Software Requirements Specification

For

Car Reselling Website Website

Version 1.3.0 approved

Prepared by:- Mohit Saini

Priyaj Sony

Mohit Saini

Organization:- P.I.E.T

Date:- 20-Nov-2023

Submitted in partial fulfilment
Of the requirements of
3CDS4-07 Software Engineering

A

SOFTWARE REQUIREMENT SPECIFICATION

ON

Car Reselling Website

UNDER

NON SYLLABUS PROJECT

Submitted To:

Ms. Surbhi Saxena

(Assistant Professor)

Submitted by:

Mohit Saini (PIET22CD044)

Priyaj Sony (PIET22CD048)

Yash Saini (PIET22CD058)

DEPARTMENT OF ARTIFICIAL INTELLIGENCE & DATA SCIENCE
POORNIMA INSTITUTE OF ENGINEERING & TECHNOLOGY, JAIPUR
(Academic Year 2023-24)(Odd)

DECLARATION

I hereby declare that the Non syllabus Project report entitled "**CAR RESELLING WEBSITE** " was carried out and written by me under the guidance of Ms. Surbhi Saxena, Assistant Professor, Department of Computer Engineering, Poornima Institute of Engineering & Technology, Jaipur. This work has not been previously formed the basis for the award of any degree or diploma or certificate nor has been submitted elsewhere for the award of any degree or diploma.

Place: Jaipur, Rajasthan

Student Name : Mohit Saini

Date: 20 November, 2023

Reg. No. : PIET22CD044

Revision History

Date	Version	Description	Author
<20/11/23>	<1.0>	SRS 1.0.0	Author-1
<21/11/23>	<2.0>	SRS 1.1.0	Author-1
<24/11/23>	<3.0>	SRS 1.2.0	Author-1
<24/12/23>	<4.0>	SRS 1.3.0	Author-1
<04/01/24>	<5.0>	SRS 1.4.0	Author-1

List of Tables

Table of Contents

CHAPTER 1: INTRODUCTION OF PROJECT

Welcome to our Car Reselling Website –FASTLANE– a web-based solution designed for streamlined and secure car-buying experiences. This project is tailored for administrators seeking efficient online management, featuring a user-friendly interface that facilitates quick understanding. Versatile in its application, our platform is an ideal choice for dealerships and individuals alike, providing a straightforward approach to cataloging and managing car records. Boasting simplicity, speed, and accuracy, it optimises functionality while keeping storage needs to a minimum. In the digital age, our website taps into the widespread accessibility of the Internet, reshaping the car reselling landscape with user-friendly, efficient processes. Explore a platform where ease of use meets the sophistication of online car transactions.

1. Objective of Project

The objective of our Car Reselling Website project is to establish a user-friendly and efficient online platform for the buying and selling of used cars. This includes providing administrators with seamless management tools, ensuring a straightforward experience for both buyers and sellers, and optimising overall functionality. The platform aims to cater to diverse users, including dealerships and individuals, offering a secure and intuitive cataloging system for car records. With a focus on simplicity, speed, and accuracy, the project seeks to revolutionise the digital landscape of car transactions, contributing to the ongoing transformation of the automotive industry. Enhanced security measures and comprehensive vehicle listings further support the goal of creating a trustworthy and dynamic online space for car reselling

2. Types of Users

- User: Administrator

Functions: The Administrator is the super user and has complete control over all the activities that can be performed. The application notifies the administrator of all car buying/selling requests, and the administrator can then approve or reject them. The administrator also manages the list of available car models/brands. The administrator can also view and delete stock database.

- User: Service Facilitator

Functions: An employee can submit a buy/sell request through the application. When the request is approved by the Administrator, the requester is notified, and from there on is given the role of Service Facilitator. The Service Facilitator is responsible for setting up the Services and maintaining all the related data. This job involves managing the car dealings and transactions. The Service Facilitator can also decide to remove the listing from the site.

- User: Customer/Guests

Functions: A Customer can browse through the listed car categories. Therein he/she can select a specific car listing and view the detailed information where it is available with price. To proceed with the dealing, the customer is prompted to login. Also, the customer can modify personal profile information (such as phone number and e-mail address) stored by the application. The customer can also view the status of any previous dealings, and cancel any transaction that has not commenced yet.

3. Constraints and Dependency

Constraints

- ➢ One user can login within a system at a time.
- ➢ Regulatory policies
- ➢ Interface to other applications
- ➢ Parallel operation
- ➢ Audit functions
- ➢ Control functions
- ➢ Safety and security considerations
- ➢ Report format
- ➢ Data naming
- ➢ Accounting procedures
- ➢ Audit Tracing

Dependencies

- The customer should know how to use the Graphical User Interface (GUI).
- The employee should provide timely information regarding current dealings.
- Customer should enable E-Banking facilities or should have Credit / Debit Cards.

4. Methodology Used or Project Lifecycle:-

The Waterfall Model is a linear sequential flow. In which progress is seen as flowing steadily downwards (like a waterfall) through the phases of software implementation. This means that any phase in the development process begins only if the previous phase is complete. The waterfall approach does not define the process to go back to the previous phase to handle changes in requirement. The waterfall approach is the earliest approach that was used for software development.

CHAPTER 2: REQUIREMENT ANALYSIS

1. Functional Requirement

Initial functional requirements will be: -

- **Secure registration and profile management facilities for Customers.**
- *Adequate searching mechanisms for easy and quick access to particular Car model/brand listings and their availability.*
- *Regular updates to registered customers of the ORACLE SMALL BUSINESSES SUITE (OSBS) about new availabilities.*
- *Notifying about hot selling cars in their genre.*
- Strategic data and graphs for Administrators and Service Team Members about the cars that are popular in each category and brand
- Maintaining database of regular customers of different needs.
- Feedback mechanism, so that customers can give feedback.
- **Adequate payment mechanism and gateway for all popular credit cards, cheques and other relevant payment options, as available from time to time.**

For the previous paragraph, depicting the functions of the system, from the perspective of the various users of the system, the following text codes have been used:

- BOLD for administrator
- ITALICS for customer
- UNDERLINED for the employees.

2. Non-Functional Requirement

Initial non functional requirements will be: -

- Secure access of confidential data (user's details). Secure Socket Layers (SSL) can be used.
- 24 X 7 availability.
- Better component design to get better performance at peak time.
- Advertisement space where it will effectively catch the customer's attention and as a source of revenue.

In addition to the above mentioned points, due to the highly evolving nature of the project, the following are planned to be delivered if deemed necessary: -

- More payment gateways.
- Dynamic price model by which prices can be changed based on demand and supply.
- Dynamic Storefront: Each customer will have a web page personalized based on his or her recent purchases. This is the equivalent of having a unique storefront for each customer in hopes of drawing in as many return customers as possible.

3. Technology Used

- HTML
- CSS
- JAVASCRIPT
- BOOTSTRAP

4. H/w Configuration

System will consist of mainly two major components, a back-end platform and a front-end application. Customized applications can be built on top of the backend platform to cater different user requirements.
Back-end platform will be deployed in one of the network operator's servers. Front-end application may also be deployed in the same server or another remote server. Users will be able access the front-end application through web using desktop machines and through WAP using mobiles, PDAs or any other WAP enabled device. Not only that, any mobile device having the STK application capabilities should be able to use the product.

5. Graphical User Interface

The system shall provide a uniform look and feel between all the web pages.
The system shall provide a digital image for each car when the user chooses to see more details.
The system shall provide use of icons and toolbars.

6. Advantages

- Saves time (availability of all popular cars at a single platform)
- Easy to access the system anywhere and anytime.

7. Disadvantages

- Requires an active internet connection.
- Cars might get sold out soon due to limited resellers online

8. Applications

- This application can be used by showrooms , brands for expanding their business online.

9. Product perspective

OSBS is aimed towards the vendors who want to reach out to the maximum cross-section of customer and common people who can be potential customer. This project envisages bridging the gap between the seller, the retailer and the customer. OSBS should be user-friendly, 'quick to learn' and reliable software for the above purpose. OSBS is intended to be a stand-alone product and should not depend on the availability of other software. It should run on multiple platforms smoothly like Windows, Android, iOS, etc.

A Car Database system stores the following information.

- **Car details: -** It includes car name , model , brand name , engine specification , reg. state , kms. Driven , condition , price etc.
- **Customer description: -** It includes customer code, name, address and phone number. This information may be used for keeping the

records of the customer for smooth dealings or for any other kind of information

- **Reservation description: -** It includes customer details, code number, car booking number, date of booking.

10. System Product Functions

Mobile Billing will be providing the below mentioned main functions.

The followings are the summarized version of the significant functions of the system. A finer description about those functions is further given below.

Billing

Specifications of billing mechanisms will be found here. The pending bill payment will be done using this function.

11. Advanced Features of the System. User Classes and Characteristics

- e-bill : Payment of bill will be done on a single click.

- Duplicate bill: A duplicate copy of the bill can be downloaded using this function.

- Pay w/o login: Payment can be done without logging on to any account, it can be done simply with a mobile number.

