

# **McQueen : A car dealing website**

A Minor Project Report  
submitted in partial fulfillment of the requirements for  
the award of the degree of

**Bachelor of Engineering**

in  
**Artificial Intelligence and Data Science**

By  
**J. Monesh (160121771039)**  
**Md. Mushtaq (160121771050)**  
**G. Sai Aashrith Reddy (160121771058)**

*Under the esteemed guidance of*

**V. Krishna Aravindha**

Assistant Professor



**DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE  
CHAITANYA BHARATHI INSTITUTE OF TECHNOLOGY  
HYDERABAD – 500075  
DECEMBER 2023**



**DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE  
CHAITANYA BHARATHI INSTITUTE OF TECHNOLOGY  
HYDERABAD – 500075**

**INSTITUTE VISION**

“To be the center of excellence in technical education and research”.

**INSTITUTE MISSION**

“To address the emerging needs through quality technical education and advanced research”.

**DEPARTMENT VISION**

”To be a globally recognized center of excellence in the field of Artificial Intelligence and Data Science that produces innovative pioneers and research experts capable of addressing complex real-world challenges and contributing to the socio-economic development of the nation.”

**DEPARTMENT MISSION**

1. To provide cutting-edge education in the field of Artificial Intelligence and Data Science that is rooted in ethical and moral values.
2. To establish strong partnerships with industries and research organizations in the field of Artificial Intelligence and Data Science, and to excel in the emerging areas of research by creating innovative solutions.
3. To cultivate a strong sense of social responsibility among students, fostering their inclination to utilize their knowledge and skills for the betterment of society.
4. To motivate and mentor students to become trailblazers in Artificial Intelligence and Data Science, and develop an entrepreneurial mindset that nurtures innovation and creativity.



**DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE  
CHAITANYA BHARATHI INSTITUTE OF TECHNOLOGY  
HYDERABAD – 500075**

**PROGRAM EDUCATIONAL OBJECTIVES (PEOs)**

Graduates of AI & DS will be able to:

1. Adapt emerging technologies of Artificial Intelligence & Data Science and develop state-of-the-art solutions in the fields of Manufacturing, Agriculture, Health, Education, and Cyber Security.
2. Exhibit professional leadership qualities to excel in interdisciplinary domains.
3. Possess human values, professional ethics, application-oriented skills, and engage in lifelong learning.
4. Contribute to the research community to meet the needs of public and private sectors.

**PROGRAM SPECIFIC OUTCOMES (PSOs)**

After successful completion of the program, students will be able to:

1. Exhibit proficiency in Artificial Intelligence and Data Science in providing sustainable solutions by adapting to societal, environmental, and ethical concerns to real-world problems.
2. Develop professional skills in the thrust areas like ANN and Deep learning, Robotics, Internet of Things, and Big Data Analytics.
3. Pursue higher studies in Artificial Intelligence and Data Science in reputed Universities and work in research establishments.



**DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE  
CHAITANYA BHARATHI INSTITUTE OF TECHNOLOGY  
HYDERABAD – 500075**

**PROGRAM OUTCOMES**

1. **Engineering Knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization for the solution of complex engineering problems
2. **Problem analysis:** Identify, formulate, review, research literature, and analyse complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for public health and safety, and cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools, including prediction and modeling to complex engineering activities with an understanding of the limitations.
6. **The engineer and society:** Apply reasoning informed by contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. **Environment and sustainability:** Understand the impact of professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. **Communication:** Communicate effectively on complex engineering activities with the engineering community and with the society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
11. **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
12. **Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.



**DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE  
CHAITANYA BHARATHI INSTITUTE OF TECHNOLOGY  
HYDERABAD – 500075**

**MINOR PROJECT-I**

**COURSE OBJECTIVES**

1. To enable students to learn by doing.
2. To develop the capability to analyze and solve real-world problems.
3. To inculcate innovative ideas of the students.
4. To impart team building and management skills among students.
5. To instill writing and presentation skills for completing the project.

**COURSE OUTCOMES**

Upon successful completion of this course, students will be able to:

1. Interpret Literature to formulate a project proposal.
2. Plan, Analyse, Design and Implement a project using the SDLC model.
3. Find the solution to the identified problem with the help of modern technology and give priority to real-time scenarios.
4. Plan to work as a team and to focus on getting a working project done and submit a report within an estimated period.
5. Prepare and submit the Report and deliver a presentation before the Departmental Committee.



**DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE  
CHAITANYA BHARATHI INSTITUTE OF TECHNOLOGY  
HYDERABAD – 500075**

**CO-PO MAPPING**

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	3	3	3	3	2	1	2	3	3
CO2	3	3	3	3	3	3	3	2	1	2	3	3
CO3	3	3	3	3	3	3	3	2	-	2	3	3
CO4	2	2	2	3	3	3	3	2	3	3	2	3
CO5	1	2	1	2	3	3	-	-	2	3	-	-

Mapping of Course Outcomes with Program Outcomes

**CO-PSO MAPPING**

	PSO1	PSO2	PSO3
CO1	2	3	3
CO2	3	3	3
CO3	3	3	3
CO4	2	3	3
CO5	-	3	-

Mapping of Course Outcomes with Program Specific Outcomes



**DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE  
CHAITANYA BHARATHI INSTITUTE OF TECHNOLOGY  
HYDERABAD – 500075**

**DECLARATION CERTIFICATE**

We hereby declare that the project titled **McQueen : A car dealing website** submitted by us to the **Artificial Intelligence and Data Science CHAITANYA BHARATHI INSTITUTE OF TECHNOLOGY, HYDERABAD** in partial fulfillment of the requirements for the award of **Bachelor of Engineering** is a bonafide record of the work carried out by us under the supervision of **V. Krishna Aravinda**. We further declare that the work reported in this project, has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma of this institute or of any other institute or University.

Project Associates

**J. Monesh (160121771039)**

**Md. Mushtaq (160121771050)**

**G. Sai Aashrith Reddy (160121771058)**



**DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE  
CHAITANYA BHARATHI INSTITUTE OF TECHNOLOGY  
HYDERABAD – 500075**

**BONAFIDE CERTIFICATE**

This is to certify that the project titled **McQueen : A car dealing website** is a bonafide record of the work done by

**J. Monesh (160121771039)**

**Md. Mushtaq (160121771050)**

**G. Sai Aashrith Reddy (160121771058)**

in partial fulfillment of the requirements for the award of the degree of **Bachelor of Engineering in Artificial Intelligence and Data Science** to the **CHAITANYA BHARATHI INSTITUTE OF TECHNOLOGY, HYDERABAD** carried out under my guidance and supervision during the year 2023-24. The results presented in this project report have not been submitted to any other university or Institute for the award of any degree.

**V. Krishna Aravindh**

Guide

**Dr. K. Ramana**

Head of the Department

Submitted for Semester Minor-Project viva-voce examination held on \_\_\_\_\_

**Examiner-1**

**Examiner-2**

## ABSTRACT

The automotive industry is undergoing a profound transformation in today's fast-paced world. Our mission is to empower consumers and dealers alike, offering an intuitive, secure, and efficient online marketplace redefining how vehicles change hands. Our platform looks at leveraging cutting-edge technology to create a seamless experience for buyers and sellers by making it easier for buyers to find, compare, and purchase vehicles, whether new or used, maintaining a user-friendly interface and robust search tools and providing comprehensive vehicle information, including histories and condition reports, and for sellers to promote transparency and build trust between buyers and sellers. The data displayed about the cars is real-time data and is regularly updated as we are using APIs from RapidAPI for fetching live car details.

For buyers, we provide a vast inventory of pre-owned and new vehicles, comprehensive vehicle histories, and personalized search tools, ensuring they easily find their dream car. For sellers, we offer a user-friendly listing process, expert pricing guidance, and a broad audience reach, resulting in quicker sales at the best prices. Moreover, we intend to prioritize safety, and security, and provide secure payment options.

