A Real Time Research Project/ Societal Related Project Report

On

Rent Ease: An AI Powered Online Rental System

Submitted in fulfillment of the requirements for the award of the

Bachelor of Technology

In

Department of Computer Science and Engineering

By

A Divya Sri	22241A05D1
B Susanna	22241A05D9
K Manju Sri	22241A05F5
M Chaithra	22241A05G2

Under the Esteemed guidance of

Dr. G. Charles Babu

Professor

Department of Computer Science and Engineering

GOKARAJU RANGARAJU INSTITUTE OF ENGINEERING AND TECHNOLOGY

(Autonomous)

Bachupally, Kukatpally, Hyderabad, Telangana, India, 500090

GOKARAJU RANGARAJU

INSTITUTE OF ENGINEERING AND TECHNOLOGY

(Autonomous)

CERTIFICATE

This is to certify that the Real Time Research Project/ Societal Related Project entitled "**Rent Ease: An AI Powered Online Rental System**" is submitted by **A Divya Sri(22241A05D1), B Susanna(22241A05D9), K Manju Sri(22241A05F5), M Chaithra(22241A05G2)** in fulfillment of the award of a degree in BACHELOR OF TECHNOLOGY in Computer Science and Engineering during the academic year **2023-2024.**

INTERNAL GUIDE

Dr. G. CHARLES BABU

Professor

HEAD OF THE DEPARTMENT

Dr. B. SANKARA BABU

Professor

ACKNOWLEDGEMENT

Many people helped us directly and indirectly to complete our project successfully. We would like to take this opportunity to thank one and all. First, we wish to express our deep gratitude to our internal guide **Dr. G. Charles Babu, Professor,** Department of CSE for his/her support in the completion of our project report. We wish to express our honest and sincere thanks to **Dr. V. Srilakshmi and Ms. S. Ritwika** for coordinating in conducting the project reviews, **Dr. B. Sankara Babu, HOD,** Department of CSE for providing resources, and to the principal **Dr. J. Praveen** for providing the facilities to complete our Real Time Research Project/ Societal Related Project. We would like to thank all our faculty and friends for their help and constructive criticism during the project completion phase. Finally, we are very much indebted to our parents for their moral support and encouragement to achieve goals.

A Divya Sri **(22241A05D1)**
B Susanna **(22241A05D9)**
K Manju Sri **(22241A05F5)**
M Chaithra **(22241A05G2)**

DECLARATION

We hereby declare that the Real Time Research Project/ Societal Related Project entitled "**RentEase: An AI Powered Online Rental System**" is the work done during the period from **2023-2024** and is submitted in the fulfillment of the requirements for the award of the degree of Bachelor of Technology in Computer Science and Engineering from **Gokaraju Rangaraju Institute of Engineering and Technology (Autonomous under Jawaharlal Nehru Technology University, Hyderabad).** The results embodied in this project have not been submitted to any other university or Institution for the award of any degree or diploma.

A Divya Sri	**(22241A05D1)**
B Susanna	**(22241A05D9)**
K Manju Sri	**(22241A05F5)**
M Chaithra	**(22241A05G2)**

	Table of Contents	
Chapter	**Title**	**Page No.**
	Abstract	1
1	Introduction	3
2	System Requirements	6
	2.1 Software Requirements	5
	2.2 Hardware Requirements	5
	2.3 Data Set	6
3	Literature Survey	13
4	Proposed Model, Modules Description, and UML Diagrams	21
	4.1 Modules	17
	4.2 UML Diagrams	21
5	Implementation, Experimental Results &Test Cases	51
6	Conclusion and Future Scope	52
7	References	53
	Appendix i) Snapshot of the Result ii) Optional (Like Software Installation /Dependencies/ pseudo code)	

LIST OF FIGURES

Fig. No.	Title	Page No.
4.1	System Architecture	15
4.2.1	Use case Diagram	17
4.2.2	Class Diagram	19
4.2.3	Sequence Diagram	20
4.2.4	Component Diagram	21
5.1	Login/Signup Page	41
5.2	Admin Login Page	41
5.3	Invalid Mobile Number	42
5.4	Invalid OTP	42
5.5	Homepage	43
5.6	Furniture index page	43
5.7	Appliances index page	44
5.8	Camera index page	44
5.9	Category-wise selection	45
5.10	Category database	45
5.11	Category database with one Item	46
5.12	Entering details to add a new Item	46
5.13	Furniture products parsing through database	47
5.14	Appliances products parsing through database	47
5.15	Camera products parsing through database	48
5.16	Items added to cart	48
5.17	Empty cart page	49
5.18	Transaction page	49
5.19	Order confirmation page	50

ABSTRACT

Rent is a progressive on-line platform designed to simplify the condo method for a huge variety of gadgets, inclusive of fixtures, cameras, and digital appliances. In a technology wherein flexibility and comfort are paramount, Rent offers a complete on-line platform designed to facilitate the condo of numerous gadgets, inclusive of cameras, fixtures, and appliances. This mission leverages synthetic intelligence to decorate person experience, streamline condo processes, and optimize aid control, addressing the developing want for brief get entry to notable goods.

Our platform offers a person-pleasant interface wherein people can effortlessly browse, select, and lease gadgets primarily based totally on their unique requirements. With Rent, customers can locate notable fixtures for his or her homes, professional-grade cameras for unique occasions, and the contemporary digital appliances, all to be had for lease at aggressive costs. This flexibility permits customers to get entry to top rate gadgets without the load of ownership, making it perfect for short-time period desires and brief setups.

One of the center capabilities of Rent is its clever seek and advice system. Utilizing state-of-the-art algorithms, the platform offers customized tips and guarantees that customers can quick locate gadgets that fit their choices and condo history. This complements the general person experience, making it simpler to find and lease the suitable object.

Rent additionally contains green stock control and dynamic pricing models. These assist manipulate the provision of gadgets, expect call for, and modify condo costs in real-time, making sure that customers usually get the high-quality value. For object owners, this indicates maximizing their condo profits and decreasing idle time for his or her assets.

Rent revolutionizes the condo marketplace through offering an accessible, green, and technology-pushed answer for renting fixtures, cameras, and digital appliances. Our platform now no longer best meets the developing call for condo offerings however additionally gives a realistic and cost-powerful opportunity to purchasing, aligning with the current consumer`s want for flexibility and comfort.

CHAPTER 1
INTRODUCTION

In today’s fast-paced, dynamic world, the concept of ownership is evolving. Increasingly, people are seeking flexible and cost-effective alternatives to buying, leading to the rise of the rental economy. Our project, **RentEase**, is an innovative online rental platform designed to meet this growing demand by providing a convenient solution for renting furniture, cameras, and musical instruments.

RentEase is built on the premise that renting can be a smarter, more sustainable choice than purchasing, especially for items that may only be needed temporarily. Whether you're a photographer needing high-quality equipment for a short-term project, a musician looking to try out a new instrument, or someone furnishing a home or office with stylish and functional furniture without a long-term commitment, RentEase offers a diverse selection of items to cater to your needs.

Our platform is developed using a full-stack approach, ensuring a seamless and efficient user experience. The front-end is designed to be intuitive and responsive, providing an easy-to-navigate interface for browsing and selecting rental items. The back-end is robust and secure, handling user data, transactions, and inventory management with reliability and speed. This comprehensive development strategy ensures that RentEase not only meets but exceeds user expectations in terms of functionality and performance.

RentEase is user-friendly and designed to streamline the rental process. Customers can browse our extensive catalog, select items, and arrange for delivery and pick-up with just a few clicks. We prioritize customer satisfaction and quality, ensuring that all our products are well-maintained and in excellent condition.

