **CONCLUSION**

This project provides an opportunity for the user can booking their favorite car in this application. This application will support to display the car details and model cars to an single application. Admin register the car details and registered the cars are displayed the user portal. Also the user can book the registration.

## APPENDICES APPENDIX I SAMPLE CODE

$(document).ready(function () {

$("#register").on('click', function () {

var one = $("#1").val();

var two = $("#2").val();

var three = $("#3").val();

var four = $("#4").val();

var five = $("#5").val();

var six = $("#6").val();

var seven = $("#7").val();

var eight = $("#8").val();

$.ajax({

type: "GET",

url: "http://localhost:8080/api/add\_car/" + encodeURIComponent(one) + "/" + two+ "/" + three+ "/" + four+ "/" + five+ "/" + six+ "/" + seven+ "/" + eight

}).done(function (data) {

$("#1").val("");

$("#2").val("");

$("#3").val("");

$("#4").val("");

$("#5").val("");

$("#6").val("");

$("#7").val("");

$("#8").val("");

alert(date)

});

});

});

function booking\_history(){

$.ajax({

type: "GET",

url: "http://localhost:8080/api/book\_histoy"

}).done(function (data) {

for(var i=0;i<data.length;i++){

var row=data[i];

html += ` <tr><th scope="row">`+(i+1)+`</th>

<td>`+row[1]+`</td>

<td>`+row[2]+`</td>

<td>`+row[3]+`</td>

<td>`+row[4]+`</td>

<td>`+row[5]+`</td>

<td>`+row[6]+`</td>

<td>`+row[7]+`</td>

<td>`+row[8]+`</td>

</tr>`;

}

$("#body1").html(html);

});

}

function car\_details(){

$.ajax({

type: "GET",

url: "http://localhost:8080/api/car\_details"

}).done(function (data) {

for(var i=0;i<data.length;i++){

var row=data[i];

html += `<div class="card" style="width: 18rem;">

<img class="card-img-top" src="./car.jpg" alt="Card image cap">

<div class="card-body" >

<h5 class="card-title">xxxxxxxx</h5>

<p class="card-text">xxxxxxxxxxxxxxxxxxxxxxxxx</p>

</div>

<ul class="list-group list-group-flush">

<li class="list-group-item">xxxxxxxxxxxxxxxxx</li>

<li class="list-group-item">xxxxxxxx</li>

<li class="list-group-item">xxxxxxxx</li>

</ul>

<div class="card-body">

<a href="#" class="card-link">xxxxxxx</a>

<a href="#" class="card-link">xxxxx</a>

</div>

</div>`;

}

$("#body2").html(html);

});

}

function loading(){

booking\_history();

car\_details();

}

$(document).ready(function () {

$("#booking").on('click', function () {

var one = cid;

var two = cusid;

$.ajax({

type: "GET",

url: "http://localhost:8080/api/book\_car/" + encodeURIComponent(one) + "/" + two

}).done(function (data) {

alert(date)

});

});

});

function car\_details(){

$.ajax({

type: "GET",

url: "http://localhost:8080/api/car\_details"

}).done(function (data) {

for(var i=0;i<data.length;i++){

var row=data[i];

html += `<div class="card" style="width: 18rem;">

<img class="card-img-top" src="./car.jpg" alt="Card image cap">

<div class="card-body" >

<h5 class="card-title">xxxxxxxx</h5>

<p class="card-text">xxxxxxxxxxxxxxxxxxxxxxxxx</p>

</div>

<ul class="list-group list-group-flush">

<li class="list-group-item">xxxxxxxxxxxxxxxxx</li>

<li class="list-group-item">xxxxxxxx</li>

<li class="list-group-item">xxxxxxxx</li>

</ul>

<div class="card-body">

<a href="#" class="card-link">xxxxxxx</a>

<a href="#" class="card-link">xxxxx</a>

</div>

</div>`;

}

$("#body2").html(html);

});

}

function loading(){

car\_details();