Automobile marketplace websites such as autocarindia.com and cars24.com have emerged as transformative platforms within the automotive industry. These digital marketplaces play a pivotal role in facilitating the sale of vehicles. Although they offer convenience and trust, such platforms typically offer a price for your vehicle that is lower than what you might get through a private sale. Also, these platforms charge a service fee or commission for their services, which can reduce the final amount you receive for your vehicle.

We intend to create our platform by making use of the MERN stack ( MongoDB, Express.js, React.js, and Node.js stack) to make our platform a culmination of simplistic UI, consisting of a vast array of cars to choose from, which is completely based on the user's choice and preference. We offer pricing guidance and data analytics to help sellers determine fair and competitive vehicle prices. Along with that, the platform seeks to provide fair and transparent pricing, all, without charging any service fees, in order to ensure that both buyers and sellers can transact without incurring any additional costs or charges.

## **ACKNOWLEDGEMENTS**

We would like to express our deepest gratitude to the following people for guiding us through this course and without whom this project and the results achieved from it would not have reached completion.

**V. Krishna Aravindha**, Assistant Professor, Department of Artificial Intelligence and Data Science, for helping us and guiding us in the course of this project. Without his/her guidance, we would not have been able to complete this project. His/Her patience and genial attitude are and always will be a source of inspiration to us.

**Dr. K. Ramana**, the Head of the Department, Department of Artificial Intelligence and Data Science, for allowing us to avail the facilities at the department.

We are also thankful to the faculty and staff members of the Department of Artificial Intelligence and Data Science, our parents, and our friends for their constant support and help.

## TABLE OF CONTENTS

Title	Page No.
<b>ABSTRACT</b> . . . . .	i
<b>ACKNOWLEDGEMENTS</b> . . . . .	ii
<b>TABLE OF CONTENTS</b> . . . . .	iii
<b>LIST OF FIGURES</b> . . . . .	v
<b>CHAPTER 1 INTRODUCTION</b> . . . . .	1
1.1 OVERVIEW . . . . .	1
1.2 PROBLEM STATEMENT . . . . .	1
1.3 ORGANIZATION OF THE PROJECT . . . . .	2
<b>CHAPTER 2 SYSTEM REQUIREMENTS</b> . . . . .	3
2.1 SOFTWARE REQUIREMENTS . . . . .	3
2.2 HARDWARE REQUIREMENTS . . . . .	3
<b>CHAPTER 3 LITERATURE SURVEY</b> . . . . .	4
3.1 MongoDB . . . . .	4
3.2 Express . . . . .	4
3.3 React . . . . .	4
3.4 Node . . . . .	4
3.5 Javascript . . . . .	5
3.6 JSON Web Tokens (JWT) . . . . .	5
3.7 Multer . . . . .	5
3.8 Axios . . . . .	5
<b>CHAPTER 4 IMPLEMENTATION</b> . . . . .	6
4.1 Frontend Development . . . . .	8

4.2	Backend Development . . . . .	9
4.3	Search Cars . . . . .	10
4.4	External APIs Integration . . . . .	11
<b>CHAPTER 5</b>	<b>RESULTS . . . . .</b>	<b>12</b>
<b>CHAPTER 6</b>	<b>FUTURE SCOPE . . . . .</b>	<b>20</b>
<b>CHAPTER 7</b>	<b>CONCLUSION . . . . .</b>	<b>21</b>

## LIST OF FIGURES

4.1 Main Page code snippet . . . . .	9
4.2 Backend code snippet . . . . .	9
4.3 Search cars code snippet . . . . .	10
5.1 Landing Page . . . . .	12
5.2 Services Section . . . . .	12
5.3 Portfolio Section . . . . .	13
5.4 Team Section . . . . .	13
5.5 About Section . . . . .	14
5.6 Contact Us Section . . . . .	14
5.7 Explore Cars Page . . . . .	15
5.8 Checkout Page . . . . .	15
5.9 Verification Page . . . . .	16
5.10 Confirmation Page . . . . .	16
5.11 Sign In Page . . . . .	17
5.12 Sign Up Page . . . . .	17
5.13 Dashboard Page . . . . .	18
5.14 Profile Page . . . . .	18
5.15 Transactions Page . . . . .	19

# CHAPTER 1

## INTRODUCTION

### 1.1 OVERVIEW

McQueen is a cutting-edge online automotive marketplace that redefines the car-buying experience by seamlessly merging innovation with reliability. This platform is designed to be the ultimate destination for individuals seeking a hassle-free and transparent journey into the world of vehicle ownership. At the heart of McQueen is a commitment to delivering unparalleled service, ensuring that users discover not just a car but an immersive driving experience.

One of McQueen's key strengths lies in its innovative approach to car buying. The platform leverages state-of-the-art technology to provide users with a seamless and intuitive interface, making it easier than ever to explore, compare, and purchase vehicles. With a vast inventory comprising the latest models and meticulously maintained pre-owned cars, McQueen caters to diverse preferences and budgets. The emphasis on innovation ensures that users embark on a journey where every click brings them closer to finding their dream car.

What sets McQueen apart is its dedication to reliability and trust. The platform is more than just a marketplace; it's a destination where users can trust the information provided, ensuring a confident and informed decision-making process. Whether users are in search of the latest models or high-quality pre-owned vehicles, McQueen positions itself as the go-to platform for quality, trust, and exceptional service. As users explore the joy of driving with McQueen, they discover a reliable partner in their automotive endeavors, making every journey on the road a memorable experience.

### 1.2 PROBLEM STATEMENT

In the rapidly evolving landscape of the automotive industry, there's a growing demand for a modernized and user-centric online platform dedicated to simplifying the complex processes associated with buying and selling cars. The traditional methods prevalent in the market often prove time-consuming, lack transparency, and fall short of offering a diverse array of choices to users. This inefficiency is further exacerbated by the absence of a centralized and streamlined online marketplace, resulting in a fragmented experience for both buyers and sellers alike. Recognizing

these challenges, our ambitious objective is to introduce a revolutionary Cars Buying and Selling website that redefines the entire automotive transaction process, serving as a comprehensive and convenient hub for users navigating the complexities of vehicle transactions.

At the core of our vision is the commitment to providing a centralized and efficient solution, bridging the gap between buyers and sellers in the automotive sphere. The envisioned platform will not only simplify complex transactions but also offer an extensive array of choices for users. By prioritizing user convenience, trustworthiness, and transparency, our goal is to revolutionize the conventional methods of car transactions. This Cars Buying and Selling website seeks to establish itself as the go-to destination, ushering in a new era where individuals can engage in car transactions with confidence and ease, setting a new standard for the automotive marketplace.

### 1.3 ORGANIZATION OF THE PROJECT

**1}Defining the problem:** The problem to be solved was clearly defined to allow for the efficient usage of our time by preventing the need for any changes after starting the project.

**2}Task Allocation:** Assign specific tasks to team members based on their expertise and project requirements. Ensure a balanced distribution of workload to optimize productivity.

**3}Technology Stack and Tools:** Confirm the technology stack, including the MERN components (MongoDB, Express.js, React, Node.js). Select appropriate development tools, version control systems, and collaboration platforms.

**4}Checks for code issues:** The code is checked for issues using Github actions to ensure the code's quality.

**5}Reusable Web components:** To ensure the reusability of the components created for this project for any future projects, various reusable components are created using React and Bootstrap.

# **CHAPTER 2**

## **SYSTEM REQUIREMENTS**

### **2.1 SOFTWARE REQUIREMENTS**

- Operating System: Windows, Linux, Mac
- Any Chromium/Firefox-Based Browser

### **2.2 HARDWARE REQUIREMENTS**

- x86, 64-bit CPU (Intel / AMD architecture), Arm CPU
- 2 GB RAM
- 5 GB free disk space.
- Stable Internet Connection

# **CHAPTER 3**

## **LITERATURE SURVEY**

### **3.1 MongoDB**

MongoDB, a No-SQL database, plays a pivotal role in the MERN stack. Researchers and practitioners highlight its capability to handle large volumes of unstructured data efficiently. The flexibility of MongoDB's document-oriented structure is well-documented, enabling seamless storage and retrieval of diverse data types. Studies emphasize its scalability, which aligns with the project's objective of accommodating a growing user base and expanding content.