Moreover, RentEase embraces the principles of sustainability by promoting the reuse of goods, thereby reducing waste and the environmental impact associated with manufacturing and disposing of these items. By choosing to rent instead of buy, our customers are contributing to a more sustainable and economically sensible lifestyle.

Join us in revolutionizing the way people access and use furniture, cameras, and musical instruments. With RentEase, you get the freedom to enjoy high-quality products on your terms, without the burden of ownership.

Existing System

The current condo system has some major technological limitations. Many landlords and property managers are still relying on outdated methods like paper-based systems and manual bookkeeping. This leads to mistakes, inefficiencies, and data loss, making it difficult for tenants to keep track of payments and maintenance requests. Plus, there's a lack of effective data analytics, which could provide valuable insights for improving tenant satisfaction and operational efficiency. Another problem is the absence of a reliable platform for comprehensive and verified peer reviews, resulting in a lack of trust among users. Security is also a concern, as many systems don't adequately protect personal information, putting tenants at risk of identity theft and fraud. Advanced security measures like encryption and two-factor authentication are not widely used.

Sustainability is another area that's often overlooked. There are few incentives for landlords and tenants to adopt eco-friendly practices. By implementing green certifications and promoting energy-efficient properties, we could attract environmentally conscious renters. Additionally, there's insufficient support for international tenants who face challenges due to language barriers and unfamiliar legal systems. Providing multilingual assistance and resources on local rental laws could make the market more accessible to foreign renters.

Gaps in Existing System:

• **Lack of Transparency:** Rental agreements and terms are often complex and not clearly communicated to renters, leading to misunderstandings and disputes.

• **Inefficient Communication:** Communication between landlords and tenants can be slow or ineffective, causing delays in addressing issues or resolving conflicts.

• **Limited Access to Information:** Renters may struggle to find accurate information about available rental properties, pricing, and neighborhood amenities.

• **Difficulty in Property Inspection:** Renters may encounter difficulties in thoroughly inspecting properties before renting, resulting in unexpected problems after moving in.

• **Security Deposit Handling:** Disputes over security deposit deductions and delays in returning deposits are common issues in rental agreements.

CHAPTER 2

SYSTEM REQUIREMENTS

2.1. Software Requirements

2.1.1. HTML

HTML or Hyper Text Markup Language, is vital for net improvement, supplying the form and layout for net content material. The system content material on internet pages, inclusive of headers, paragraphs, lists, and links, which organizes statistics approximately condominium listings, consumer profiles, and transactions in a coherent and handy manner. HTML lets in embedding multimedia factors like belongings photos, digital tours, and video testimonials, appreciably improving consumer revel in with the aid of using supplying wealthy and interactive content material. Additionally, HTML enables linking exclusive internet pages for clean navigation, permitting customers to effortlessly pass among belongings listings, their account dashboard, and help pages.

2.1.2. CSS

CSS (Cascading Style Sheets) is an essential era for net development, that specialize in styling and formatting HTML and XML documents. It permits builders to outline how factors seem on net pages through the use of selectors to goal unique factors and making use of fashion homes along with colors, fonts, margins, and layouts. CSS syntax includes selectors that explain which factors to fashion and declarations that outline the homes and values for the one factor.

2.1.3. Java Script

JavaScript is critical to improving a condo device task through permitting interactive features, shape validation, and dynamic consumer interfaces that reply to consumer moves and information changes. JavaScript`s occasion dealing with skills guide functionalities like including objects to carts and filing condo requests, even as client-facet garage mechanisms make sure overall performance through caching information. Overall, JavaScript empowers builders to create responsive, secure, and consumer-pleasant condo structures with interactive maps, dynamic content material updates, and green information management.

2.1.4. Visual Studio Code

Visual Studio Code (VS Code) is a flexible and incredibly customizable supply code editor advanced with the aid of using Microsoft, to be had for Windows, macOS, and Linux. VS Code helps a big selection of programming languages and frameworks through extensions to be had in its marketplace, which decorate capability with functions like debugging,

undertaking automation, and language-unique support. Its included terminal permits for seamless command execution in the editor, even as customizable themes, settings, and keyboard shortcuts cater to man or woman developer preferences.

2.2 Hardware Requirements

2.2.1. Processor

A multi-middle processor (e.g., Intel Core i5 or higher) is usually recommended due to the fact condominium structures frequently manage a couple of concurrent requests from customers surfing listings, making reservations, and updating availability. A successful processor guarantees that the server can manage those operations effectively without slowdowns.

2.2.2. Memory (RAM)

At least 8GB of RAM is vital to deal with the software`s operating set, which incorporates the running machine, database server, and the condominium machine software itself. Sufficient RAM prevents common disk access (swapping) and guarantees clean performance, in particular all through height utilization periods.

2.2.3. Storage

SSD storage is favored over conventional HDDs because of its quicker read/write speeds. Rental structures frequently contain common database operations (reads and writes), so SSDs assist in handing over faster reaction instances for queries and updates, enhancing average machine responsiveness.

2.2.4. Stable and High-Speed Internet

A reliable connection is essential for apartment structures that manage on-line transactions, stock updates, and person interactions in real-time. High-velocity net guarantees that statistics may be transferred quick among customers and the server, offering a clean person experience.

2.2.5. Desktop

Users gaining access to the rental system via laptop structures want to have present day laptop structures with internet browsers (e.g., Chrome, Firefox, Edge). These browsers want to be stored up to date to ensure compatibility with the presentday internet era used withinside the rental system`s frontend.

2.3 Data Set

2.3.1 Product Inventory Dataset

For the RentEase on line system, a powerful product stock dataset could encompass crucial information to control the to be had condominium merchandise efficiently. This dataset ought to consist of the Product ID to uniquely perceive every item, Product Name, and Category (consisting of Electronics, Furniture, or Tools) to facilitate clean classification. A complete Description affords important records approximately the product, at the same time as the Quantity in Stock facilitates in stock management.

2.3.2 Customer Information Dataset

A comprehensive customer information dataset is critical for coping with person information effectively. This dataset ought to consist of the Customer ID to uniquely discover every customer, at the side of their First Name, Last Name, and Email for communication purposes. It ought to additionally seize the Phone Number and Address to facilitate touch and deliveries. Additionally, it ought to keep Payment Information which include desired price strategies and billing information, Loyalty Points for praise programs, and Preferences, like desired product categories, to decorate customer enjoy and provider personalization.

2.3.3 Rental Transactions Dataset

This dataset ought to encompass the Transaction ID to uniquely discover every rental transaction, together with the Customer ID linking to the client concerned and the Product ID linking to the rented item. The dataset ought to additionally seize the Payment Status (Paid, Pending, Overdue) and Payment Method (Credit Card, Bank Transfer, etc.). Additionally, the Employee ID may be covered if body of workers help is concerned, together with the Rental Agreement ID linking to any formal agreements, and the Timestamp to document whilst the transaction turned into created.

CHAPTER 3

LITERATURE SURVEY

Title: Case Study: Multimedia Equipment Rental

Authors: Inayatul Ulya Ahyati, Evi Lestari Pratiwi, Ramadhani Noor Pratama

Published Year: 2024

Observations: The studies at the person interface modeling for an apartment statistics device highlights numerous key observations. Participants liked the streamlined manner and the readability of statistics presented, which contributed to the general person satisfaction. The inclusion of capabilities consisting of patron rankings and reviews, a couple of fee options, and notification structures had been mainly well-obtained and taken into consideration useful for reinforcing the person revel in and satisfaction.