12. Operating Environment

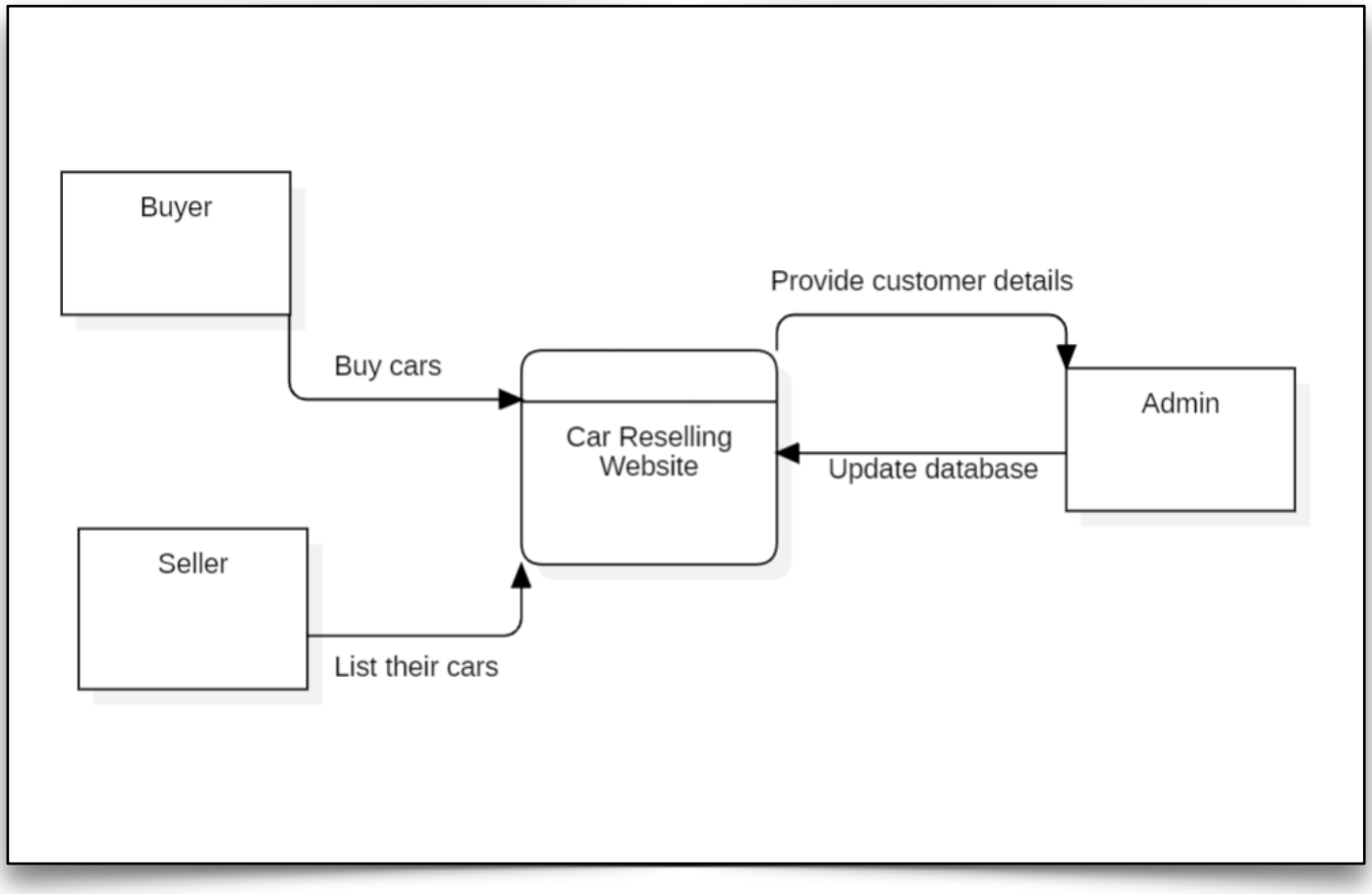
Hardware Platform: Smartphone's, PCs, Tablets, etc.
Software Components : Windows XP or more, Android 2.3 or more, iOS 4.0 or more, Firefox 21.0 or more, etc

CHAPTER 3: DESIGN

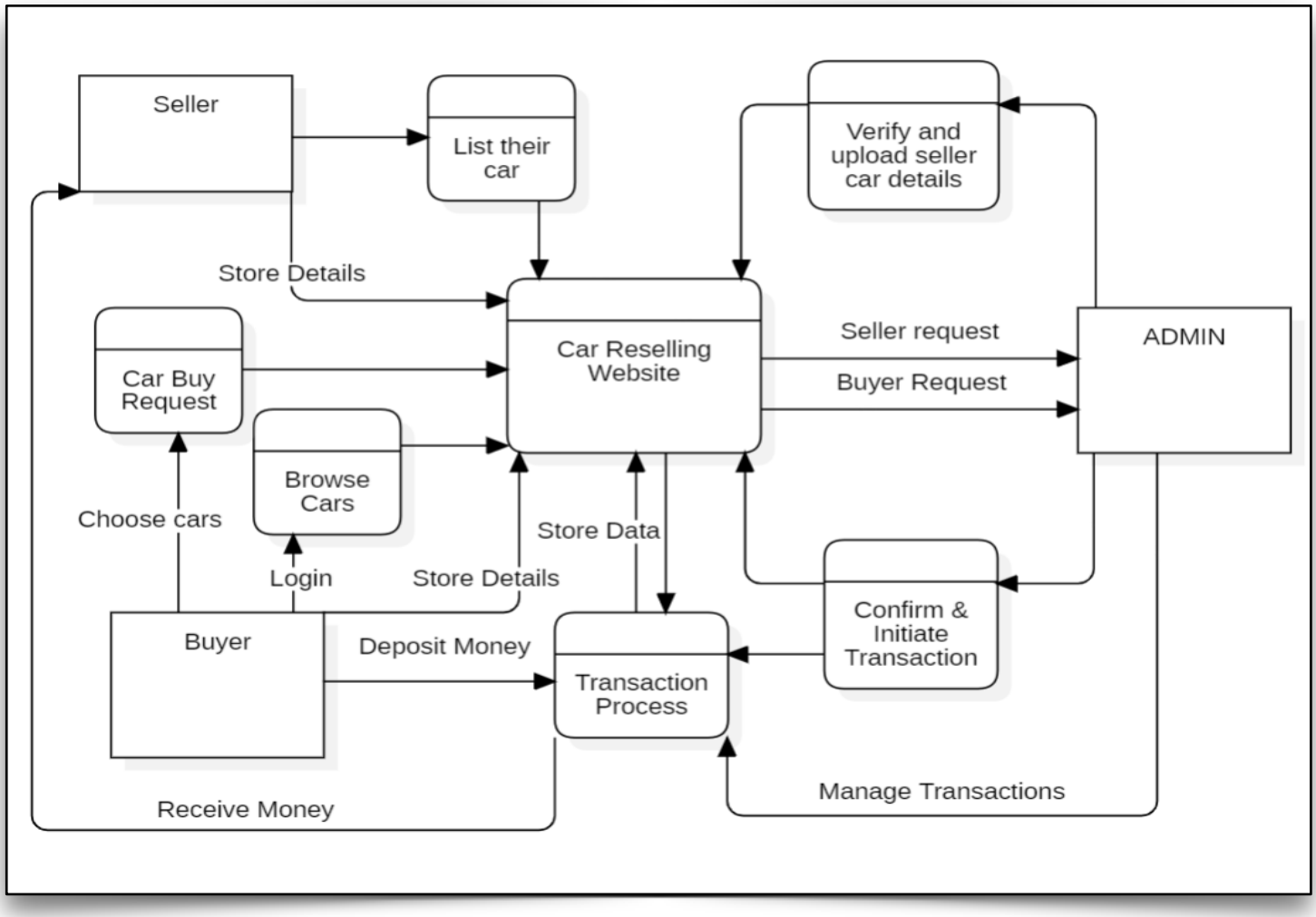
1. Data Flow Diagram (DFD)

The Data Flow Diagram (DFD) is a graphical representation of the flow of data through an information system. It enables you to represent the processes in your information system from the viewpoint of data. The DFD lets you visualize how the system operates, what the system accomplishes and how it will be implemented, when it is refined with further specification. Data flow diagrams are used by systems analysts to design information-processing systems but also as a way to model whole organizations.