### **3.2 Express**

Express a web framework for Node, is extensively explored in the literature for its role in server-side development. Scholars acknowledge its minimalist approach, providing a robust foundation for building web applications. The middleware support in Express is particularly praised, enabling efficient handling of HTTP requests. The project's utilization of Express aligns with the trend of leveraging lightweight frameworks for scalable server-side development.

### **3.3 React**

React is a free and open-source front-end JavaScript library. It is used to create interactive user interfaces. React is very popular, and it is used by some of the largest tech companies in the world, including Facebook, Instagram, and Netflix.

### **3.4 Node**

Node the JavaScript run-time environment, is extensively studied for its event-driven, non-blocking I/O model. Literature acknowledges its efficiency in handling concurrent connections, making it suitable for real-time applications. The project's choice of Node aligns with the broader trend of using JavaScript on both the client and server sides, fostering code consistency and streamlined development.

### **3.5 Javascript**

JavaScript is the programming language of the Web. It is used to create interactive web pages and web applications. JavaScript is also a powerful tool for data visualization and animation.

### **3.6 JSON Web Tokens (JWT)**

JSON Web Tokens (JWT) have gained prominence in the literature as a compact, URL-safe means of representing claims between two parties. Researchers discuss the advantages of using JWT for authentication and information exchange. The token's simplicity, scalability, and ease of implementation are often highlighted, showcasing its relevance in secure communication protocols.

### **3.7 Multer**

Multer is a popular middleware for handling multipart/form-data, which is primarily used for uploading files in Node applications, offering flexibility, configuration options, and seamless integration with Express. It is particularly well-suited for scenarios where you need to handle file uploads, such as images or documents, from forms on the client side.

### **3.8 Axios**

Axios is a popular JavaScript library for making HTTP requests from web browsers and Node.js environments. It simplifies the process of sending asynchronous HTTP requests, allowing developers to interact with APIs effortlessly. Axios supports various features, including request and response interception, the ability to cancel requests, and the automatic transformation of JSON data. Its straightforward syntax and promise-based architecture make it a preferred choice for handling HTTP requests in modern web applications. Whether fetching data from a server or posting updates, Axios provides a clean and efficient way to manage HTTP communication within a web development project.

# **CHAPTER 4**

## **IMPLEMENTATION**

The website's front end is built using React, which is a widely used JavaScript library for developing user interfaces, alongside HTML and CSS. To handle routing within the website, the react-router-dom library is utilized. Additionally, it makes use of templates from 'bootstrap' for styling to make the website's design cleaner and attractive.

The application code is designed to be modular and reusable, wherein the components can be reused across other React projects and potentially even allow scalability.

The backend server is implemented using MongoDB, Express JS, and Node JS. The server handles the processing and serving of data to the front end in JSON format. The website, for one, utilizes an API from 'RapidAPI.com' to retrieve real-time (and updated) new-car data.

The project's structure is as follows:-

```
Front-End
├── .git
├── .vscode
│   └── settings.json
├── node_modules
└── public
    ├── bootstrap-4.3.1-dist
    │   └── css
    │       ├── bootstrap.css
    │       ├── bootstrap.css.map
    │       ├── bootstrap.min.css
    │       ├── bootstrap.min.css
    │       ├── bootstrap.min.css
    │       ├── bootstrap.min.css.map
    │       ├── bootstrap-grid.css
    │       ├── bootstrap-grid.css.map
    │       ├── bootstrap-grid.min.css
    │       ├── bootstrap-grid.css.min.map
    │       ├── bootstrap-reboot.css
    │       └── bootstrap-reboot.css.map
```

```
├── bootstrap-reboot.min.css
│   └── bootstrap-reboot.min.css.map
└── js
    ├── bootstrap.bundle.js
    ├── bootstrap.bundle.js.map
    ├── bootstrap.bundle.min.js
    ├── bootstrap.bundle.min.js.map
    ├── bootstrap.js
    ├── bootstrap.js.map
    ├── bootstrap.min.js
    └── bootstrap.min.js.map

```

```
└── img
    ├── portfolio
    │   ├── ev.jpg
    │   ├── regular.jpeg
    │   └── sportscar.jpeg
    ├── team
    │   └── team.png
    ├── back.mp4
    ├── backback.mp4
    ├── icon.ico
    ├── logo.png
    ├── map-image.png
    ├── title.png
    ├── title-b.png
    ├── title-logo.png
    └── vehicle.mp4
    └── index.html
    └── scripts.js
    └── styles.css

```

```
└── src
    └── components
        ├── CheckOut.js
        ├── Dashboard.js
        ├── Dashboard.js
        └── Landing.js

```

```
    └── Profile.js
    └── Purchase.js
    └── Search.js
    └── Sell.js
    └── SignIn.js
    └── SignIn1.js
    └── SignUp.js
    └── Transactions.js
    └── App.js
    └── index.css
    └── index.js
    └── .gitignore
    └── package.json
    └── package-lock.json
```

```
Back-End
└── .git
└── models
    └── purchases.js
    └── sales.js
    └── userModel.js
└── node_modules
└── .gitignore
└── index.js
└── package.json
└── package-lock.json
```

## 4.1 Frontend Development

McQueen's front-end development involved a meticulous crafting process that utilized a combination of React, HTML, CSS, and Bootstrap. The primary emphasis was on creating a visually appealing and user-friendly interface. The design approach centered around simplicity and responsiveness, aiming to provide users with an effortless navigation experience across the platform's diverse features.

```

    app > src > App.js
      App.js
      ...
      > open editors
      > untitled (workspace)
        > app
          > .vscode
          > node_modules
        > public
          > bootstrap-4.3.1-dist
          > img
        > index.html
        > script.js
        # styles.css
      < src
        > components
          > App.js
            # index.css
            > index.js
            .gitignore
            > index.html
            package.json
            README.md
            tailwind.config.js
            > xyz.html
        < server
          > car_images
        < models
          > purchases.js
          > sales.js
          > userModels.js
        < node_modules
        > payments
        < .gitignore
        > index.js
        package-lock.json
        package.json
      > outline
      > timeline
      > main
      > 0.0.0
      > 0
      > Live share
      In 19, Col 42 | Spaces: 2 | UTF-8 | CR/LF | JavaScript | Go Live | □
  
```

Figure 4.1: Main Page code snippet

To enhance interactivity and ensure a seamless user experience, dynamic elements were incorporated using JavaScript. These elements not only added a layer of interactivity but also contributed to the overall fluidity of the platform. The integration of these technologies collectively resulted in a well-crafted front end for McQueen, combining aesthetic appeal with functional efficiency.