Shortcomings: The studies paper on person interface modeling for a condominium statistics gadget the use of a person-targeted layout (UCD) method identifies numerous gaps and shortcomings. The examine shows that destiny studies need to growth the quantity of contributors and encompass people from numerous backgrounds and age organizations to achieve greater complete feedback.

Title: Vehicle Renting System

Authors:

Published Year: 2024

Observations: The machine is designed with a robust recognition on consumer satisfaction, meditated withinside the improvement of a consumer-centric internet site that helps smooth browsing, booking, and control of automobile leases throughout numerous demographics. Operational techniques are meticulously outlined, masking the whole lot from consumer registration to automobile pickup, making sure a streamlined and consumer-pleasant experience. Furthermore, the inclusion of systematic analyses consisting of feasibility research and financial exams underscores the robustness of the proposed machine, making sure it meets each technical necessities and commercial enterprise objectives.

Shortcomings: The studies paper lacks particular empirical facts or case research from present automobile condominium structures to verify its theoretical framework. Concrete examples and empirical proof may want to offer sensible insights into the demanding situations and successes of imposing such structures in real-global scenarios.

Title: Property Rental Management System

Authors: Riya Garg

Published Year: 2023

Obsevations: The studies paper on the auto condo gadget outlines the layout and implementation of a comprehensive, web-primarily based totally platform to automate automobile reservation and condo management. Key capabilities encompass consumer registration, on-line automobile reservations, and automated database updates, specializing in user-pleasant interfaces and sturdy blunders handling. The technique consists of unique feasibility research and using Gantt charts for task tracking, making sure green and stable statistics management.

Shortcomings: The scalability of the gadget in dealing with huge volumes of records and customers concurrently isn't very well examined or discussed, that is vital for real-global application.

Title: Design and Implementation of a House Rental Management System: A Comprehensive Approach

Authors: Aposika Francis, Charles Nsiah Frimpong

Published Year: 2023

Observations: It makes a speciality of developing a consumer-pleasant platform that automates key condominium processes, complements transparency, and improves basic performance withinside the condominium market. The device automates critical responsibilities inclusive of rent control and condominium collection, lowering guide workloads and minimizing errors.

Shortcomings: There is a loss of emphasis on strong protection measures, critical for protective person records and making sure transaction integrity. Moreover, there may be no dialogue on making sure compliance with felony and regulatory necessities associated with housing and rentals, an important issue of assets control.

Title: Room Rental Application

Authors: Ms.Surbhi Khare, Shriya Samaddar, Krunal Bahoriya, Sneha Kunwar, Tejaswini

Domke, Ashu Rahangdale

Published Year: 2023

Observations: The paper ambitiously addresses the cutting-edge want for green housing answers via a cell app, focused on each landlord and tenants. It emphasizes the benefit and accessibility of the use of clever era to streamline condo processes, highlighting capabilities like belongings listings, consumer registration, and real-time communication. The absence of consumer comments or trying out effects additionally undermines the paper`s credibility in demonstrating the app's realistic usability and effectiveness.

Shortcomings: The paper lacks readability in defining its studies goals and the unique hassle declaration it aims to address, which influences its normal focus. Secondly, even as it outlines using cutting-edge technology like Android Studio and Firebase for app development, it fails to offer in-intensity technical info or code examples, making it hard for builders to duplicate or validate the proposed solution.

Title: The Rental Zone

Authors: Joy Paul

Published Year: 2022

Observations: The improvement of an internet platform geared toward simplifying the residence renting process. Initial trying out confirmed the prototype to be effective, with destiny enhancements deliberate to encompass a cellular application, improved security, AI features, and higher consumer interface design.

Shortcomings: The research paper highlights numerous gaps and shortcomings, inclusive of insufficient safety capabilities and the absence of a cellular application, that is deliberate for destiny development. There is likewise no integration of synthetic intelligence (AI) at present, that is slated for destiny enhancement.

Title: Online Rental System

Authors: Abhishek Hatwar, Vijaya Paunikar, Gauri Sayare, Shruti Ghumade, P. A. Kuchewar

Published Year: 2022

Observations: The paper provides a web-primarily based totally platform geared toward simplifying the procedure of locating condo rooms and vehicles. Key functions encompass the capacity to look for leases through place the use of Google Maps, add belongings and car

information with photos, and create reservations.

Shortcomings: It does now no longer thoroughly cope with sturdy protection capabilities vital for shielding consumer facts and making sure stable transactions. lacks an in-depth technique for integrating ongoing consumer remarks and iterative upgrades. While consumer-targeted layout is mentioned, in addition upgrades are had to cope with usability problems and completely meet the various wishes of customers.

Title: Peer to Peer Online Rental Platform

Authors: Akash S, Mehroof, Able Sobichan, Srinivas R, Nikhila G

Published Year: 2022

Observations: The uniqueness of this project is that it offers a comprehensive solution like detailed fitness and damage reports for products that enhance transparency and trust between lessors and lessees. The platform has two operation modes; direct contact mode meant for basic rentals only, while the other one is mediated service inclusive of quality checks, transportations as well as secure payment gateways. This kind of strategy takes care of customers who want minimal engagement or full-service assistance.

Shortcomings: Most of these platforms hardly ascertain users or listings thus reducing trust. They lack full user support including simple dashboards and effective dispute resolution systems. Moreover, sustainability aspect is often ignored while such consumer concerns as product damage and preference for ownership are not adequately handled.

Title: Online Rental Things

Authors: Shivendra Dwired, Vanshika Bharti, Vikas Gangwar, Mr. Krishnan

Published Year: 2021

Observation: The article gives insights on the everchanging landscape of online rental platforms and points out several things that are very important. The authors argue that many existing platforms lack intuitive interfaces, comprehensive item listings and reliable transaction processes, which leads to frustration among users and low adoption rates. As such, it becomes crucial to integrate user-friendly features like easy item listing method, strong search functionality tools as well as safe payment methods into overall usability.

Shortcomings: item quality assurance and maintenance protocols are lacking while examining these rental services with regard to legal and regulatory environments. In addition, there is a

need for investigation into market differentiation strategies, technological infrastructure scalability, consumer behavior understanding and adoption barriers.

Title: Development of Online Based Smart House Renting Web Application

Authors: Dipta Voumick, Prince Deb, Sourav Sutradhar, Mohammad Monirujjaman Khan

Published Year: 2021

Observation: The paper provides an understanding about the current state of the house rental services in Bangladesh along with its restrictions and possibilities. It establishes the need for renting properties which is driven by population growth, particularly in urban centers, noting the drawbacks of the current typical web platforms that fail to offer amenities such as location tracking and secure communication means.

Shortcomings: there are significant gaps in the location tracking and mapping systems and thus, the effective rental solution providers have the difficulties in search of ideal locations for tenants. There is limited use of secure communication networks, especially live, end-to-end secured chatting networks, which leads to communication challenges, whether or not with the house owners or other tenants and makes the system inefficient and insecure at times.

Title: Fashion Rental: Smart Business

Authors: Ciara Gyde, Lisa S McNeil

Published Year: 2021

Observations: The studies paper on fashion rental Product-Service Systems (PSS) withinside the New Zealand marketplace exhibits numerous key observations. The paper underscores that at the same time as PSS fashions in style have ability for sustainability through decreasing garment intake and waste, demanding situations continue to be in reaching good sized patron popularity and long-time period commercial enterprise viability.

Shortcomings: There may be a loss of complete frameworks to degree and benchmark the sustainability claims of style apartment businesses. While a few businesses assert their fashions are sustainable, unique operational benchmarks and verification strategies are scarce.