➢ Context Level Diagram of the project:-



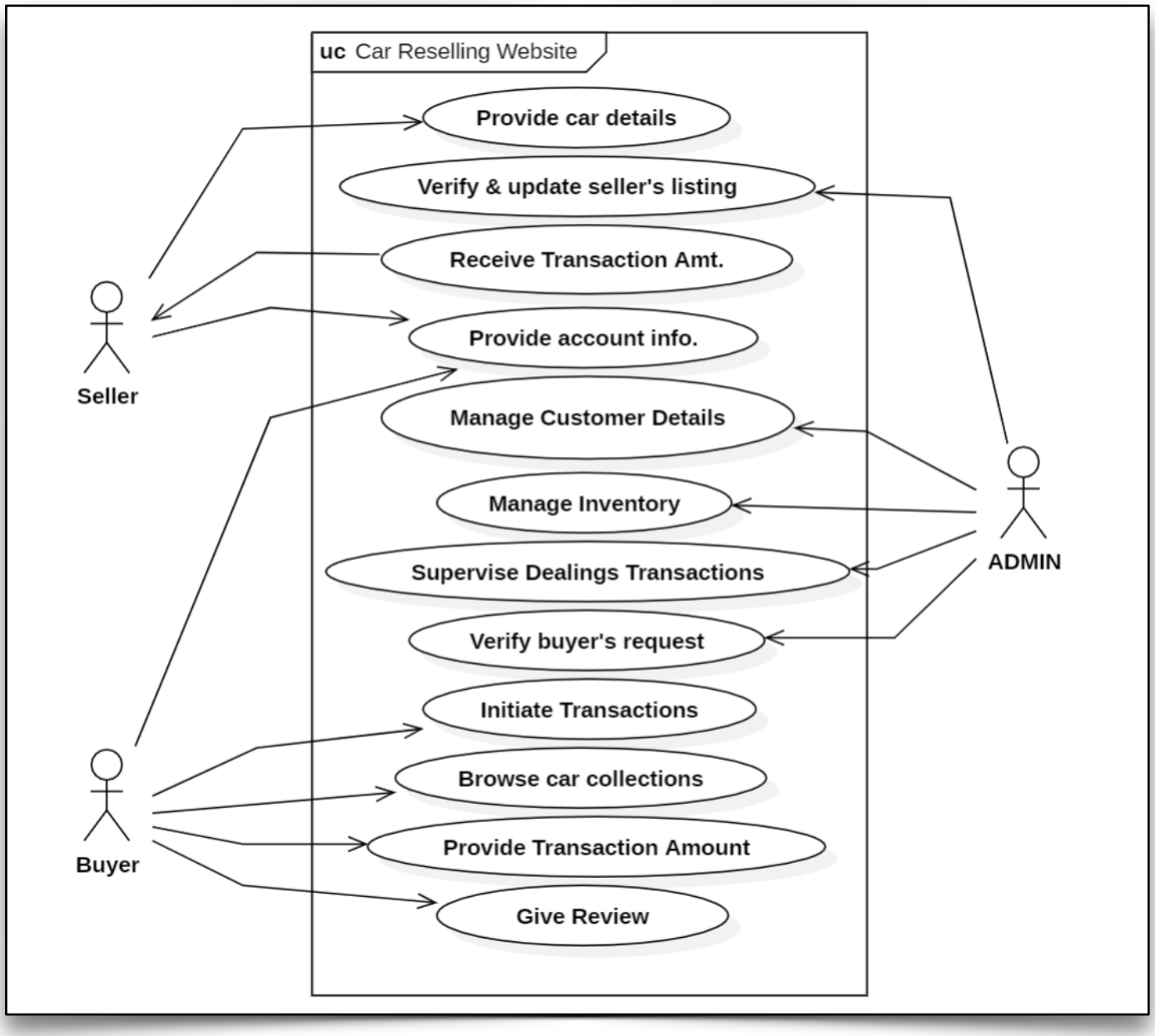
➢ Level 1 data flow diagram of the project:-



2. Unified Modelling Language (UML)

2.1. Use case Diagram:-

- **Use cases**: A use case describes a sequence of actions that provide something of measurable value to an actor and is drawn as a horizontal ellipse.
- **Actors**: An actor is a person, organization, or external system that plays a role in one or more interactions with your system. Actors are drawn as stick figures.
- **Associations**: Associations between actors and use cases are indicated in use case diagrams by solid lines. An association exists whenever an actor is involved with an interaction described by a use case.

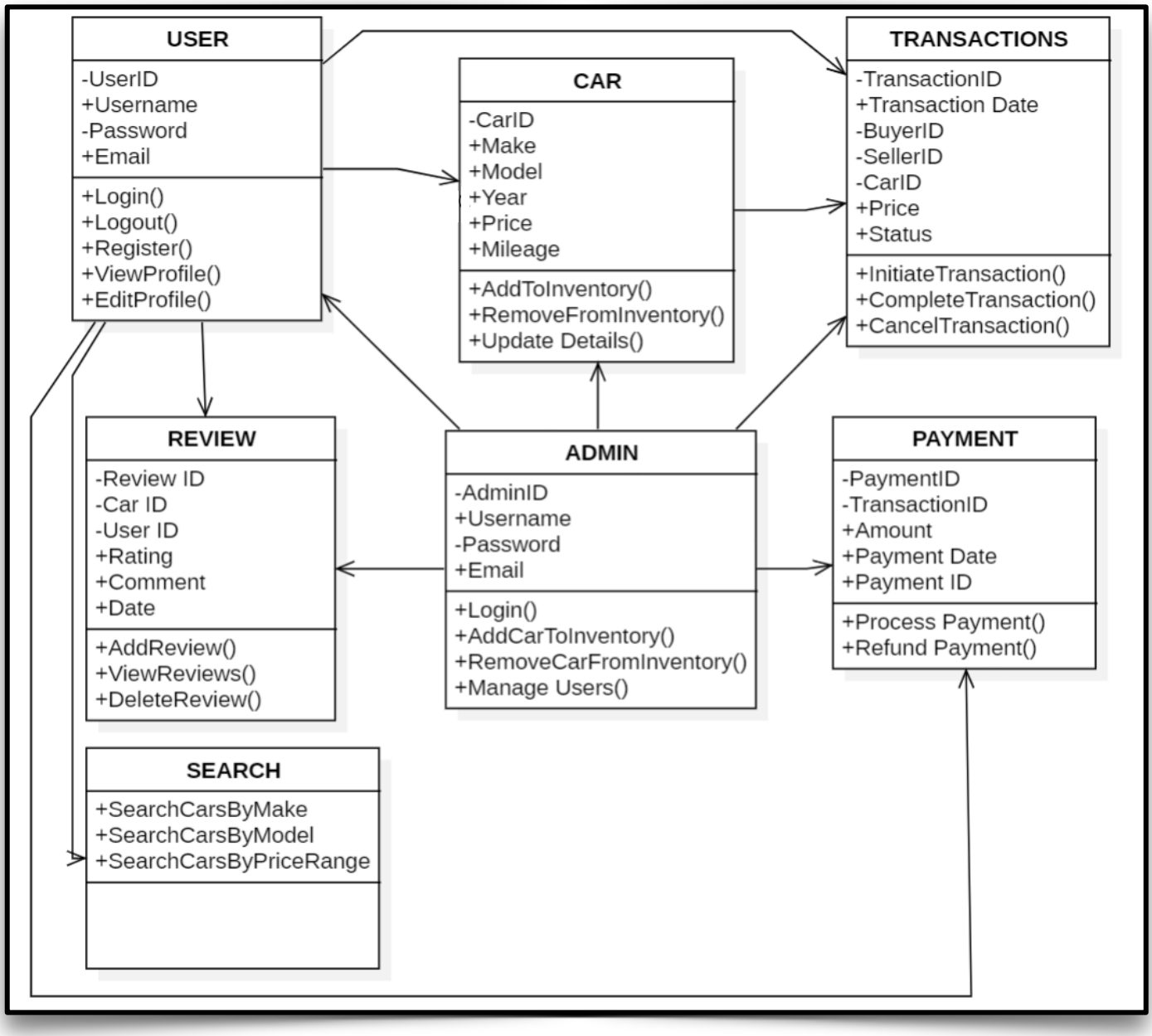


2.2. Class Diagram:-

Class Diagrams describe the static structure of a system, or how it is structured rather than how it behaves. A class diagram shows the existence of classes and their relationships in the logical view of a system.

These diagrams contain the following elements:-

- Classes and their structure and behaviour.
- Association, aggregation, dependency, and inheritance. relationships.
- Multiplicity and navigation indicators.
- Role names.