## 4.2 Backend Development

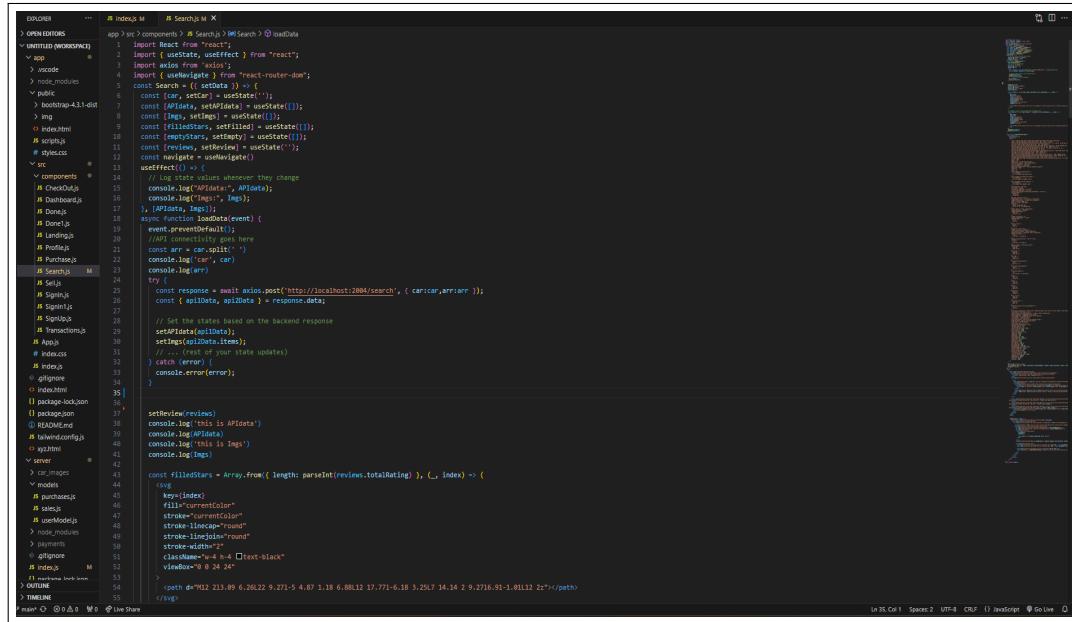
```

    server > A index.js
      index.js
      ...
      > open editors
      > untitled (workspace)
        > app
          > .vscode
          > node_modules
        > public
          > bootstrap-4.3.1-dist
          > img
        > index.html
        > script.js
        # styles.css
      < src
        > components
          > App.js
            # index.css
            > index.js
            .gitignore
            > index.html
            package.json
            README.md
            tailwind.config.js
            > xyz.html
        < server
          > car_images
        < models
          > purchases.js
          > sales.js
          > userModels.js
        < node_modules
        > payments
        < .gitignore
        > index.js
        package-lock.json
        package.json
      > outline
      > timeline
      > main
      > 0.0.0
      > 0
      > Live Share
      In 13, Col 1 | Spaces: 2 | UTF-8 | CR/LF | JavaScript | Go Live | □
  
```

Figure 4.2: Backend code snippet

The Node.js application, built on the Express framework, serves as the backend for McQueen, a web application dedicated to facilitating car transactions. Dependencies such as Axios for HTTP requests, Cors for cross-origin resource sharing, Mongoose for MongoDB connectivity, and Bcrypt for password hashing are incorporated. The application defines various routes, including those for user registration (/signup), user login (/signin), posting car purchases (/purchase), posting car sales (/sell), fetching user transactions (/transactions/:email), and searching for car-related data (/search). Bcrypt is utilized to hash user passwords before storing them in the database, and JSON Web Tokens (JWT) are employed for user authentication. Additionally, Multer is integrated for handling file uploads, and Nodemailer is used for sending emails.

### 4.3 Search Cars



A screenshot of a code editor showing the 'Search.js' component code. The code uses React hooks like useState and useEffect. It makes an axios POST request to '/search' with a car model as data. The response is then processed to set state and generate a star rating visualization using SVG.

```

import React, { useState, useEffect } from "react";
import axios from "axios";
import { useNavigate } from "react-router-dom";
const Search = ({ carData }) => {
  const [car, setCar] = useState("");
  const [APIData, setAPIData] = useState([]);
  const [reviews, setReviews] = useState([]);
  const [rating, setRating] = useState(0);
  const [filledStars, setFilledStar] = useState(0);
  const [emptyStars, setEmptyStar] = useState(5);
  const [loading, setLoading] = useState(false);
  const [error, setError] = useState(null);
  const [isErrorVisible, setIsErrorVisible] = useState(false);
  useEffect(() => {
    // Log state values whenever they change
    console.log("APIData:", APIData);
    console.log("Reviews:", reviews);
    console.log("Rating:", rating);
  }, [APIData, reviews]);
  async function loadData(event) {
    event.preventDefault();
    const carModel = document.querySelector("#car").value;
    const arr = carModel.split(" ");
    console.log(`car: ${arr}`);
    const response = await axios.post(`http://localhost:2004/search`, { car: arr[0] });
    const { APIData, reviews } = response.data;
    setAPIData(APIData);
    setReviews(reviews);
    // Set the states based on the backend response
    setRating(rating);
    setFilledStars(filledStars);
    setEmptyStars(emptyStars);
    setLoading(false);
    if (error) {
      setError(null);
    }
    console.log(`Logs: ${logs}`);
  }
  return (
    <div>
      <input type="text" value={car} onChange={e => setCar(e.target.value)} placeholder="Search a car" />
      <button onClick={loadData}>Search</button>
      <div>
        <div>
          <span>{rating}</span>
          <span>{filledStars}</span>
          <span>{emptyStars}</span>
        </div>
        <div>
          <span>{APIData}</span>
          <span>{reviews}</span>
        </div>
      </div>
    </div>
  );
}

```

Figure 4.3: Search cars code snippet

The provided React component, named "Search," is a crucial part of the McQueen project's front end, responsible for handling user searches and displaying relevant car information. The component employs the useState and useEffect hooks from the React library to manage the component's state and handle side effects, respectively. Upon a user entering a search query, the component triggers an asynchronous function, loadData, using the Axios library to make a POST request to the server's /search endpoint. This endpoint communicates with external APIs to gather information about the specified car, such as its model, make, and images.

The retrieved data is then stored in the component's state variables, APIdata and Img, and subsequently logged.

The rendered JSX includes a search bar, allowing users to input their car search queries. The user's input is captured using the setCar function, and upon triggering the search, the loadData function is called. The retrieved API data is mapped over to dynamically generate sections displaying car details, such as the model, make, and images. Users can view pertinent information about the cars, including their fuel type, year, and pricing. The "Check Out" button is linked to the handleCheckout function, enabling users to proceed to the checkout page with selected car details. Additionally, the component utilizes SVG icons to represent star ratings based on user reviews, creating an informative and visually appealing user interface for the McQueen platform.

#### 4.4 External APIs Integration

The Node.js application incorporates several API integrations within the McQueen project. In particular, the /search route showcases the utilization of three external APIs to fetch relevant car-related data. The first API (<https://cars-by-api-ninjas.p.rapidapi.com/v1/cars>) retrieves details about cars based on specified parameters such as make and model. The second API (<https://google-search72.p.rapidapi.com/imagesearch>) is employed to obtain images related to the provided car search query. The third API (<https://cars-data3.p.rapidapi.com/cars-data>), although currently commented out, seems intended for retrieving additional details about cars. These API responses are then consolidated and sent as a JSON object in the HTTP response to the client, providing comprehensive information for the user's car search within the McQueen platform.

The integration of these APIs enhances the user experience by enriching the platform with diverse and dynamic data, ranging from car specifications to visual representations. This approach allows McQueen to offer users a more comprehensive and visually engaging presentation of car-related information, contributing to a more informed and satisfying user interaction. The commented-out section suggests that there might be considerations or potential plans for incorporating additional data from the third API in the future development of McQueen.