Title: Online Rental Housing

Authors: Sahreen Afzal, Toiba Rouf, Sumaiya Qadir, Sahila Shah

Published Year: 2021

Observations: The proposed on-line condo housing device is a well-dependent initiative aimed toward simplifying the condo assets seek and control process, specifically for college kids and employees. Forward-looking, the challenge goals for non-stop improvement, pushed with the aid of using person remarks and down to earth in a radical literature review, positioning it as a precious device for modernizing condo housing control.

Shortcomings: Advanced capabilities along with real-time updates on belongings availability or automatic fee processing aren't genuinely detailed, that may restrict its enchantment to customers watching for extra state-of-the-art functionalities. Addressing those troubles via complete testing, function enhancement, stringent protection protocols, and stepped forward accessibility measures might bolster the system`s reliability, protection, and person satisfaction.

Title: Online Vehicle Rental System

Authors: Ansh Agarwal, Rishabh Mathur

Published Year: 2020

Observations: The survey delves into the technological advancements that have reshaped the sector, emphasizing the integration of GPS tracking, mobile applications, and online payment gateways to enhance user experience and operational efficiency. Various business models employed by online vehicle rental companies, from peer-to-peer sharing to subscription-based services, are analyzed, alongside regulatory challenges such as insurance, liability, and compliance.

Shortcomings: In terms of technology, the "Online Vehicle Rental System: A Comprehensive Analysis" survey may overlook emerging tech like blockchain and AI, lack depth in discussing technical challenges, and miss innovative solutions for fleet management or user interfaces**.**

Title: LekeDe: Online Rental System

Authors: Amika Mehta, Vedant Patil, Apurva Shinde

Published Year: 2019

Observations: LeKeDe allows creation of a unique account for every individual buyer and seller, the buyer can directly contact the seller through email or over a phone call or meet in person. The need for such an application has become very crucial, time value of money is a concept greatly accepted by majority of the target audience, and money received in the present

time is worth more than the same amount received in the future due do the potential earning capacity, wherein money can earn interest.

Shortcomings: LeKeDe application is targeted to be used with any media device. The only constraint is, this application would be supported only for Android OS and not for ios. To properly tailor the security and privacy requirements, the operating system must be above API 16 i.e. Android 4.1, Jelly Bean.

Title: A Prototype of a Mobile Car Rental System

Authors: Chit Su Mon, Tan Khee Tee, Amir Aatieff Amir Hussin

Published Year: 2019

Observations: This shift goals to enhance consumer comfort with the aid of using allowing car browsing, reservation, and price via smartphones. Key functionalities consist of consumer registration, stable login, real-time car availability checks, and seamless price processing through credit score cards. Administrators advantage from sturdy manipulate over the system, dealing with car listings, patron bookings, and operational insights via included technology like Google`s Firebase for statistics storage.

Shortcomings: It lacks certain assessment and validation via real-global consumer trying out or overall performance metrics, leaving the system`s effectiveness and consumer popularity uncertain. Security concerns are cited however now no longer very well explored, with inadequate element on unique measures to shield consumer records and transactions.

CHAPTER 4
PROPOSED MODEL, MODULES DESCRIPTION AND UML DIAGRAMS

Proposed Model

To create an efficient and consumer-pleasant online rental platform, numerous key capabilities want to be implemented. The object seek characteristic must consist of a centralized seek bar with autocomplete functionality, superior filters, and sorting alternatives to assist customers locate favored rental gadgets quickly. The consumer-pleasant rental interface should have an intuitive layout with a responsive layout, clean navigation menus, and specific object pages. This interface must facilitate seamless surfing and choice of rental items, incorporating capabilities like a wishlist and mobile-pleasant access.

Additionally, a stable transaction device is important to make sure secure rental transactions, incorporating user authentication, encryption, and fraud detection mechanisms. By that specialize in those aspects, the platform can supply a reliable, stable, and enjoyable rental experience.

System Architecture

The "Online Things Rental System: Needs Following Elements Each of which interacts well with others to rent out facilities. The system will mostly be used by Buyer, who will search for items on the platform to rent them. The main responsibility of the Searching Items component is to enable the Buyer to search for the available items based on their queries so that he can view a comprehensive list of items that can be rented. When the Buyer done selection of the items require, they moved to the Place Order Component wherein they can add items to cart process the checkout.

After successful checkout, the Order Successfully Placed component informs the Purchaser that the order has been taken and displays the complete purchase funnel.

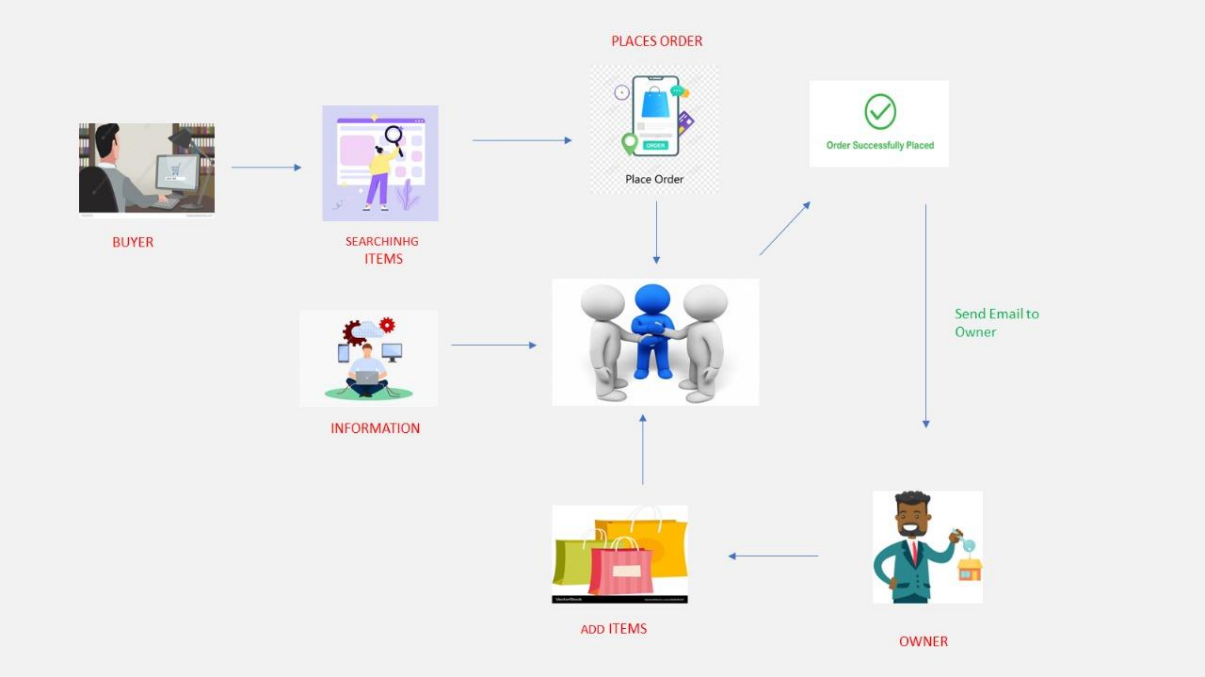

Fig.4.1 System Architecture

4.1. Modules

4.1.1. Login/Signup Module

Users can enter their mobile number to receive an OTP for login verification. The user login form accepts the admin's email, username, and password. Upon successful login, the system authenticates the admin and grants access to the admin dashboard.

4.1.2. Homepage Module

This features a navigation bar, a pop-up login form, product listings with images and pricing, and a section highlighting additional services such as free maintenance and easy returns. The JavaScript functions handle the login process and navigation to product and cart pages.