}

$(document).ready(function () {

$("#register").on('click', function () {

var one = $("#1").val();

var two = $("#2").val();

var three = $("#3").val();

var four = $("#4").val();

var five = $("#5").val();

// var six = $("#6").val();

// var seven = $("#7").val();

// var eight = $("#8").val();

$.ajax({

type: "GET",

url: "http://localhost/2020/AGSC/CAR%20SHOP/add\_customer.php?fname=" + encodeURIComponent(one) + "?lname=" + two+ "?emailid=" + three+ "?password=" + four+ "?mobile=" + five

}).done(function (data) {

$("#1").val("");

$("#2").val("");

$("#3").val("");

$("#4").val("");

$("#5").val("");

});

});

});

<html>

<head>

<link rel="stylesheet" href="./style.css" />

<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.10.2/jquery.min.js"></script>

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css"

integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm" crossorigin="anonymous">

<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js"

integrity="sha384-JZR6Spejh4U02d8jOt6vLEHfe/JQGiRRSQQxSfFWpi1MquVdAyjUar5+76PVCmYl"

crossorigin="anonymous"></script>

</head>

<body class="bodys">

<div class="tabs">

<input type="radio" name="tabs" checked="checked" id="toggle-cherries" />

<label for="toggle-cherries">

<span class="text-title">Car Details</span>

</label>

<input type="radio" name="tabs" id="toggle-oranges" />

<label for="toggle-oranges">

<span class="text-title">Logout</span>

</label>

<div class="tab tab-cherries">

<div class="card" style="width: 18rem;">

<img class="card-img-top" src="./car.jpg" alt="Card image cap">

<div class="card-body">

<h5 class="card-title">xxxxxxxx</h5>

<p class="card-text">xxxxxxxxxxxxxxxxxxxxxxxxx</p>

</div>

<ul class="list-group list-group-flush">

<li class="list-group-item">xxxxxxxxxxxxxxxxx</li>

<li class="list-group-item">xxxxxxxx</li>

<li class="list-group-item">xxxxxxxx</li>

</ul>

<div class="card-body">

<button type="submit" class="btn btn-primary">Book</button>

</div>

</div>

</div>

</div>

</body>

<script>

</script>

</html>

<html>

<head>

<link rel="stylesheet" href="./style.css" />

<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.10.2/jquery.min.js"></script>

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css"

integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm" crossorigin="anonymous">

<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js"

integrity="sha384-JZR6Spejh4U02d8jOt6vLEHfe/JQGiRRSQQxSfFWpi1MquVdAyjUar5+76PVCmYl"

crossorigin="anonymous"></script>

</head>

<body class="bodys">

<div class="tabs">

<input type="radio" name="tabs" id="toggle-apples" checked="checked" />

<label for="toggle-apples">

<span class="text-title">Car Registration</span>

</label>

<input type="radio" name="tabs" id="toggle-bananas" />

<label for="toggle-bananas">

<span class="text-title">Booking History</span>

</label>

<input type="radio" name="tabs" id="toggle-cherries" />

<label for="toggle-cherries">

<span class="text-title">Car Details</span>

</label>

<input type="radio" name="tabs" id="toggle-oranges" />

<label for="toggle-oranges">

<span class="text-title">Logout</span>

</label>

<div class="tab tab-apples">

<div class="form-group">

<label>Branch Name</label>

<input class="form-control" id="1">

</div>

<div class="form-group">

<label>Car Name</label>

<input class="form-control" id="2">

</div>

<div class="form-group">

<label>Model</label>

<input class="form-control" id="3">

</div>

<div class="form-group">

<label>Engine Type</label>

<input class="form-control" id="4">

</div>

<div class="form-group">

<label>Width</label>

<input class="form-control" id="5">

</div>

<div class="form-group">

<label>Height</label>

<input class="form-control" id="6">

</div>

<div class="form-group">

<label>Price</label>

<input class="form-control" id="7">

</div>

<div class="form-group">

<label>Publiched Year</label>

<input class="form-control" id="8">

</div>

<button id="register1" type="submit" class="btn btn-primary">Car Registration</button>