# CHAPTER 5

## RESULTS

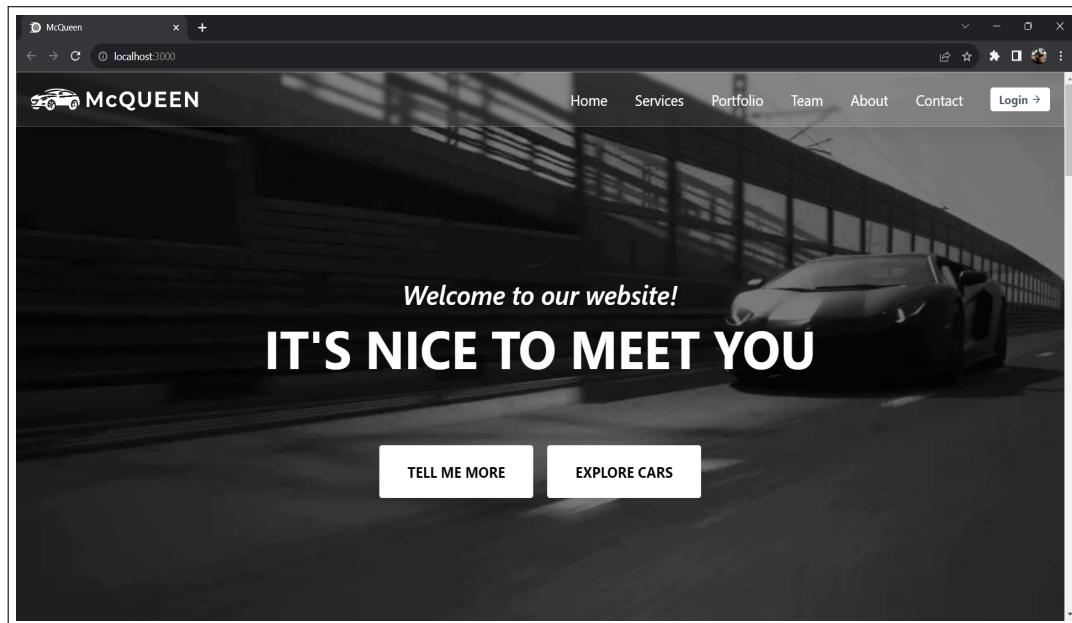


Figure 5.1: Landing Page

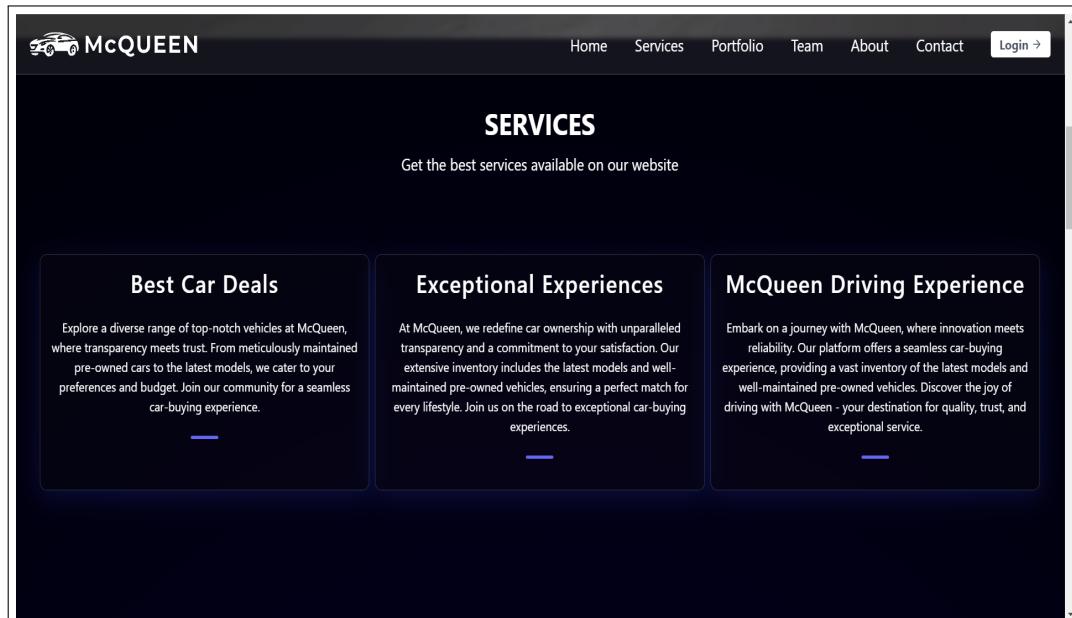


Figure 5.2: Services Section

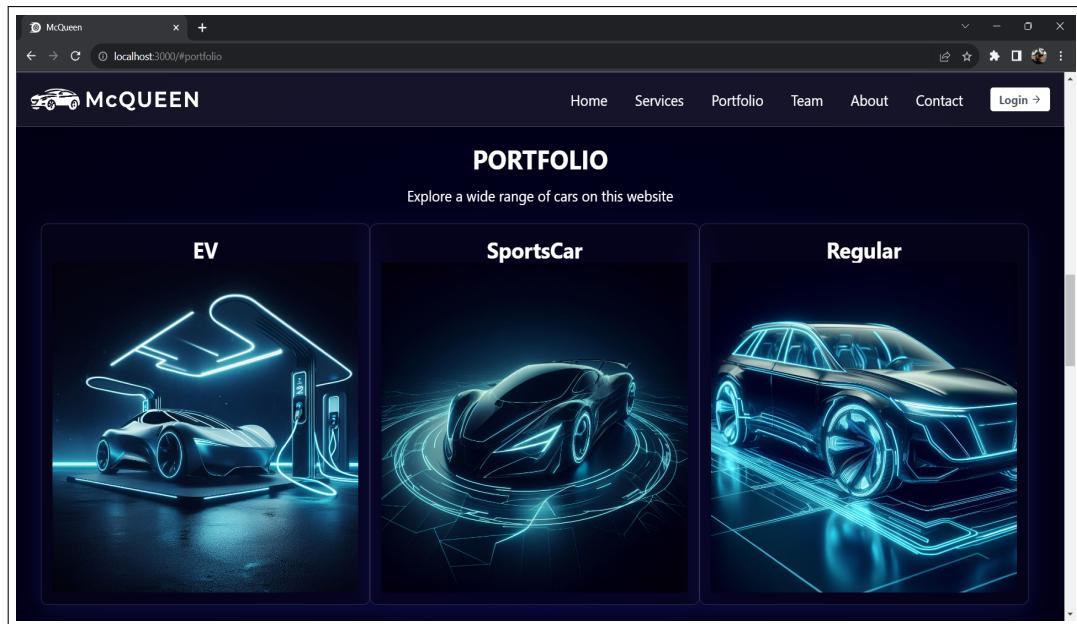


Figure 5.3: Portfolio Section

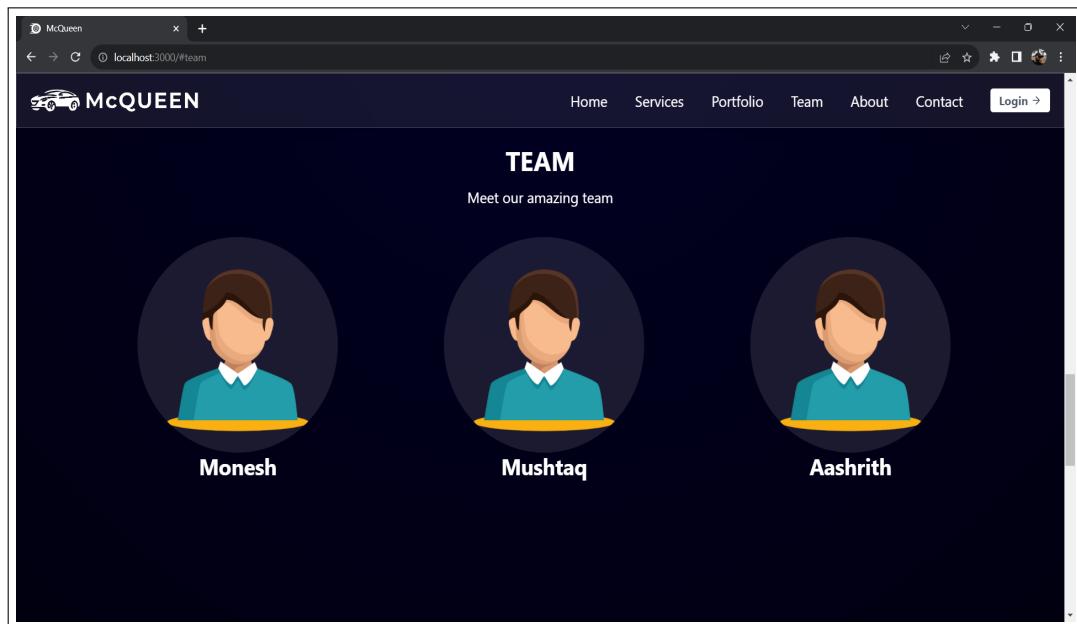


Figure 5.4: Team Section

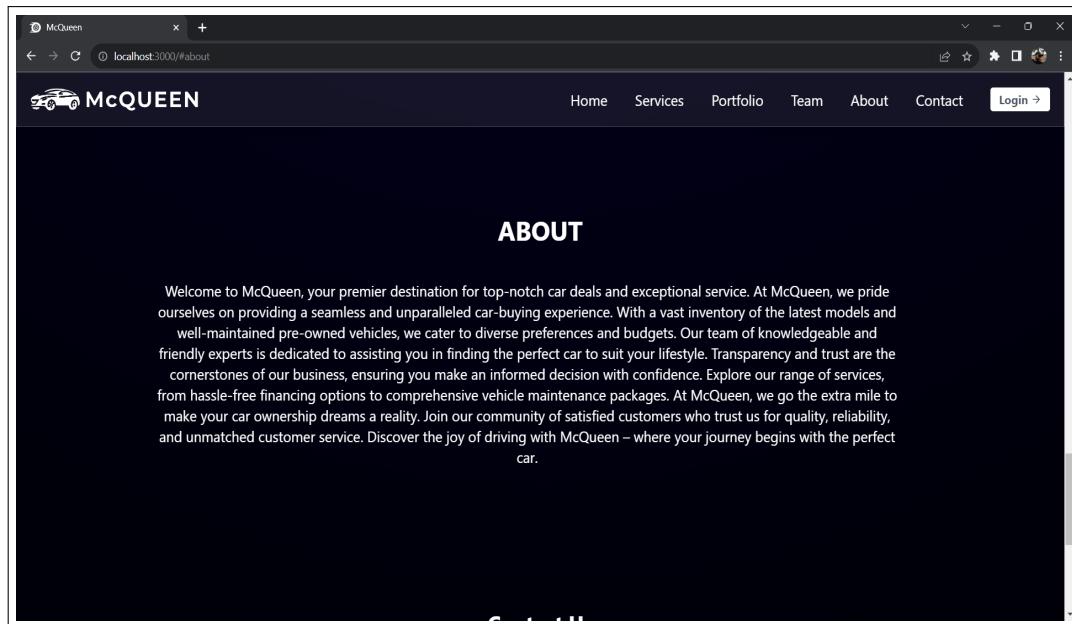


Figure 5.5: About Section

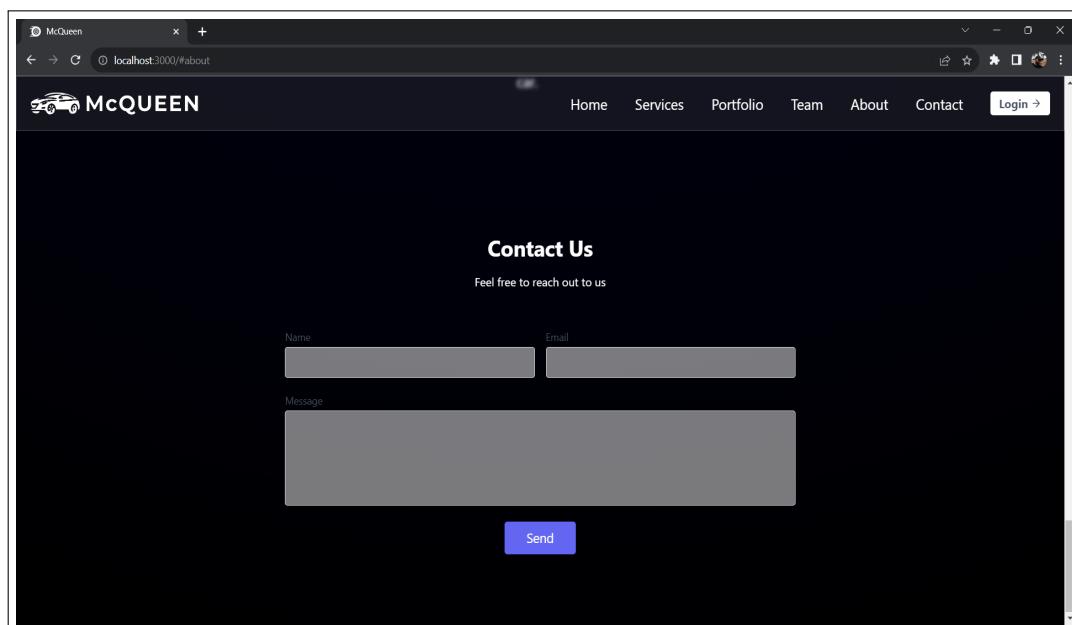


Figure 5.6: Contact Us Section

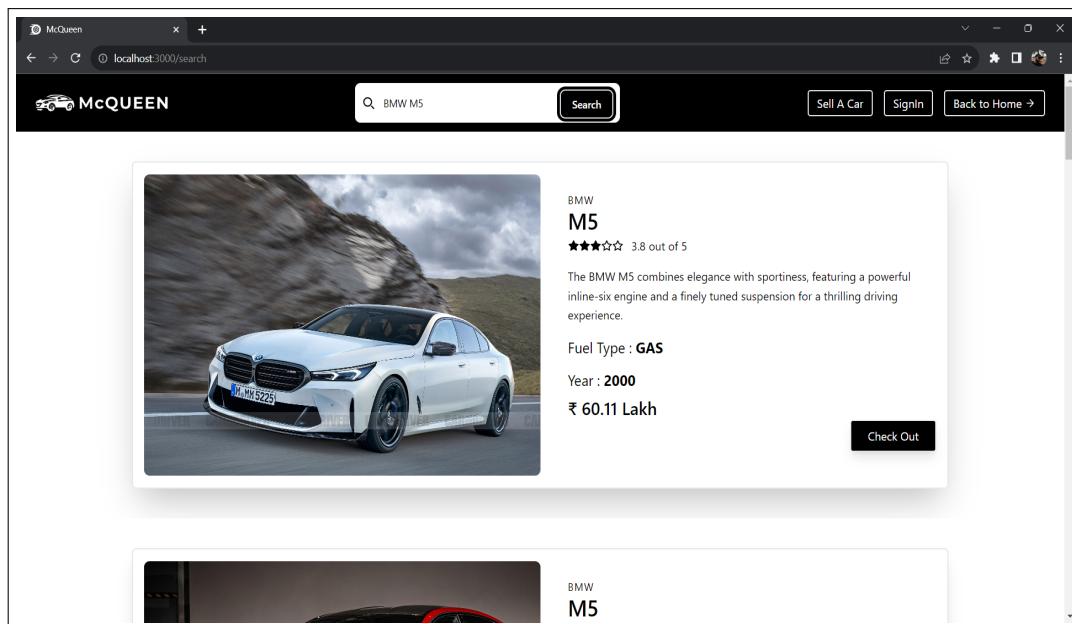


Figure 5.7: Explore Cars Page

FEATURE	RATING
Comfort	3.8
Driving	4
Utility	3.5
Interior	3.7
Technology	4
Performance	4.1
Quality	3.8
Reliability	3.6
Styling	3.9
Value	3.4

**BMW M5 Specifications**

ENGINE AND TRANSMISSION      FUEL AND PERFORMANCE      SUSPENSION AND STEERING      DIMENSIONS AND CAPACITY

Figure 5.8: Checkout Page

The screenshot shows a web browser window titled "McQueen" with the URL "localhost:3000/purchase". The page has a dark header with the "McQUEEN" logo and navigation links for "Home" and "Contact". The main content area is titled "Verification" and contains instructions: "Please provide the following details". It includes fields for "Name" (text input), "Email" (text input), "Phone" (text input), and a file upload field for "Token Payment Screenshot (Rs. 5000)" with a "Choose File" button and a message "No file chosen". There is also a large "Message" text area. At the bottom is a black "Button".