4.1.3. Category Module

Upon logging into the furnishings category, 3 alternatives are presented: add, edit, and delete.

Add: Enables the addition of a product to the category.

Upon clicking "Add", directors are brought about to go into an ID, name, photo, and price. Once added, the product is saved withinside the database at the side of its photo. Subsequently, edit and delete buttons end up visible. Clicking "Edit" activates for the ID, photo URL, name, and price.

Upon modifying and saving, the product information is up to date withinside the

respective category`s database. The edit and delete buttons are handiest displayed if as a minimum one product is gift withinside the database. Upon including a product, those buttons end up enabled for similarly action.

4.2. UML Diagrams

4.2.1. Use Case Diagram

Actors

Customer: Located at the higher left nook of the diagram with the image of stick determine characterizing men.

Administrator: Another stick determines beneath the Customer and represented the necessities signal, as in Equation (1). The use case is represented withinside the shape of ovals in the boundary of `Online Rental System` at the proper aspect of the diagram. Use case name is aligned with the actor(s) name this is close to that use case and contours are drawn among them supplying interaction.

Browse Items: The Customers also can pull out a listing of the to be had objects this is maximum appropriate for them.

Search Items: The Customer can lead the gadget to specific merchandise that she or he can be fascinated with.

View Item Details: The Customer navigate via and look at statistics concerning a selected product they`re fascinated in.

Rent Item: Through this structure, the Customer could have the possibility to hire an object.

Return Item: The Customer could have a booked object returned.

Manage Inventory: The Administrator can manipulate inventory objects which might be in use for renting.

Manage Customers: The Administrator: There is usually the Administrator who's in charge of purchaser statistics and their condo activities. The Administrator has an affiliation with the 2 use instances - Manage Inventory and Manage Customers, thus, elucidating what an Administrator is able to doing.

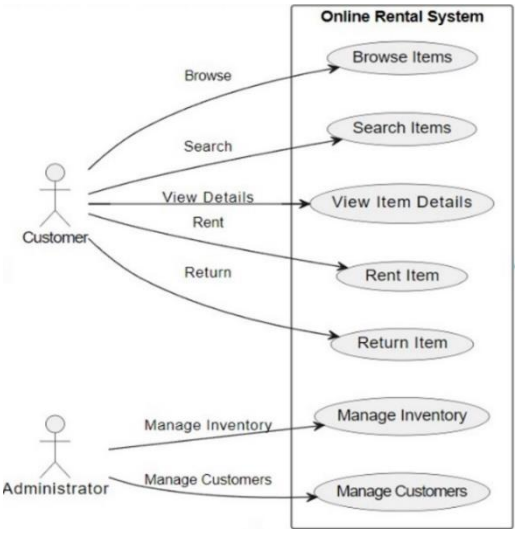

Fig. 4.2.1 Use case Diagram

4.2.2. Class Diagram

The system has four main classes: User, Item, Rental Transaction, and Payment are the key entities involved in renting out and paying for items.

· **Users:** Users can go to their accounts and log in and use the system to search the available items and also be able to rent and return items they had rented.

· **Item:** This class characterizes the items which the user is able to rent such as furniture, music instruments, and cameras.

· **Rental Transaction:** This class represent the event of renting an item when a given user rents an item for a given time.

Every transaction in the table received a unique number, as well as numbers of users and items which was rented, date of renting and returning.

· **Payment:** This class reflects the payment for the rental transactions made.

There will be one row for each payment and it has four columns as follows; The payment ID, the rental transaction ID, the amount given and the payment date.

A user can be associated with multiple rental transactions and these rental transactions are single items.

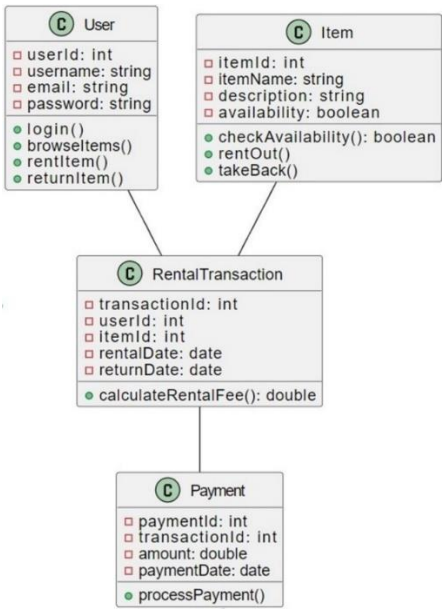

Fig. 4.2.2 Class Diagram

4.2.3. Sequence Diagram

According to the “Online Things Rental System,” the major entity of the system is called the Customer who interacts with the system through the User Interface component. Whenever a customer is browsing through items and selecting them, the User Interface sends signal to the Shopping Cart component that stocks the chosen products. The modification also occurs in parallel as the User Interface relays the new information to the Shopping Cart with the intention of mirroring the customer`s choices. If the customer selects to proceed to checkout, he clicks on the checkout button and the User Interface initiates the payment process by sending a request to the Payment Processing component.

Once an order is created and passed from the Order Management component the Payment Processing component returns a payment confirmation to the User Interface. Final, the User Interface shows the customer an acknowledgment of the order in an effort to show the details of the transaction.

This elaborate process between the Customer, User Interface and the Product Listings, Shopping Cart, Payment Processing and Order Management components ensures a well-coordinated and smooth solution and workflow within the Rented Things Online System, demonstrating the flow of information and basic operations in the rental process.

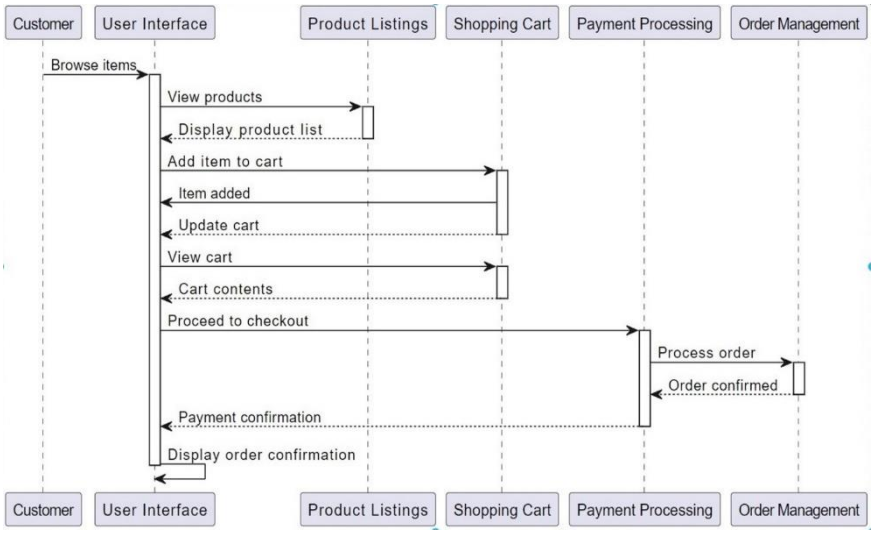

Fig. 4.2.3 Sequence Diagram

4.2.4. Component Diagram

"Component diagrams are used in identifying the internals and interactions between components within a system.

Customer: Is a user of the system who is an end user. The input/output is achieved via the User Interface where the customer communicates with the system.

User Interface: Most of them interface with several other components so that they allow the user to interact with the system.

Shopping Cart: Controls assets, which the customer wants to hire. Embeds with the Product Listings to show the available products. Interacts with the User Interface and enables clients to make different purchases to their list and/or remove items.