</div>

<div class="tab tab-bananas">

<table class="table table-bordered">

<thead>

<tr>

<th scope="col">#</th>

<th scope="col">Customer Name</th>

<th scope="col">Contact Number</th>

<th scope="col">Car Name</th>

<th scope="col">Model</th>

<th scope="col">Price</th>

<th scope="col">Expected Price</th>

<th scope="col">Suggestion</th>

</tr>

</thead>

<tbody id="body1">

</tbody>

</table>

</div>

<div class="tab tab-cherries" id="body2">

</div>

</div>

</body>

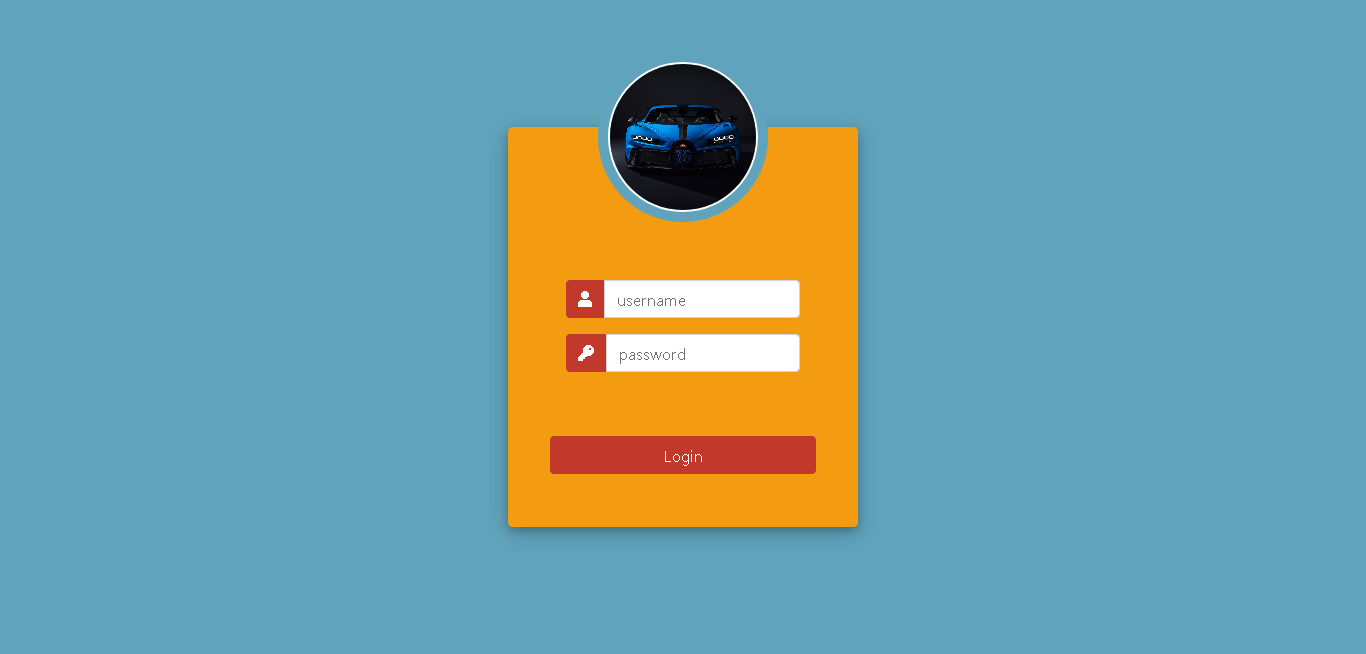