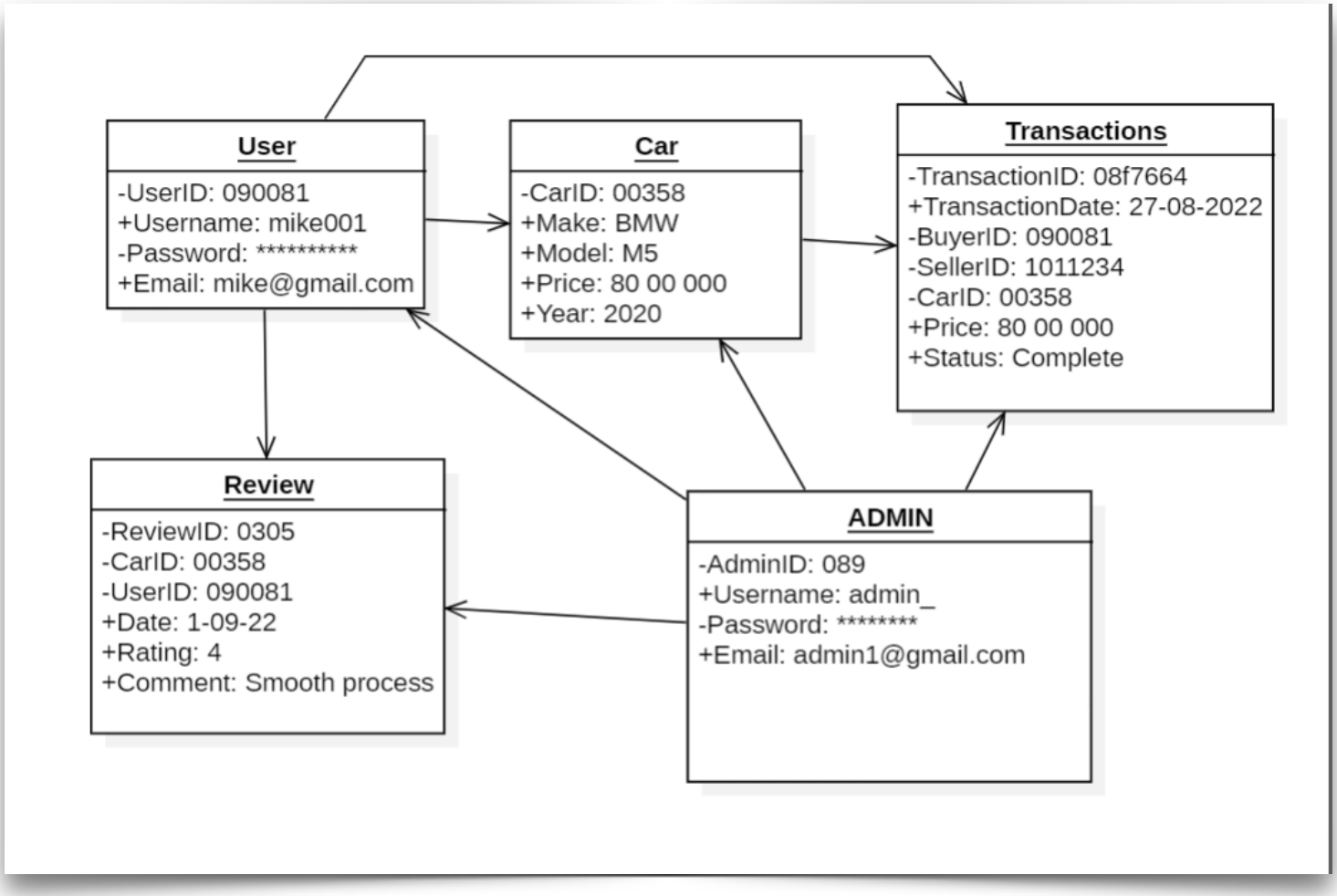


2.3. Object Diagram:-

Object Diagram shows a set of objects and their relationships. It is a static snapshot of instances. Object Diagrams describe the static structure of a system at a particular time. Whereas a class model describes all possible situations, an object model describes a particular situation.

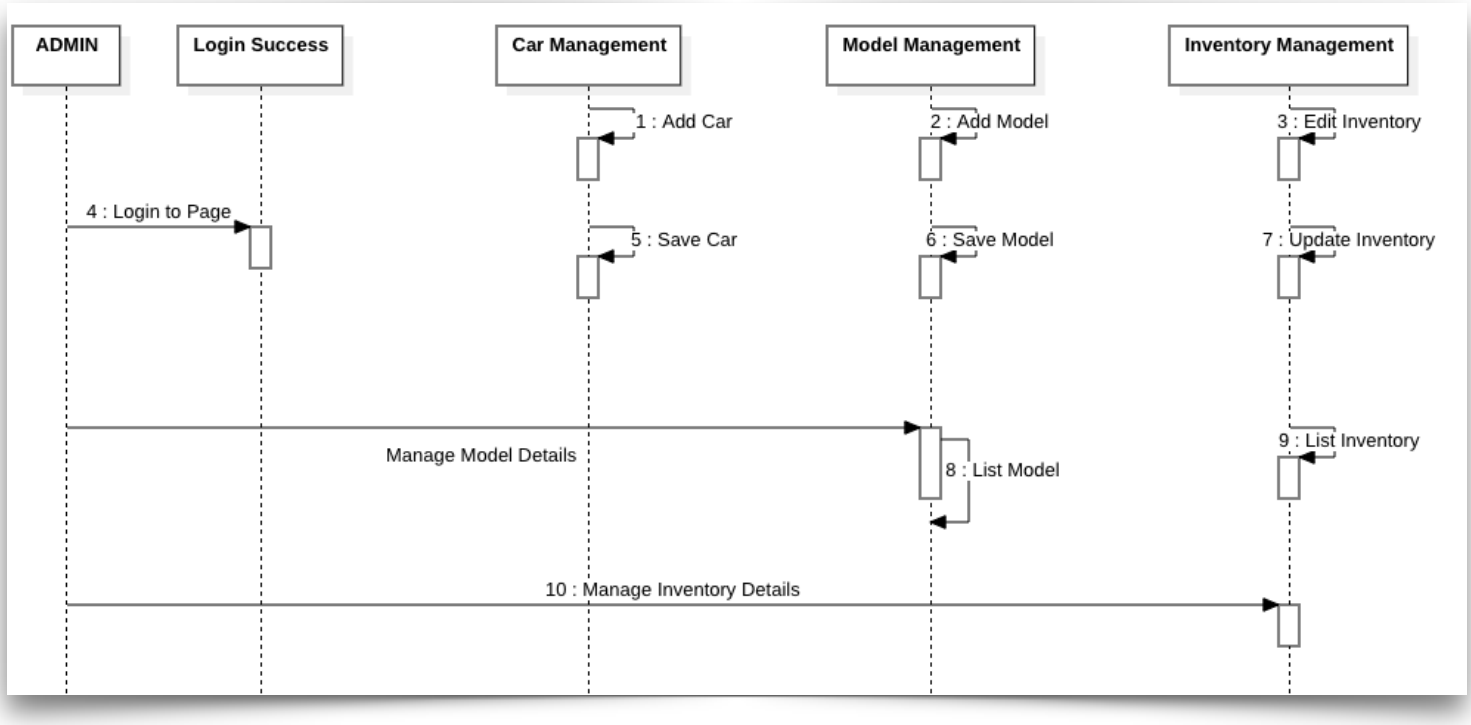
Object diagrams contain the following elements:-

Objects: It represents particular entities. These are instances of classes.
Links: It represents particular relationships between objects. These are instances of associations.



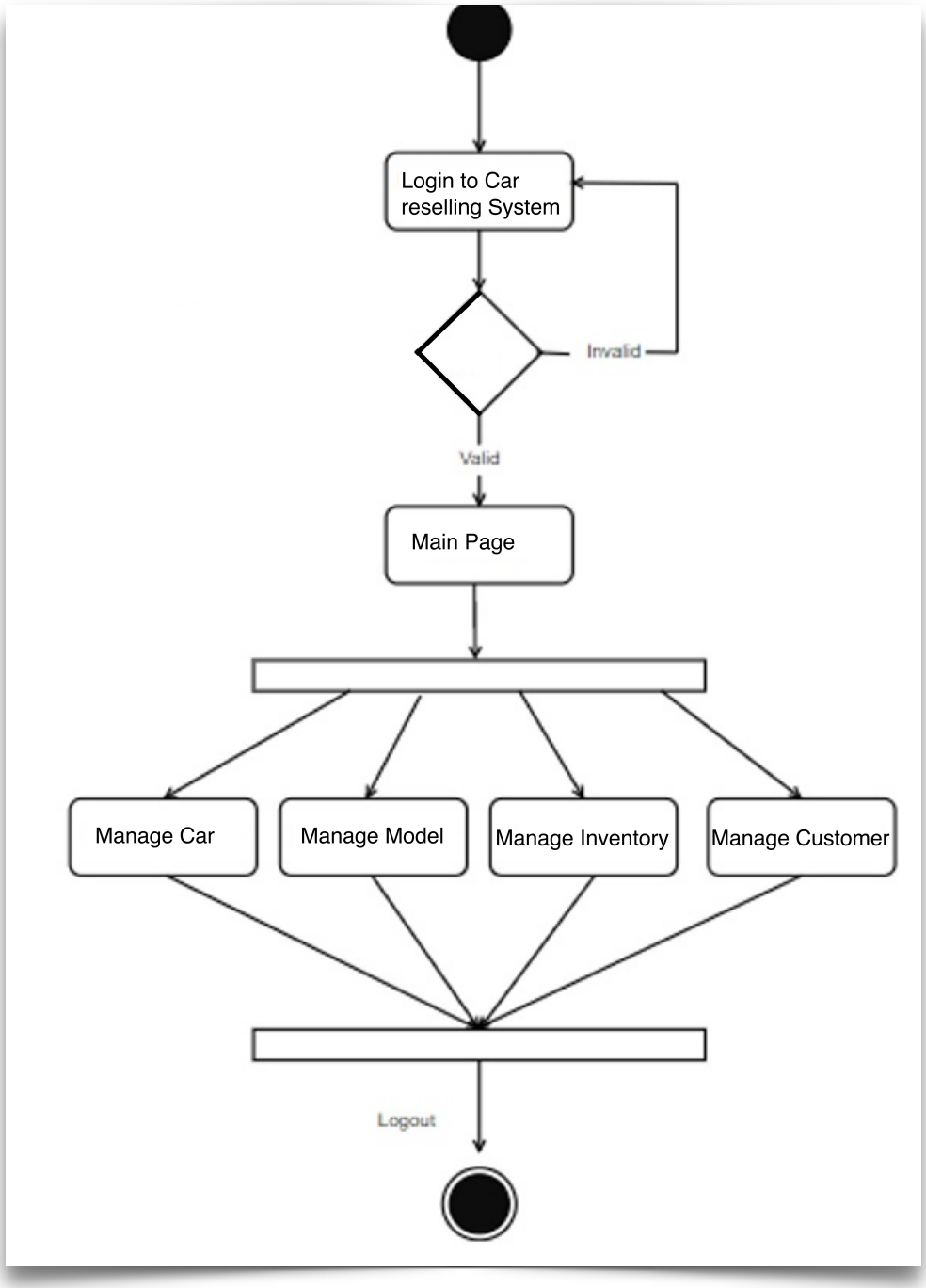
2.4. Sequence Diagram:-

- Sequence Diagrams describe interactions among classes. These interactions are modelled as exchanges of messages.
- These diagrams focus on classes and the messages they exchange to accomplish some desired behaviour.
- Sequence diagrams are a type of interaction diagrams.