Order Management: Is responsible for managing all the rental orders, including entering new orders, modifying the existing ones, and monitoring the orders. Cooperates with the User Interface to take or fulfill orders from customers. Links to the Payment Processing module to process any payment transaction that may occur.

Product Listings: Oversees on the various stock of items that are in the place for being hired. In order to complete its task, Guarantee that the User Interface can show the information about products which is actual at the moment when it is requested by the customer.

Payment Processing: Controls the realizations stemming from renting equipment. Works in close collaboration with Order Management in executing payment of orders.

Customer to User Interface: The customer employs a functional User Interface of the system through which they can search for products, check out the contents of the shopping cart option, and order products.

User Interface to Shopping Cart: The Customer Interaction through the Shopping Cart enables customers to make either additions to or deletions from the cart.

User Interface to Order Management: The User Interface enables one to create, manage, and place rental orders. The Shopping Cart derives information from the Product Listings on products and services that are available.

Order Management to Payment Processing: Order Management forwards financial data regarding Payment Processing so that the latter can perform the necessary financial operations.

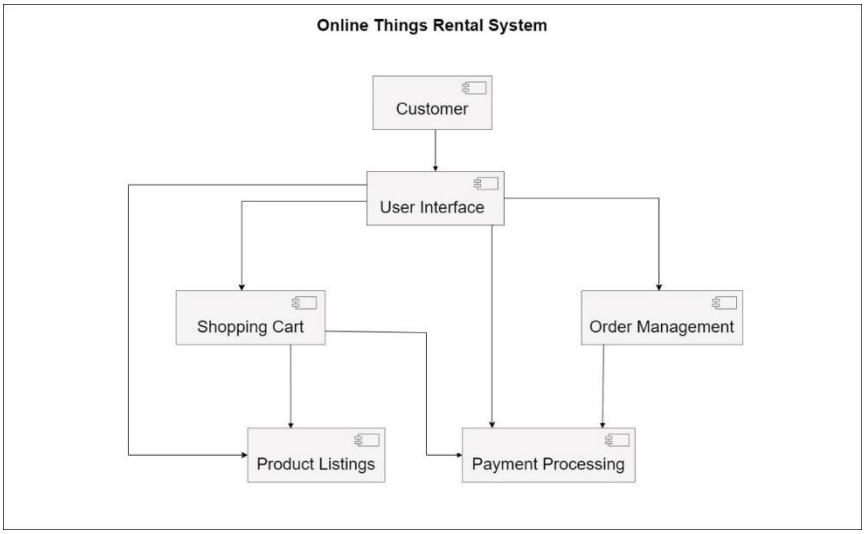

Fig. 4.2.4 Component Diagram

CHAPTER 5

IMPLEMENTATION, EXPERIMENTAL RESULTS & TEST CASES

5.1 Implementation

5.1.1 Frontend code

Index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" href="index.css">
    <title>Rent:Ease</title>
</head>
<body>
 <!-- --------  -->
<!-- Navbar section start-->
 <!-- ---------------------------------------------------------------------- -->
  <div id="navbar">
    <div onclick="goToHome()"><img src="https://is1-
ssl.mzstatic.com/image/thumb/Purple126/v4/f7/91/ef/f791ef78-2f7f-8227-e0da-
fb48bfbd117f/AppIcon-1x_U007emarketing-0-7-0-85-220.jpeg/1200x630wa.png"
alt="err">
    </div>
        <div>Rent:Ease</div>
        <div><input type="text" id="nav_search" oninput="nav_search_fn()"
             placeholder="Search for Product " >
        <!-- <input type="image" src="./search.png"> -->
        <div> <a href="cart.html"> <img src="search.png" alt=""> </a> </div>
        </div>
        <!-- <img src="./search.png" alt=""> -->
        <div><img src="https://storied-mermaid-
e7f527.netlify.app/img/cart.png"
onclick="window.open('cart.html')"></div>
        <div id="loginn"><button
id="login_btn"  onclick="show_login_popup()"><a href="login.html">LOGIN </a>
            <a href="">/ SIGNUP</a></button></div>
    </div>
    <div id="log">
        <div>
            <img src="cat.png" height="300px" width="300px" />
        </div>
        <div id="log_2">
```

```
        <div><button id="login_hide_btn">+</button></div>
        <div>
             <p>Enter your number to Login</p>
            <input type="text" id="mobileNo" placeholder="Enter Your Mobile
Number">
            <div id="otp">
                <input type="text" id="OtpNo" placeholder="Enter Your OTP">
            </div>
        </div>
        <div>
            <Button id="loginButton" onclick="alter()">Login</Button>
        </div>
        <button style="width: 50px; display: block; margin: auto; "><a
style="text-decoration: none; color: black;"
href="adminlogin.html">Admin</a></button>
    </div>
    </div>
<div id="sectionitem">
        <h2>You'll love to</h2>
        <h2 class="thin">take these home</h2>
<div id="mainProducts" onclick="window.open('Furnitureprev.html')">
   <div>
     <img src="https://p.rmjo.in/productSquare/247ark4n-500x500.jpg" alt="">
     <h5>Felix 3-Seater Fabric Sofa</h5>
     <div style="line-height: 40px;"> <span>₹ 789/month</span> <button
onclick="window.open('Furnitureprev.html')" >See more</button></div>
   </div>
   <div>
    <img src="https://p.rmjo.in/productSquare/9o9eoir3-500x500.jpg" alt="">
    <h5>Garfield Ottoman Sofa</h5>
    <div style="line-height: 40px;"> <span>₹ 189/month</span> <button
onclick="window.open('Furnitureprev.html')">See more</button></div>
  </div>
  <div>
    <img src="https://p.rmjo.in/productSquare/xal2ozbg-500x500.jpg" alt="">
    <h5>LED TV - 32"</h5>
    <div style="line-height: 40px;" > <span>₹ 949/month</span> <button
onclick="window.open('appliancesprev.html')">See more</button></div>
  </div>
  <div>
    <img src="https://p.rmjo.in/productSquare/f9pryp0h-500x500.jpg" alt="">
    <h5>Washing Machine</h5>
    <div style="line-height: 40px;"> <span>₹ 769/month</span> <button
onclick="window.open('appliancesprev.html')">See more</button></div>
  </div>
  <div >
    <img src="https://p.rmjo.in/productSquare/b28pmhld-500x500.jpg" alt="">
    <h5>Fabric Sofa</h5>
```

```
    <div style="line-height: 40px;"> <span>₹ 719/month</span> <button
onclick="window.open('Furnitureprev.html')">See more</button></div>
  </div>
</div>
</div>
 <div id="service">
    <h2>There's More</h2>
    <h2>to renting.....</h2>
    <div id="cont3">
        <div >
            <img src="https://i.ibb.co/6r032rM/sub1.jpg" alt="">
            <h3>Finest-quality products</h3>
            <p>Quality matters to you, and us! That's why we do a strict
quality-check for every product.</p>
        </div>
        <div >
            <img src="https://i.ibb.co/wMwTZPV/sub2.jpg" alt="">
            <h3>Free relocation</h3>
            <p>Changing your house or even your city? We'll relocate your
rented products for free.</p>
        </div>
        <div >
            <img src="https://i.ibb.co/ckv3sKg/sub3.jpg" alt="">
            <h3>Free maintenance</h3>
            <p>Keeping your rented products in a spick and span condition is
on us, so you can sit back and
                relax.</p>
        </div>
        <div >
            <img src="https://i.ibb.co/dW4yGF7/sub4.jpg" alt="">
            <h3>Cancel anytime</h3>
            <p>Pay only for the time you use the product and close your
subscription without any hassle.</p>
        </div>
        <div >
            <img src="https://i.ibb.co/N1W5m0K/sub5.jpg" alt="">
            <h3>Easy return on delivery</h3>
            <p>If you don't like the product on delivery, you can return it
right away—no questions asked.</p>
        </div>
        <div >
            <img src="https://i.ibb.co/MN9hLbP/sub6.jpg" alt="">
            <h3>Keep upgrading</h3>
            <p>Bored of the same product? Upgrade to try another, newer design
and enjoy the change!</p>
        </div>
    </div>
</div>
            <div class="mySlides4">
```