<script src="./admin-home.js">

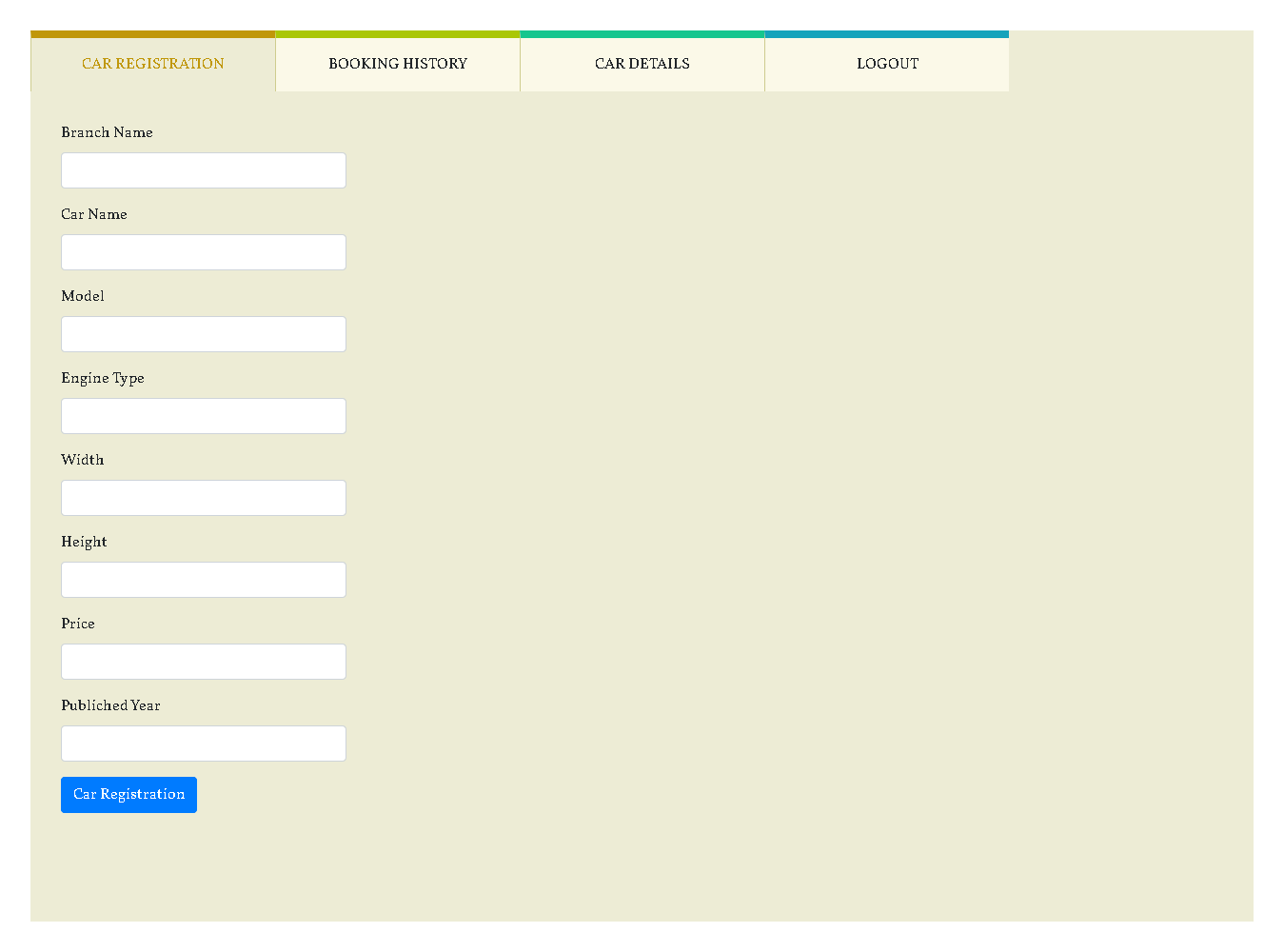
</script>

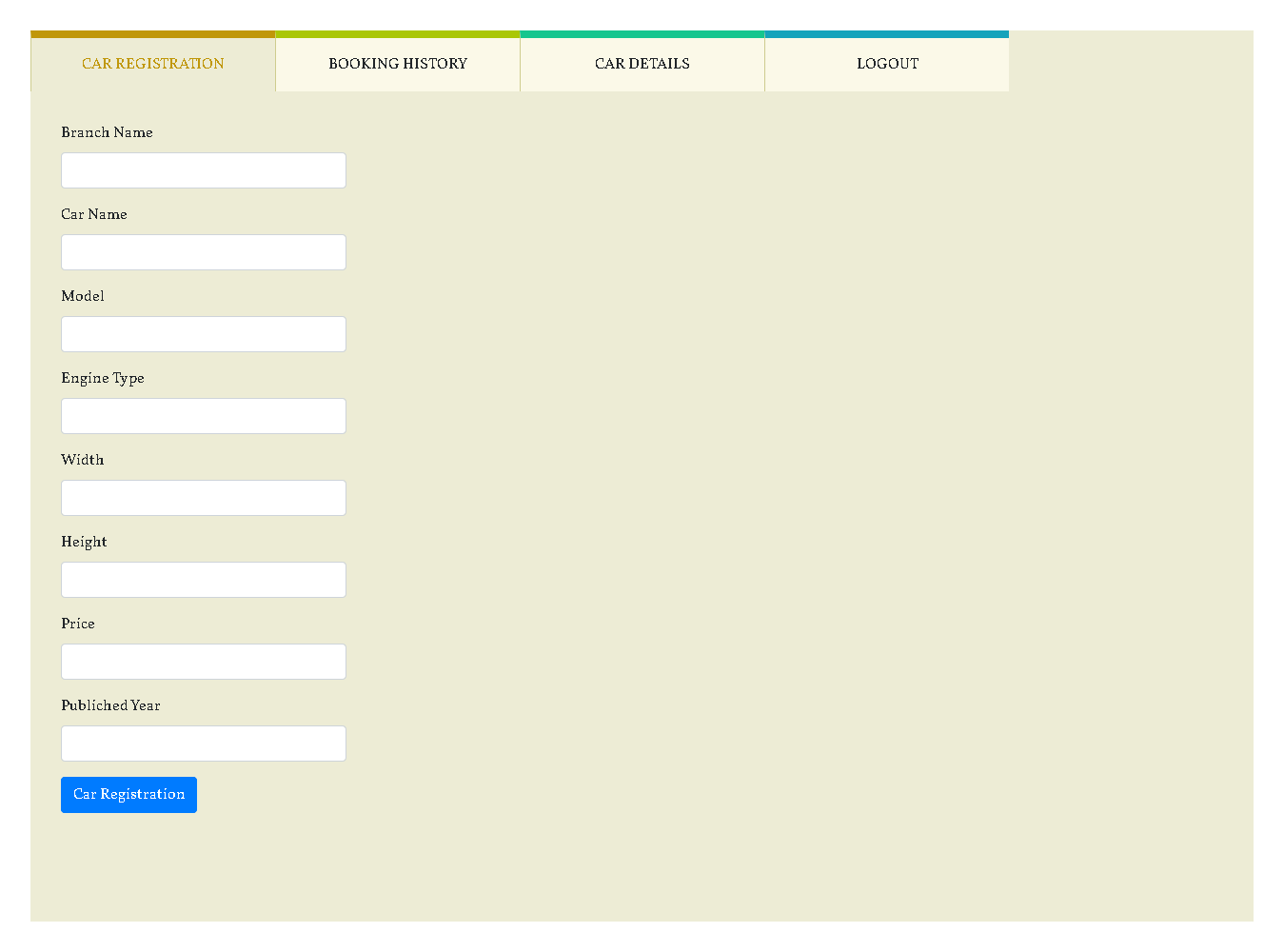
</html>

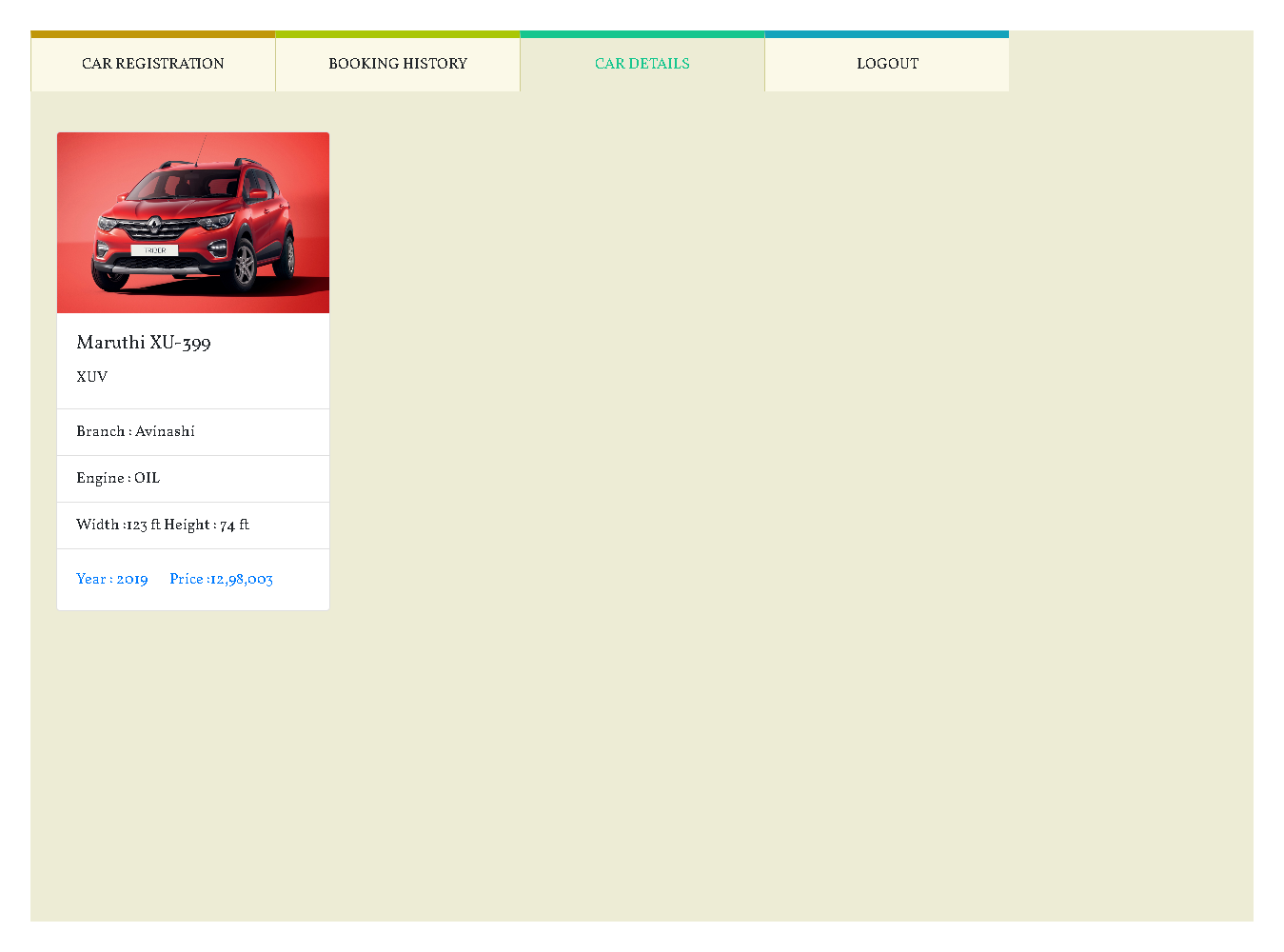
## APPENDIX II

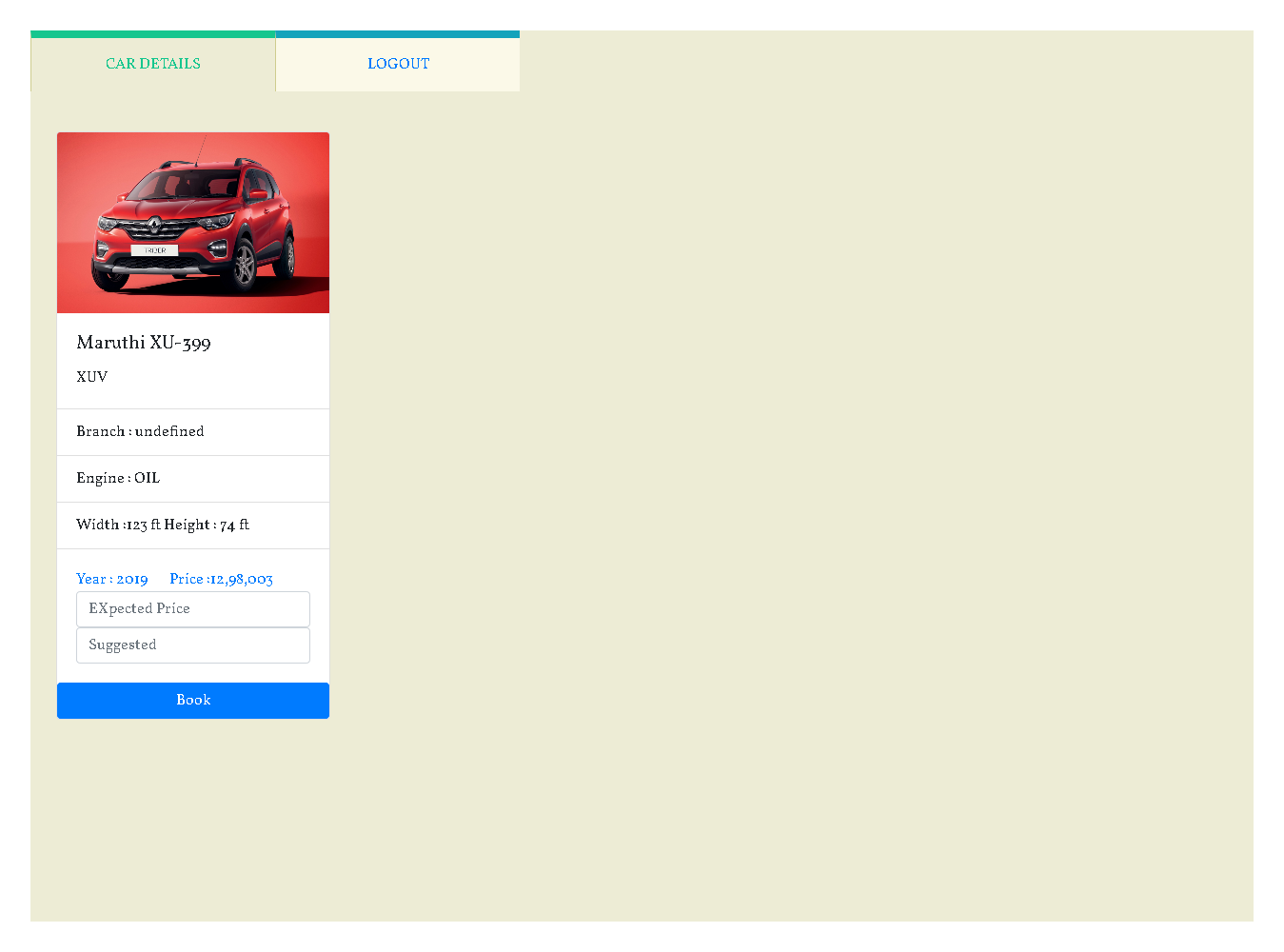
**SAMPLE SCREENSHOTS**











**REFERENCES**

**BOOK REFERENCES**

1. Jeff Forcier, Paul Bissex, Wesley J. Chun, (2018) “Python Web development with Django (Developer’s Library)”, WesleyProfessional.
2. Roger S. Pressman, (2017) “Software Engineering”, Tata McGraw-Hill –Europe.

## WEB REFERENCES

1. https://[www.javaprogramming.net](http://www.javaprogramming.net)
2. https://[www.w3schools.com](http://www.w3schools.com/)
3. https://[www.tutorialspoint.com](http://www.tutorialspoint.com/)