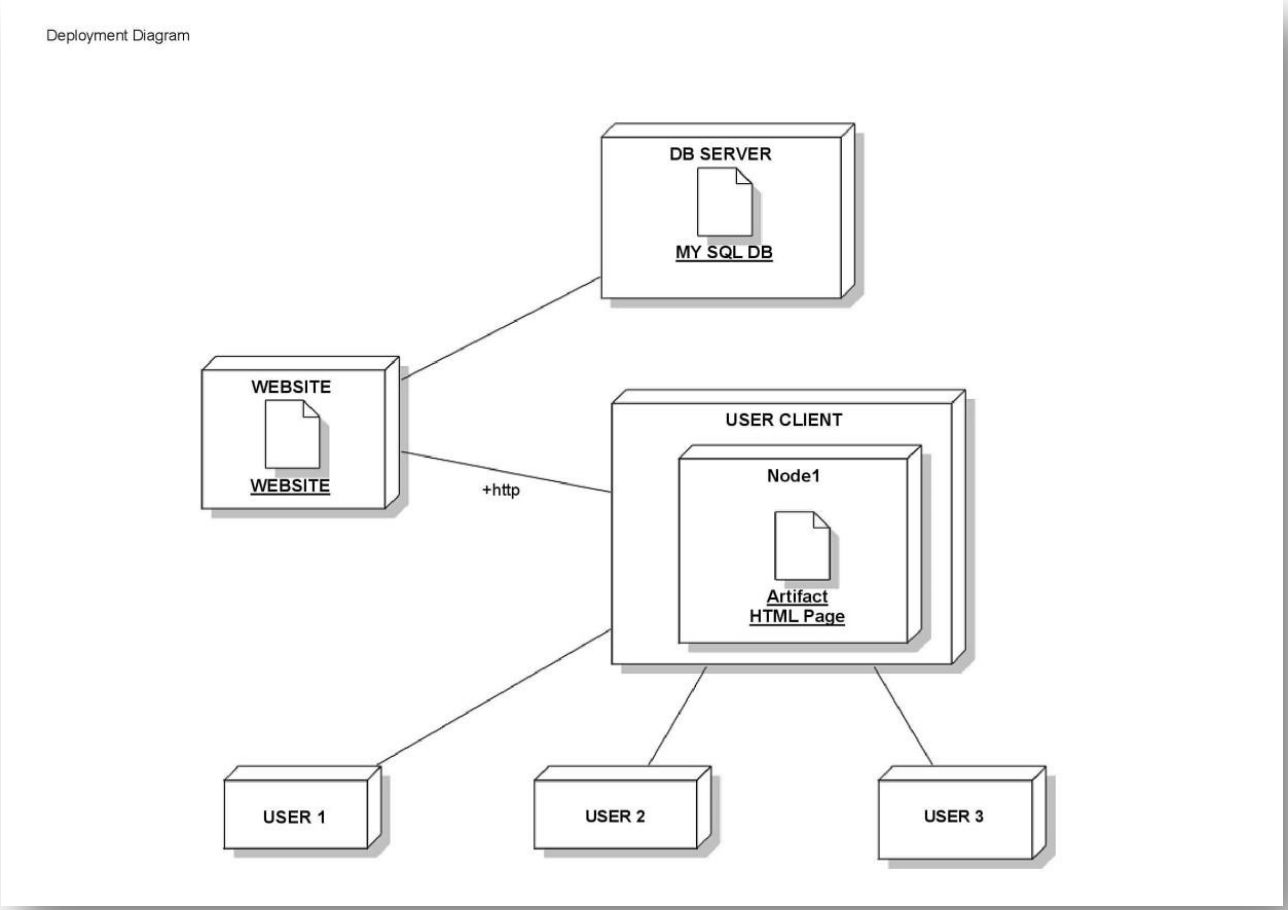
2.5. Activity Diagram:-

Activity diagrams describe the activities of a class. These diagrams are similar to Statechart diagrams and use similar conventions, but activity diagrams describe the behaviour of a class in response to internal processing rather than external events as in Statechart diagram.



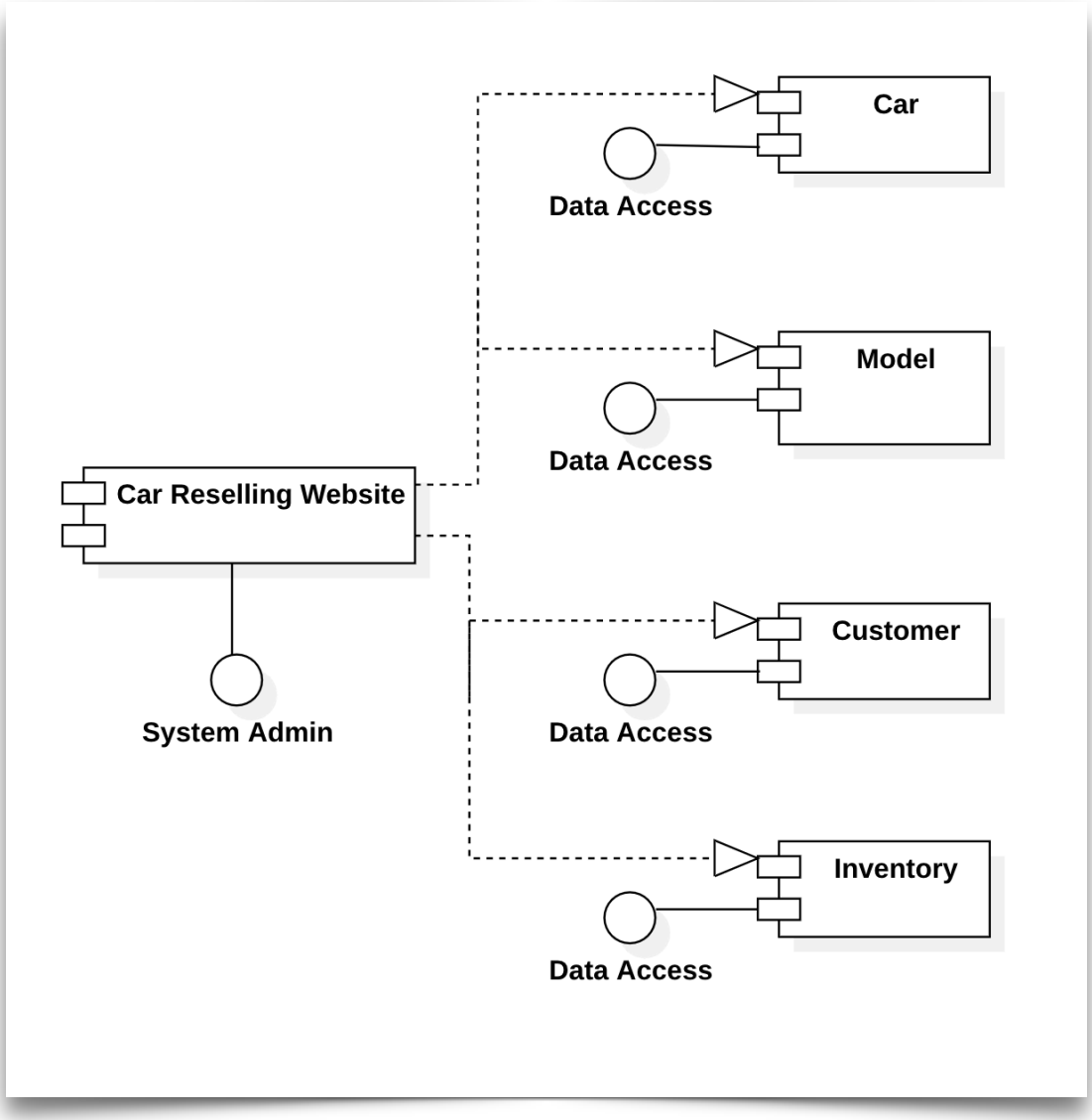
Deployment Diagram:-

Deployment diagrams describe the configuration of run-time processing resource elements and the mapping of software implementation components onto them. These Diagrams contain components and nodes, which represent processing or computational resources, including computers, printers, etc.