```
                <img src="1.jpg">
</div>
</body>
</html>
<script src="index.js"></script>
```

Furnitureprev.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="Furniturestyle.css">
  <title>Document</title>
</head>
<body>
 <div id="navbar">
  <div onclick="goToHome()"><img src="https://is1-
ssl.mzstatic.com/image/thumb/Purple126/v4/f7/91/ef/f791ef78-2f7f-8227-e0da-
fb48bfbd117f/AppIcon-1x_U007emarketing-0-7-0-85-220.jpeg/1200x630wa.png"
alt="err">
  </div>
      <div>Rent:Ease </div>
      <div>
      </div>
      <div><input type="text" id="nav_search" oninput="nav_search_fn()"
           placeholder="Search for Product " >
      <!-- <input type="image" src="./search.png"> -->
      <div> <img src="search.png" alt=""></div>
      </div>
      <div><img src="https://storied-mermaid-e7f527.netlify.app/img/cart.png"
onclick="cart_clicked()" alt=""></div>
      <div><button id="home_login_button" onclick="show_login_popup()">LOGIN /
SIGNUP</button></div>
  </div>
  <div id="header">
      <ul class="breadcrumb">
          <!-- <li><a href="#">Home</a></li> -->
          <li><a href="furnitureprev.html">Furniture</a></li>
          <li><a href="appliancesprev.html">Appliances</a></li>
          <li><a href="electronicsprev.html">Cameras</a></li>
      </ul>
      <div class="productLinksDiv">
          <ul class="productLinks">
              <!-- <li><a href="#">Product</a></li> -->
              <li><a href="fproducts.html">Furniture</a></li>
              <li><a href="aproducts.html">Appliances</a></li>
              <li><a href="eproducts.html">Cameras</a></li>
```

```
          </ul>
      </div>
  </div>
  <!-- main section -->
  <main>
      <!-- for heading and line -->
      <div class="heading">
        <h1>Browse by Room type</h1>
      </div>
      <div class="line"></div>
       <!-- for the card -->
       <div class="cardlist">
          <div class="card">
           <div class="image">
             <img
src="https://www.rentomojo.com/public/images/category/package-bg/bedroom-
v1_new.jpg" alt="error">
           </div>
           <div class="category">
              <p>Bedroom</p>
           </div>
          </div>
          <div class="card">
           <div class="image">
             <img
src="https://www.rentomojo.com/public/images/category/package-bg/living-room-
v2.jpg" alt="">
           </div>
           <div class="category">
             <p>Living Room</p>
           </div>
          </div>
          <div class="card">
            <div class="image">
              <img
src="https://www.rentomojo.com/public/images/category/package-bg/study-room-
v1.jpg" alt="">
            </div>
            <div class="category">
              <p>Work From Home(WFH)</p>
            </div>
           </div>
           <div class="card">
            <div class="image">
              <img
src="https://www.rentomojo.com/public/images/category/package-bg/dining-
v1.jpg" alt="">
            </div>
            <div class="category">
```

```
            <p>Kitchen & Dining</p>
          </div>
        </div>
        <div class="card">
          <div class="image">
            <img
src="https://www.rentomojo.com/public/images/category/package-bg/baby-
furniture.jpg" alt="">
          </div>
          <div class="category">
            <p>Baby Furniture</p>
          </div>
        </div>
        <div class="card clickable">
          <a href="fproducts.html">
            <div class="image">
              <img src="arrow.jpg" alt=""
onclick="window.open('furnitureProduct.html')">
            </div>
            <div class="category">
              <p>View All In Furniture</p>
            </div>
          </a>
        </div>
   </div>
    </main>
</body>
</html>
```

Rude.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Categories</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            margin: 0;
            padding: 0;
            display: flex;
            justify-content: center;
            align-items: center;
            height: 100vh;
            background-color: #f7f7f7;
        }
        .container {
```

```
            display: flex;
            width: 80%;
            height: 25%;
            justify-content: space-between;
        }
        .category {
            flex: 0 0 25%;
            display: flex;
            justify-content: center;
            align-items: center;
            text-align: center;
            background-color: #e0e0e0;
            border-radius: 10px;
            transition: background-color 0.3s, transform 0.3s;
            cursor: pointer;
            padding: 10px;
        }
        .category:hover {
            background-color: #d4edda;
            transform: scale(1.05);
        }
        .category h2 {
            margin: 0;
            font-size: 1.5em;
            color: #333;
        }
    </style>
</head>
<body>
    <div class="container">
        <div class="category" onclick="navigate('furniture.html')">
            <h2>Furniture</h2>
        </div>
        <div class="category" onclick="navigate('appliances.html')">
            <h2>Appliances</h2>
        </div>
        <div class="category" onclick="navigate('electronics.html')">
            <h2>Cameras</h2>
        </div>
    </div>

    <script>
        function navigate(page) {
            window.location.href = page;
        }
    </script>
</body>
</html>
```

5.1.2 Backend code

Adminlogin.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Login Page</title>
    <link rel="stylesheet" href="adminlogin.css">
</head>
<body>
    <div class="nav">
        <div class="image">
            <img src="https://is1-
ssl.mzstatic.com/image/thumb/Purple126/v4/f7/91/ef/f791ef78-2f7f-8227-e0da-
fb48bfbd117f/AppIcon-1x_U007emarketing-0-7-0-85-220.jpeg/1200x630wa.png"
alt="error">
        </div>
        <div class="brand">
            Rent:Ease
        </div>
    </div>
    <!-- form creation -->
    <main>
        <form>
            <p>Admin Login</p>
            <label for="email">Email :</label>
            <input type="email" name="email" id="email">

            <label for="Username">Username :</label>
            <input type="text" name="Username" id="Username">

            <label for="password">Password :</label>
            <input type="password" name="Password" id="Password">

            <button type="submit" id="button">Login</button>
        </form>
    </main>
</body>
<script src="adminlogin2.js"></script>
</html>
```