Figure 5.9: Verification Page

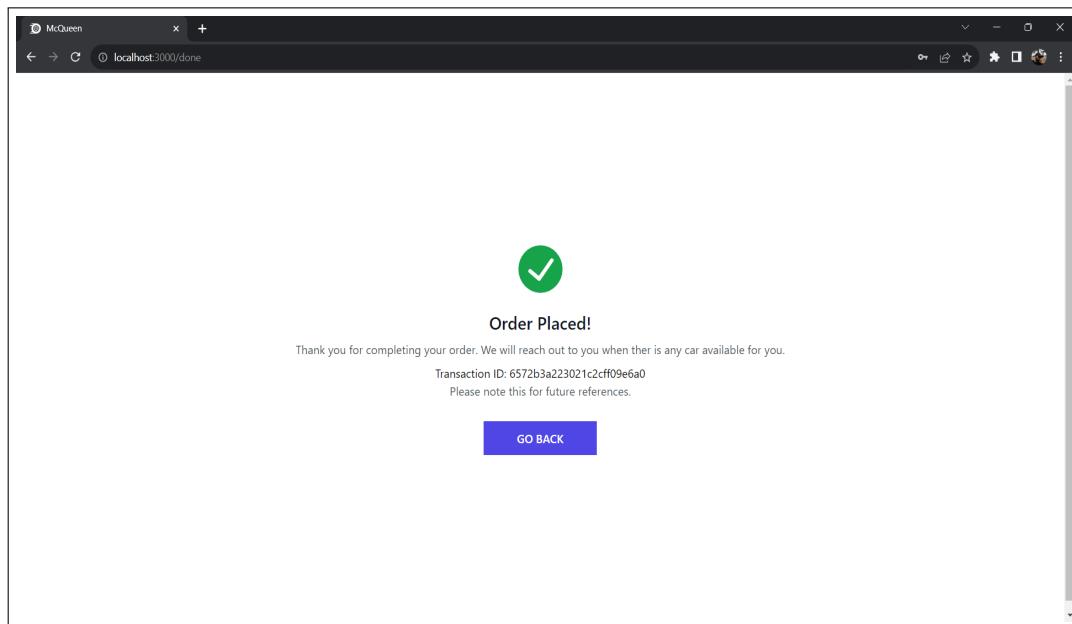


Figure 5.10: Confirmation Page

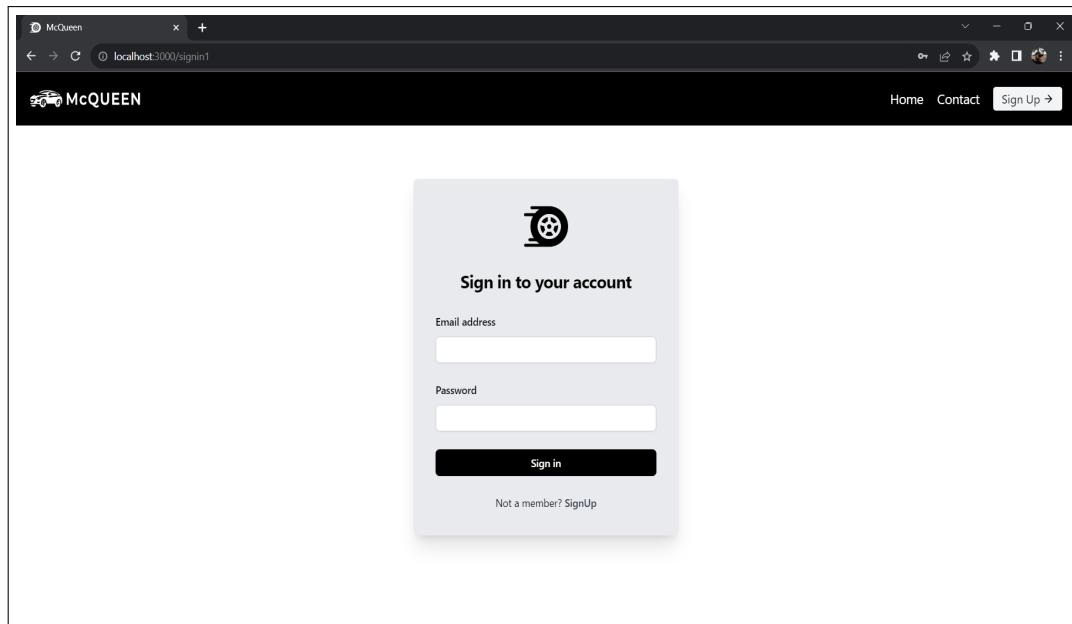


Figure 5.11: Sign In Page

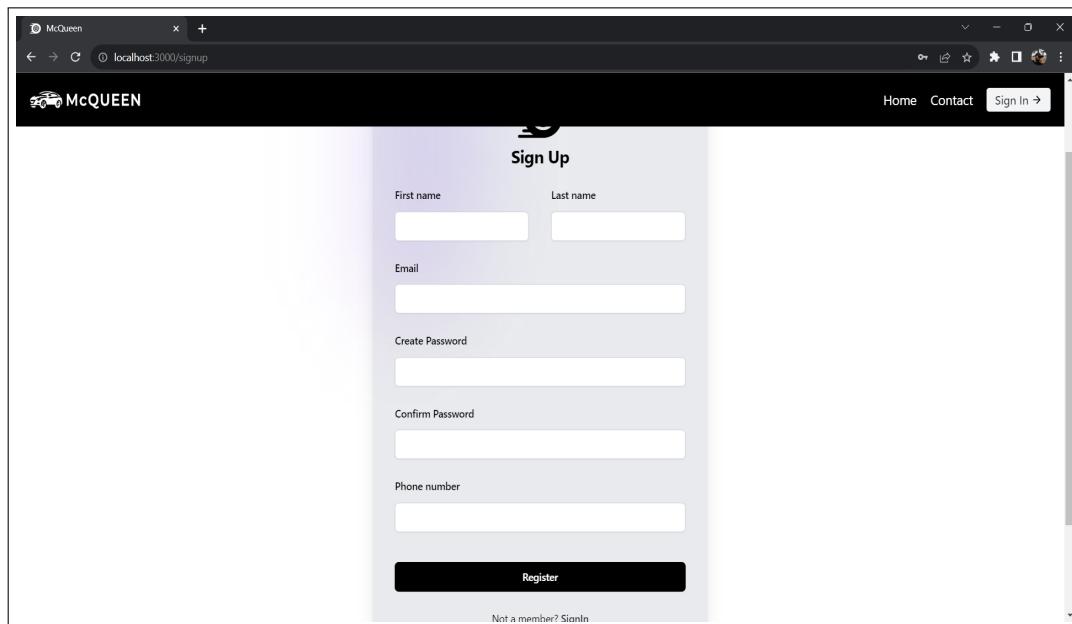


Figure 5.12: Sign Up Page

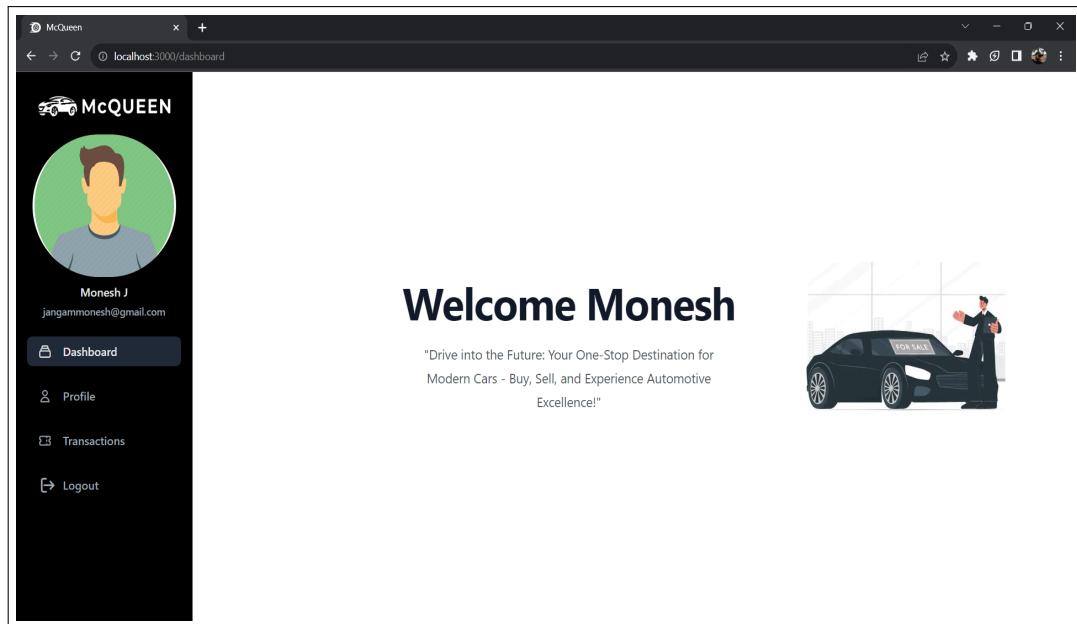


Figure 5.13: Dashboard Page

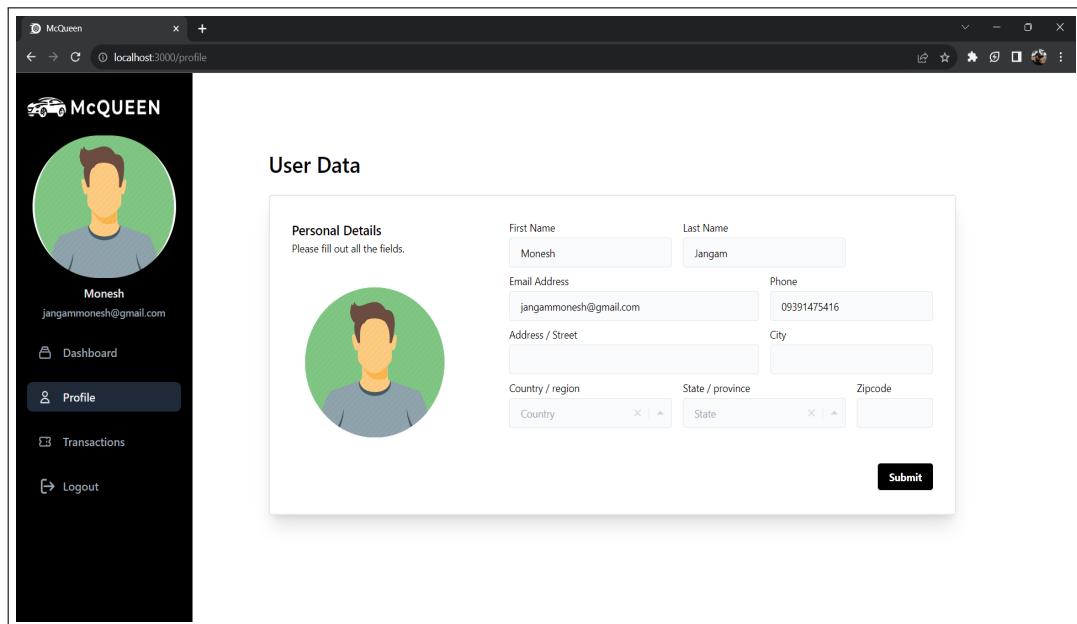


Figure 5.14: Profile Page

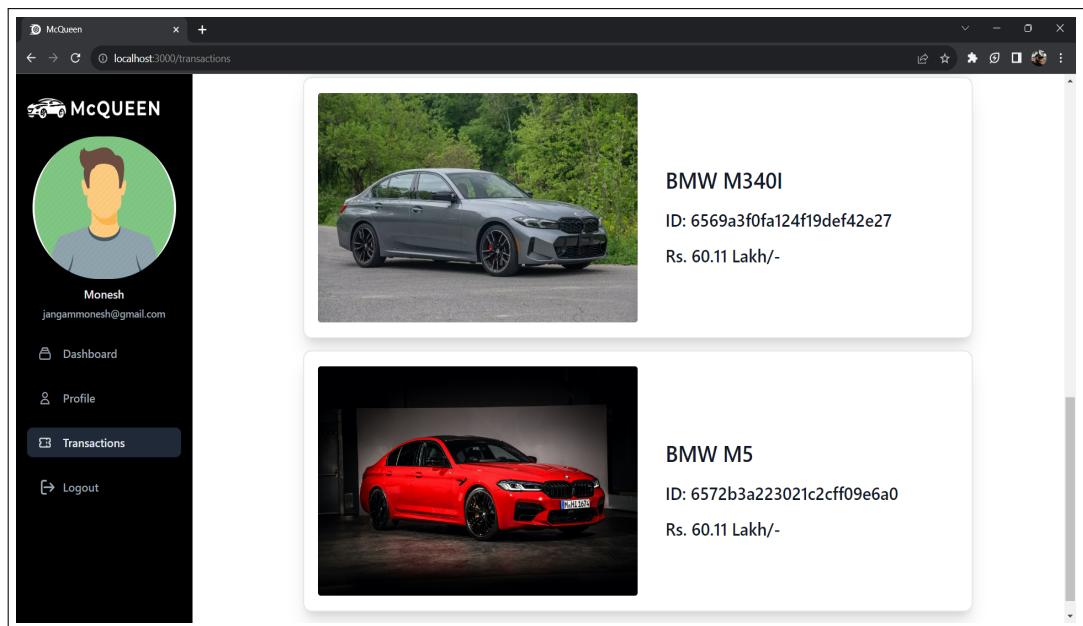


Figure 5.15: Transactions Page

The transactions the user made will be displayed here.

## **CHAPTER 6**

### **FUTURE SCOPE**

The future scope for our project is vast and holds the potential for transformative growth in the automotive marketplace. First and foremost, the integration of emerging technologies stands as a key avenue for future development. Artificial intelligence (AI) and machine learning (ML) could be leveraged to enhance the user experience by providing personalized recommendations, predictive analytics for pricing trends, and intelligent search functionalities. This would not only streamline the process further but also add a layer of sophistication to the platform.

Expanding the services offered is another promising aspect of the project's future scope. The platform can evolve into a comprehensive automotive ecosystem by incorporating features related to financing, insurance, and maintenance services. This holistic approach would cater to the diverse needs of users, creating an all-encompassing solution for their automotive requirements. Additionally, exploring opportunities for global expansion could open new markets and audiences, requiring adaptation to regional preferences and regulations.

Furthermore, the continuous evolution of the platform could involve the development of a dedicated mobile application. As mobile usage becomes increasingly prevalent, a mobile app would offer users the flexibility to engage with the platform on-the-go, providing a more convenient and accessible experience. Overall, the project's future scope revolves around embracing technological advancements, expanding services, exploring new markets, and ensuring seamless user accessibility to solidify its position as an innovative leader in the ever-evolving automotive industry.

## **CHAPTER 7**

## **CONCLUSION**

In conclusion, McQueen stands as a pivotal initiative poised to revolutionize the automotive marketplace. By addressing the inadequacies of traditional methods, our platform envisions a future where the process of buying and selling cars is characterized by convenience, transparency, and user-centricity. Through the integration of cutting-edge technologies, the project aspires to continually enhance the user experience, providing personalized solutions and intelligent insights. The envisioned expansion into comprehensive automotive services and potential global outreach further solidifies its role as an innovative leader in the industry. As we embark on this journey, the project remains committed to shaping the future of car transactions, setting a new standard for efficiency, reliability, and user satisfaction in the dynamic landscape of the automotive market.