2.8. Component Diagram:-

Component diagrams describe the organizations and dependencies among software implementation components. These diagrams contain components, which represent distributable physical units; including source code, object code, and executable code.
These are static implementation view of a system.



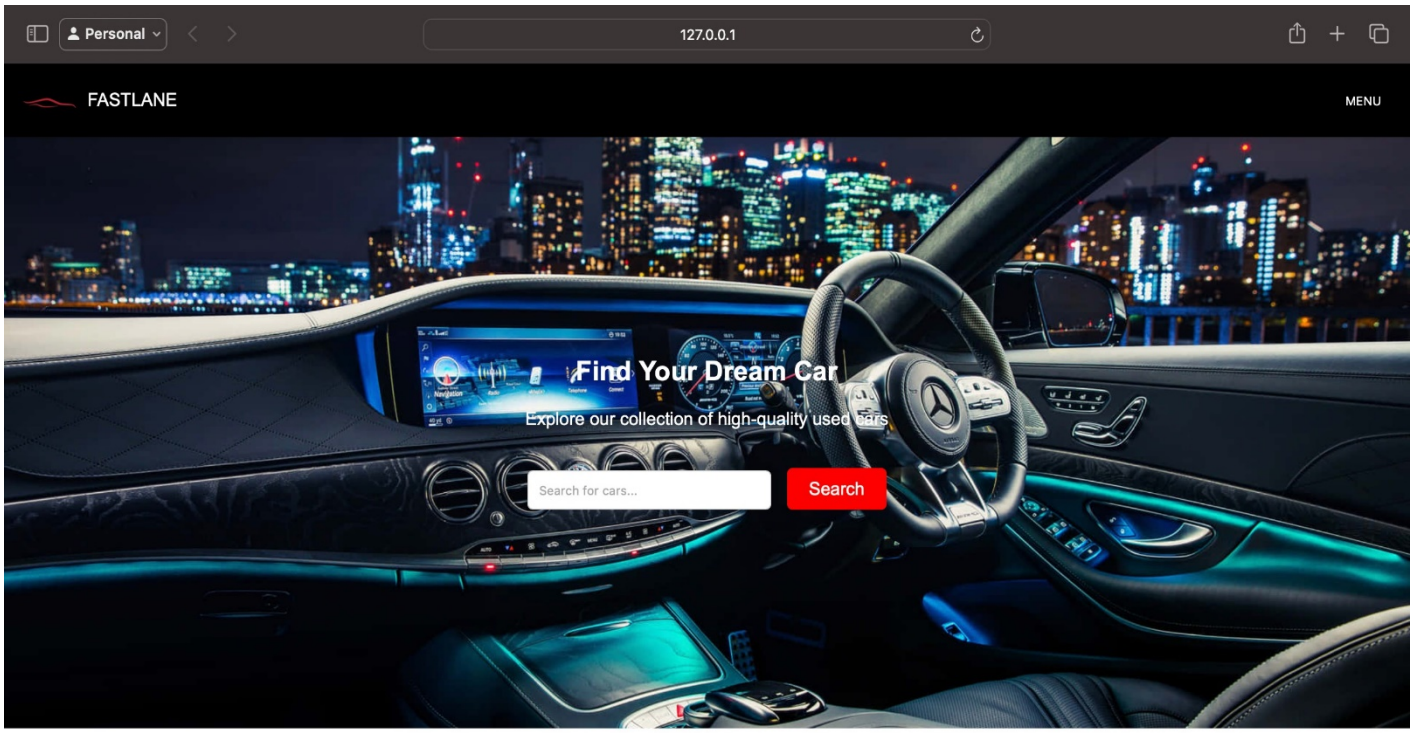
CHAPTER 4: CONCLUSION

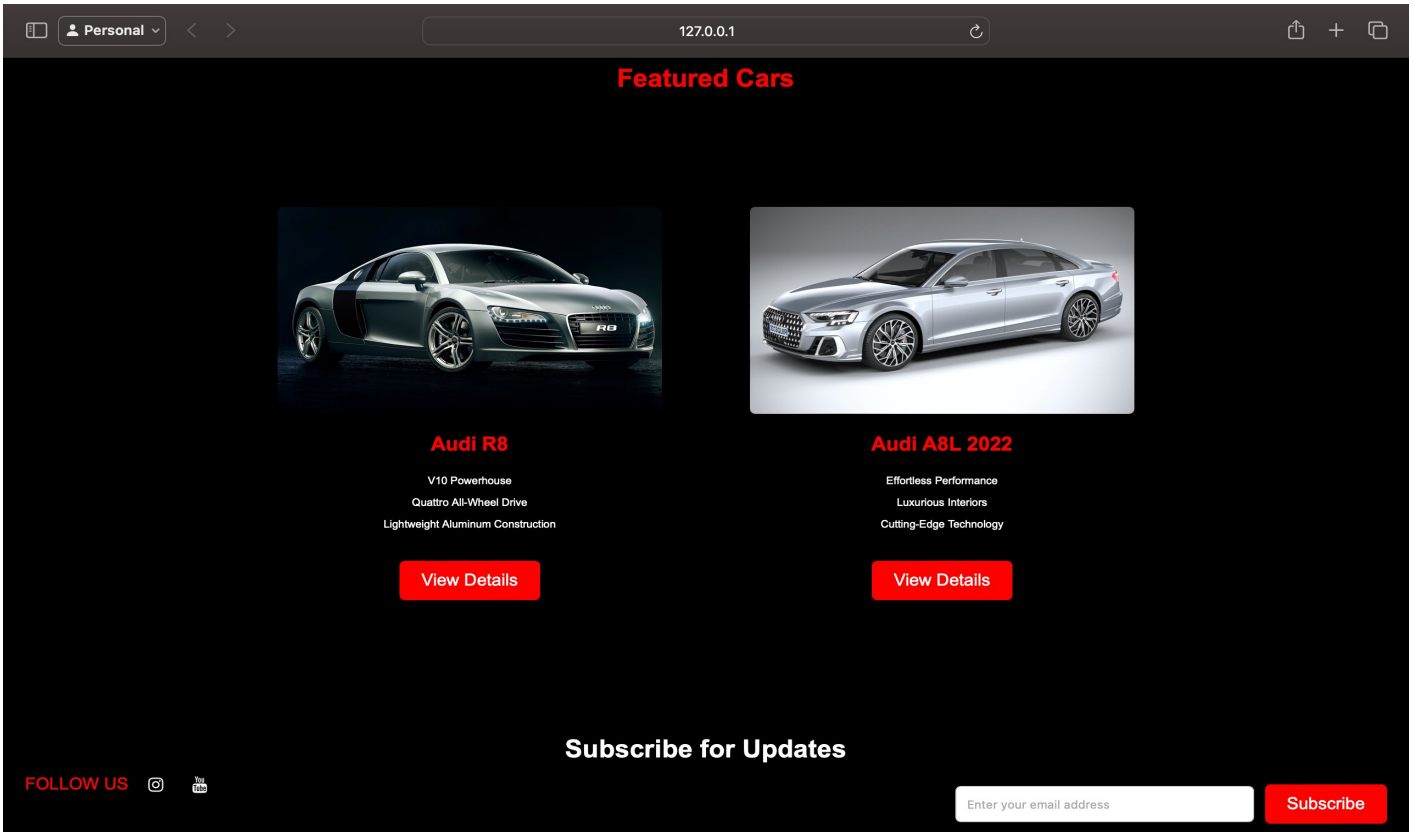
Thus people who want to sell their car can list their cars at the site and get reasonable prices for that . Customers can then browse for their choice of car brand and model and if they want to buy any car from the available listings then they can place a request for call / booking. The seller will also be notified about the same . A dealing will be initiated and the customer will be offered to have an on-site/offline inspection of the car for full satisfaction . Upon confirmation from the customer , the car will be sold and the seller will be provided the promised account .

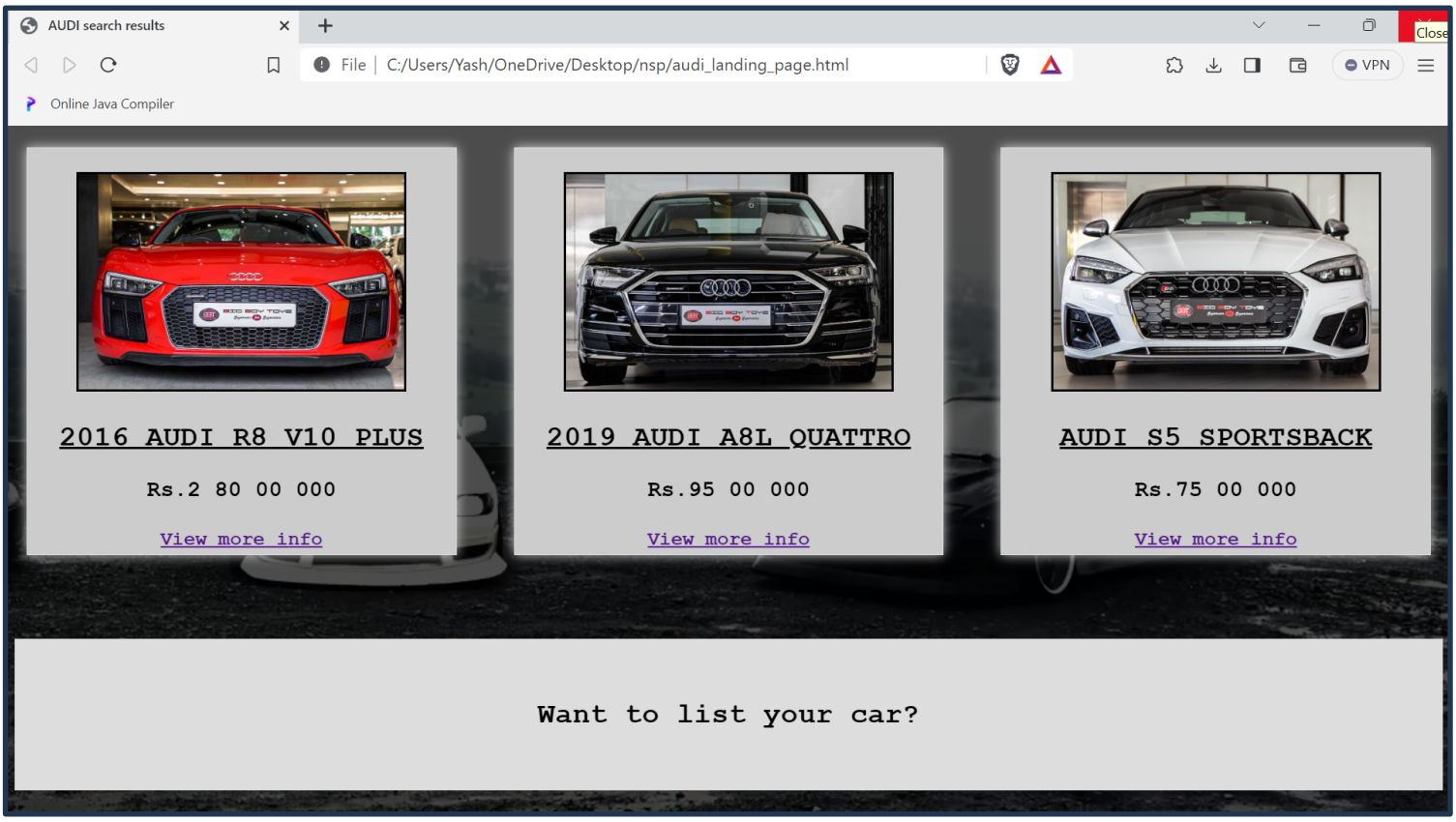
CHAPTER 5: REFERENCES

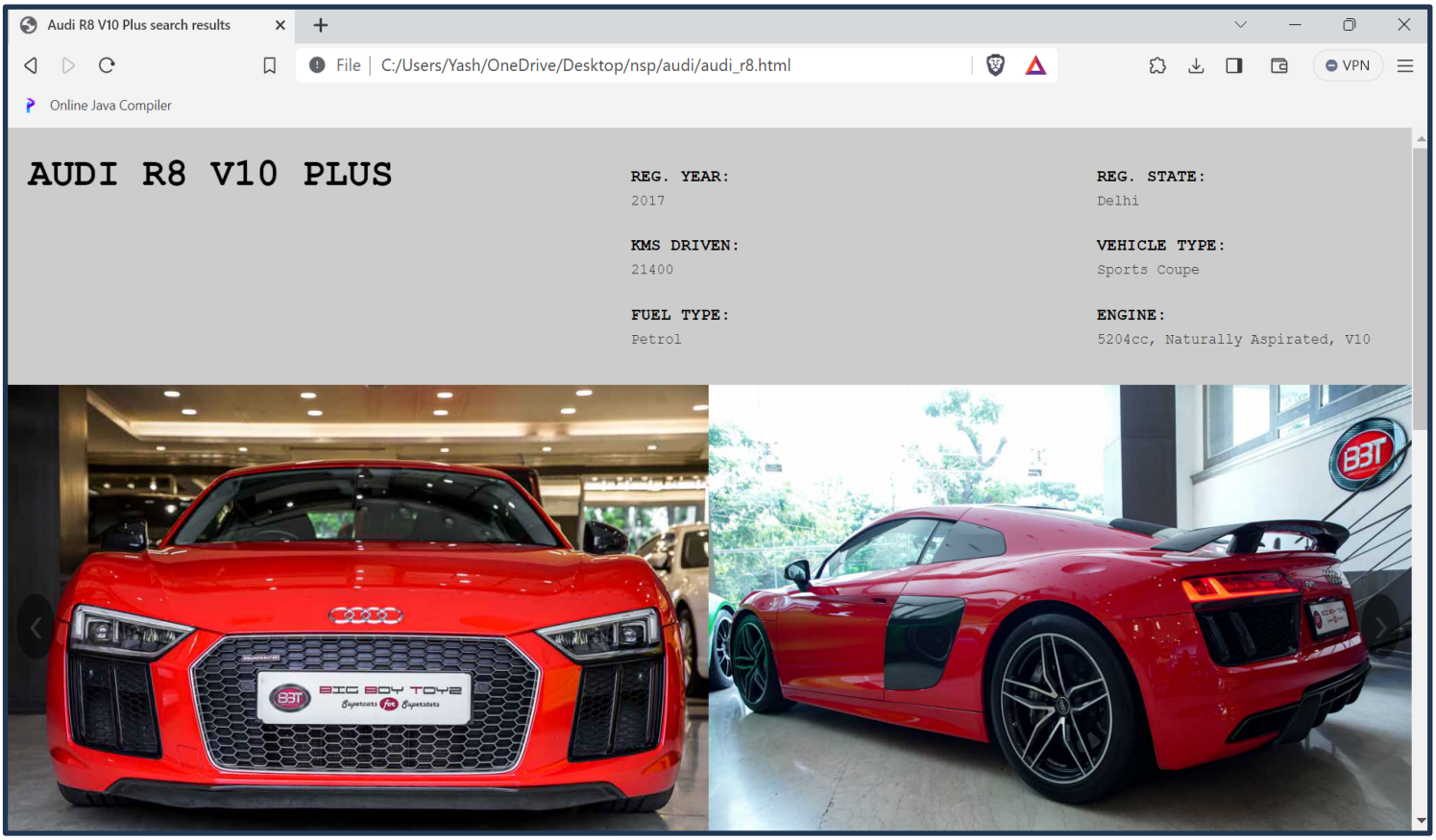
i. Big Boy Toyz website :- https://www.bigboytoyz.com/
ii. Github :- https://github.com/
iii. Stack Overflow:- https://stackoverflow.com/
iv. W3 Schools :- https://www.w3schools.com/
v. Google Online Search Platform :- https://www.google.com/
vi. College Faculty:- *Ms. Surbhi Saxena*
vii. Textbooks.

CHAPTER 6: SNAPSHOTS OF PROJECT

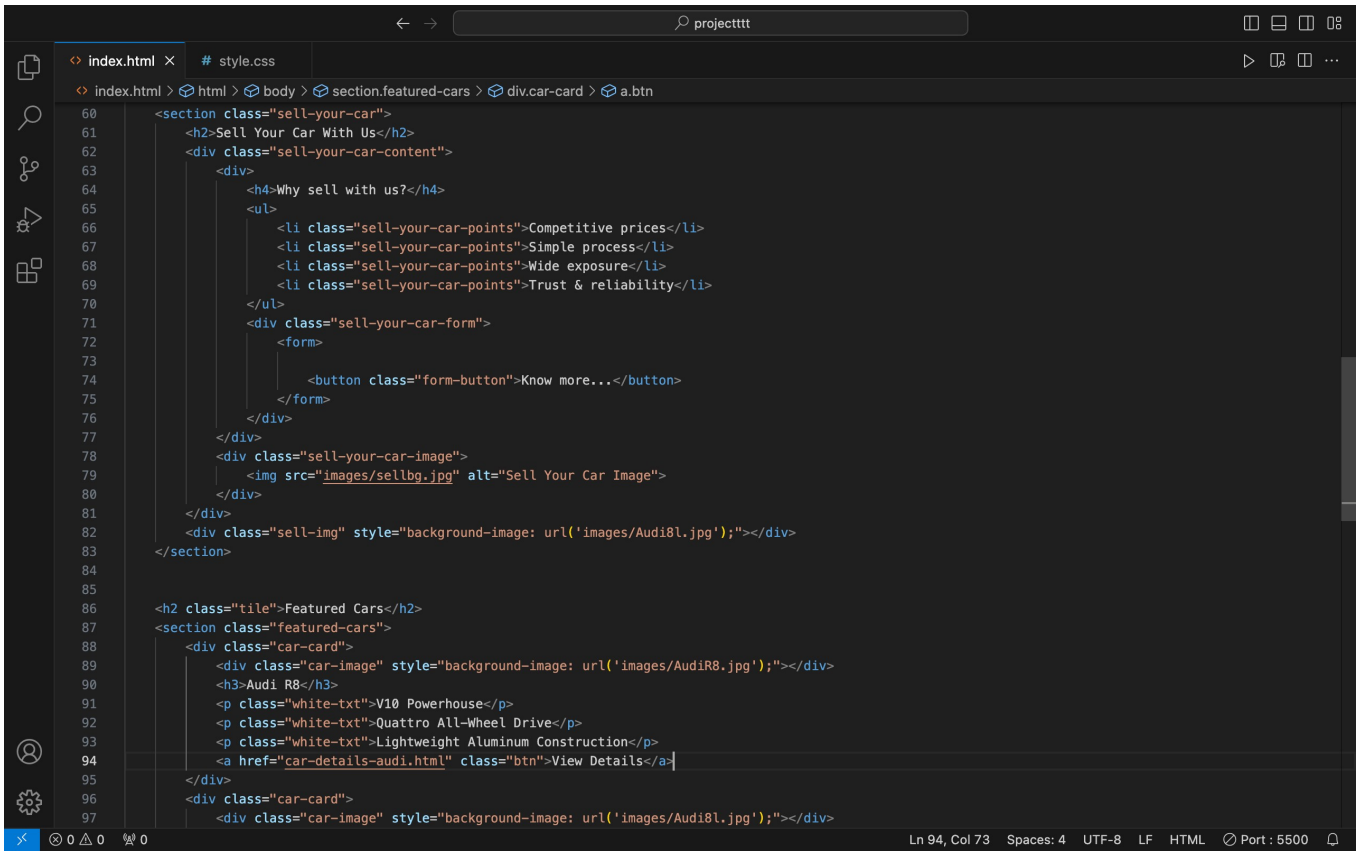





AUDI search results
File | C:/Users/Yash/OneDrive/Desktop/nsp/audi_landing_page.html
Online Java Compiler
2016 AUDI R8 V10 PLUS
Rs.2 80 00 000
View more info
2019 AUDI A8L QUATTRO
Rs.95 00 000
View more info
AUDI S5 SPORTSBACK
Rs.75 00 000
View more info
Want to list your car?


Audi R8 V10 Plus search results
File | C:/Users/Yash/OneDrive/Desktop/nsp/audi/audi_r8.html
Online Java Compiler
AUDI R8 V10 PLUS
REG. YEAR:
2017
REG. STATE:
Delhi
KMS DRIVEN:
21400
VEHICLE TYPE:
Sports Coupe
FUEL TYPE:
Petrol
ENGINE:
5204cc, Naturally Aspirated, V10

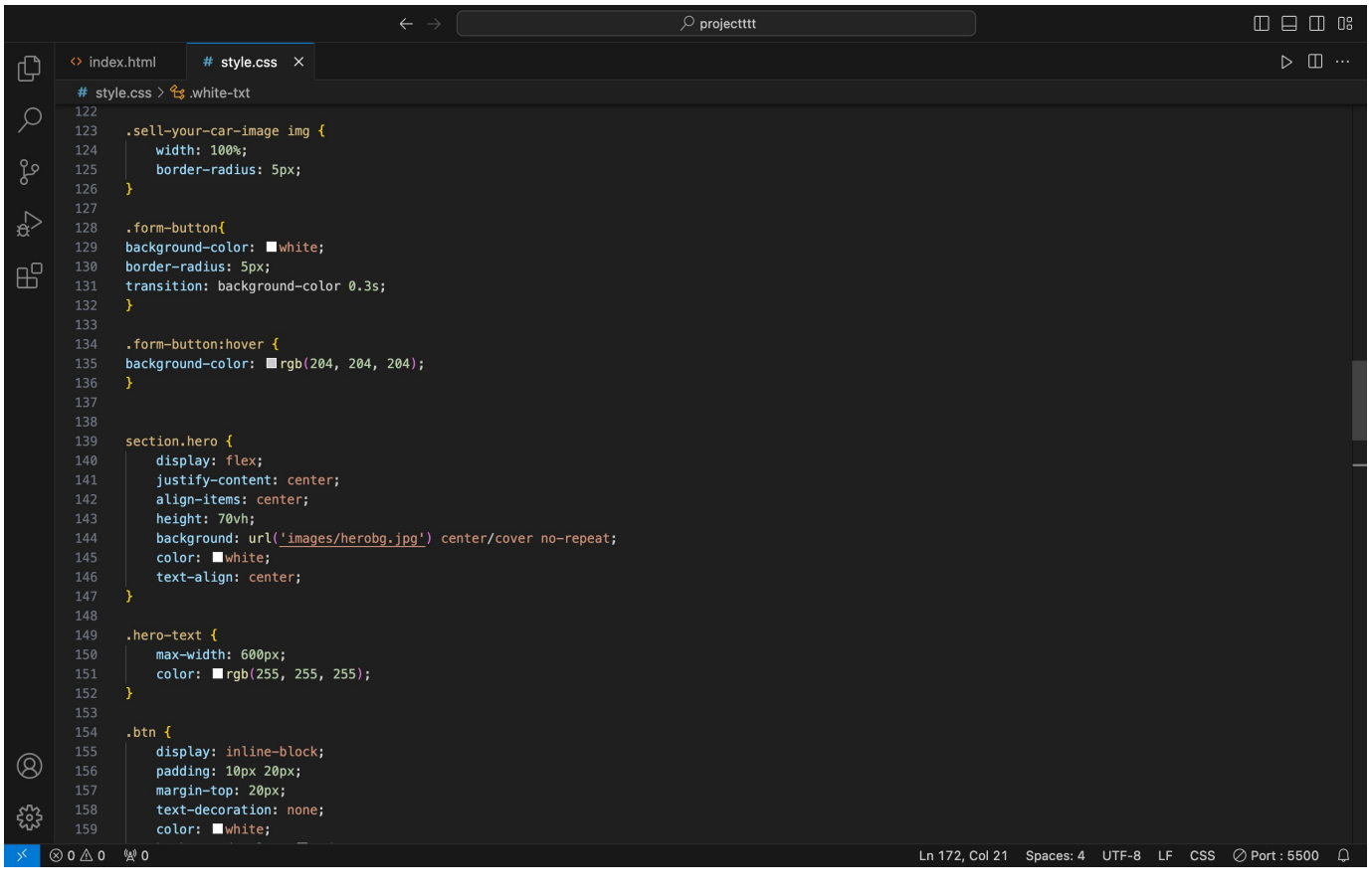
CHAPTER 7: CODE OF PROJECT



```
<section class="sell-your-car">
    <h2>Sell Your Car With Us</h2>
    <div class="sell-your-car-content">
        <div>
            <h4>Why sell with us?</h4>
            <ul>
                <li class="sell-your-car-points">Competitive prices</li>
                <li class="sell-your-car-points">Simple process</li>
                <li class="sell-your-car-points">Wide exposure</li>
                <li class="sell-your-car-points">Trust & reliability</li>
            </ul>
            <div class="sell-your-car-form">
                <form>

                    <button class="form-button">Know more...</button>
                </form>
            </div>
        </div>
        <div class="sell-your-car-image">
            <img src="images/sellbg.jpg" alt="Sell Your Car Image">
        </div>
    </div>
    <div class="sell-img" style="background-image: url('images/Audi8l.jpg');"></div>
</section>


<h2 class="tile">Featured Cars</h2>
<section class="featured-cars">
    <div class="car-card">
        <div class="car-image" style="background-image: url('images/AudiR8.jpg');"></div>
        <h3>Audi R8</h3>
        <p class="white-txt">V10 Powerhouse</p>
        <p class="white-txt">Quattro All-Wheel Drive</p>
        <p class="white-txt">Lightweight Aluminum Construction</p>
        <a href="car-details-audi.html" class="btn">View Details</a>
    </div>
    <div class="car-card">
        <div class="car-image" style="background-image: url('images/Audi8l.jpg');"></div>
```



```
.sell-your-car-image img {
    width: 100%;
    border-radius: 5px;
}

.form-button{
background-color: white;
border-radius: 5px;
transition: background-color 0.3s;
}

.form-button:hover {
background-color: rgb(204, 204, 204);
}


section.hero {
    display: flex;
    justify-content: center;
    align-items: center;
    height: 70vh;
    background: url('images/herobg.jpg') center/cover no-repeat;
    color: white;
    text-align: center;
}

.hero-text {
    max-width: 600px;
    color: rgb(255, 255, 255);
}

.btn {
    display: inline-block;
    padding: 10px 20px;
    margin-top: 20px;
    text-decoration: none;
    color: white;
```

CHAPTER 7: GLOSSARY

Term	**Definition**
Active Article	The document that is tracked by the system; it is a narrative that is planned to be posted to the public website.
Author	Person submitting an article to be reviewed. In case of multiple authors, this term refers to the *principal author*, with whom all communication is made.
Database	Collection of all the information monitored by this system
Editor	Person who receives articles, sends articles for review, and makes final judgments for publications.
Field	A cell within a form.
Historical Society Database	The existing membership database (also HS database).
Member	A member of the Historical Society listed in the HS database.
Reader	Anyone visiting the site to read articles.
Review	A written recommendation about the appropriateness of an

	article for publication; may include suggestions for improvement.
Reviewer	A person that examines an article and has the ability to recommend approval of the article for publication or to request that changes be made in the article.
Software Requirements Specification	A document that completely describes all of the functions of a proposed system and the constraints under which it must operate. For example, this document.
Stakeholder	Any person with an interest in the project who is not a developer.
User	Reviewer or Author.