appliances.html

```
<!DOCTYPE html>
<html lang="en">
<head>
```

```
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Appliances</title>
<style>
    body {
        font-family: Arial, sans-serif;
        margin: 0;
        padding: 0;
        display: flex;
        justify-content: center;
        align-items: center;
        height: 100vh;
        background-color: #f7f7f7;
    }
    .container {
        display: flex;
        width: 90%;
        height: 80%;
        box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
        border-radius: 10px;
        overflow: hidden;
        background-color: #fff;
    }
    .sidebar {
        flex: 0 0 30%;
        display: flex;
        flex-direction: column;
        align-items: center;
        padding: 20px;
        background-color: #f0f0f0;
        border-right: 1px solid #ddd;
    }
    .sidebar button {
        width: 80%;
        padding: 10px;
        margin-bottom: 20px;
        border: none;
        border-radius: 5px;
        background-color: #007bff;
        color: white;
        cursor: pointer;
        font-size: 1em;
    }
    .sidebar button:hover {
        background-color: #0056b3;
    }
    .content {
        flex: 1;
        display: grid;
```

```
    grid-template-columns: repeat(2, 1fr);
    gap: 20px;
    padding: 20px;
    width: 70%;
    justify-content: center;
    align-content: start;
}
.item {
    display: flex;
    flex-direction: column;
    align-items: center;
    padding: 10px;
    box-shadow: 0 0 5px rgba(0, 0, 0, 0.1);
    border-radius: 10px;
    background-color: #f9f9f9;
    width: 90%;
    box-sizing: border-box;
}
.item img {
    width: 80px;
    height: 80px;
    margin-bottom: 10px;
    object-fit: cover;
    border-radius: 5px;
}
.item p {
    margin: 0 0 10px;
    text-align: center;
    font-size: 0.9em;
}
.actions {
    display: flex;
    gap: 10px;
    width: 100%;
    justify-content: center;
}
.actions button {
    padding: 5px 10px;
    border: none;
    border-radius: 5px;
    background-color: #28a745;
    color: white;
    cursor: pointer;
}
.actions button.delete {
    background-color: #dc3545;
}
.actions button:hover {
    opacity: 0.8;
```

```
}
    </style>
</head>
<body>
    <div class="container">
        <div class="sidebar">
            <button onclick="navigateToAdd()">Add</button>
        </div>
        <div class="content" id="itemList">
            <p>Oops!! Empty</p>
        </div>
    </div>
    <script>
        let items = [];
        function navigateToAdd() {
            window.location.href = 'add1.html?category=appliances';
        }
        function editItem() {
            const index = prompt("Enter item index to edit (starting from
1):");
            if (index && index > 0 && index <= items.length) {
                const item = items[index - 1];
                const newId = prompt("Enter new item ID:", item.id);
                const newImage = prompt("Enter new item image URL:",
item.image);
                const newName = prompt("Enter new item name:", item.name);
                const newPrice = prompt("Enter new item price:", item.price);
                if (newId && newImage && newName && newPrice) {
                    items[index - 1] = { id: newId, image: newImage, name:
newName, price: newPrice };
                    localStorage.setItem('appliances', JSON.stringify(items));
                    displayItems();
                } else {
                    alert("All fields must be filled out to edit the item.");
                }
            } else {
                alert("Invalid item index.");
            }
        }
        function deleteItem() {
            const index = prompt("Enter item index to delete (starting from
1):");
            if (index && index > 0 && index <= items.length) {
                items.splice(index - 1, 1);
                localStorage.setItem('appliances', JSON.stringify(items));
                displayItems();
            } else {
                alert("Invalid item index.");
            }
```

```
        }
        function displayItems() {
            const itemList = document.getElementById('itemList');
            items = JSON.parse(localStorage.getItem('appliances')) || [];
            if (items.length === 0) {
                itemList.innerHTML = '<p>Oops!! Empty</p>';
            } else {
                itemList.innerHTML = items.map((item, index) =>
                    `<div class="item">
                        <img src="${item.image}" alt="${item.name}">
                        <p>ID: ${item.id}<br>Name: ${item.name}<br>Price:
$${item.price}</p>
                        <div class="actions">
                            <button
onclick="initiateEdit(${index})">Edit</button>
                            <button class="delete"
onclick="initiateDelete(${index})">Delete</button>
                        </div>
                    </div>`).join('');
            }
        }
        function initiateEdit(index) {
            const item = items[index];
            const newId = prompt("Enter new item ID:", item.id);
            const newImage = prompt("Enter new item image URL:", item.image);
            const newName = prompt("Enter new item name:", item.name);
            const newPrice = prompt("Enter new item price:", item.price);
            if (newId && newImage && newName && newPrice) {
                items[index] = { id: newId, image: newImage, name: newName,
price: newPrice };
                localStorage.setItem('appliances', JSON.stringify(items));
                displayItems();
            } else {
                alert("All fields must be filled out to edit the item.");
            }
        }
        function initiateDelete(index) {
            items.splice(index, 1);
            localStorage.setItem('appliances', JSON.stringify(items));
            displayItems();
        }
        window.onload = function() {
            displayItems();
        };
    </script>
</body>
</html>
```

Add1.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Add Item</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            display: flex;
            justify-content: center;
            align-items: center;
            height: 100vh;
            background-color: #f7f7f7;
            margin: 0;
        }
        .form-container {
            background-color: #fff;
            padding: 20px;
            border-radius: 10px;
            box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
            width: 300px;
        }
        .form-container h2 {
            margin-bottom: 20px;
            color: #333;
        }
        .form-container input {
            width: 100%;
            padding: 10px;
            margin-bottom: 10px;
            border: 1px solid #ddd;
            border-radius: 5px;
        }
        .form-container button {
            width: 100%;
            padding: 10px;
            border: none;
            border-radius: 5px;
            background-color: #5cb85c;
            color: white;
            cursor: pointer;
        }
        .form-container button:hover {
            background-color: #4cae4c;
        }
    </style>
```

```
</head>
<body>
    <div class="form-container">
        <h2>Add New Item</h2>
        <form id="addItemForm">
            <input type="text" id="itemId" placeholder="Enter ID" required>
            <input type="text" id="itemImage" placeholder="Enter Image URL"
required>
            <input type="text" id="itemName" placeholder="Enter Name"
required>
            <input type="number" id="itemPrice" placeholder="Enter Price"
required>
            <button type="submit">Add Product</button>
        </form>
    </div>
    <script>
        document.getElementById('addItemForm').addEventListener('submit',
function(event) {
            event.preventDefault();
            const id = document.getElementById('itemId').value;
            const image = document.getElementById('itemImage').value;
            const name = document.getElementById('itemName').value;
            const price = document.getElementById('itemPrice').value;
            if (!id || !image || !name || !price) {
                alert("Please fill in all fields.");
                return;
            }
            const urlParams = new URLSearchParams(window.location.search);
            const category = urlParams.get('category');
            if (!category) {
                alert("Cannot determine the category. Please return to the
main page.");
                return;
            }
            const items = JSON.parse(localStorage.getItem(category)) || [];
            items.push({ id, image, name, price });
            localStorage.setItem(category, JSON.stringify(items));
            window.location.href = category + '.html';
        });
    </script>
</body>
</html>
```

Payment.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
```

```
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Document</title>
    <link rel="shortcut icon" href="o3qd9acvznwgmslknql7.avif" type="image/x-
icon">
    <style>
      * {
        margin: 0%;
        padding: 0%;
      }
      form > input {
        margin-top: 20px;
        height: 30px;
        width: 400px;
        border-radius: 4px;
      }
      .maincon {
        box-shadow: rgba(0, 0, 0, 0.24) 0px 3px 8px;
        display: flex;
        align-items: center;
        justify-content: space-evenly;
        width: 80%;
        margin: auto;
        padding-bottom: 7%;
        border-radius: 5px;
        /* background-color: aqua; */
        height: 400px;
      }
      h3 {
        text-align: center;
        color: #dc3226;
        font-size: 40px;
        margin-bottom: 1rem;
      }
      .sub {
        align-items: center;
        border-radius: 3px;
        margin-top: 18%;
        background-color: red;
        color: white;
        font-size: 20px;
        border: 1px solid #dc3226;
        height: 2rem;
        font-weight: 600;
      }
      .sub:hover {
        background-color: white;
        cursor: pointer;
        color: #dc3226;
```

```
    }
    input {
      padding-left: 2%;
    }
    #rhs {
      margin-top: 2%;
    }
    #total {
      margin-top: 3%;
    }
    #navbar {
      display: none;
    }
    .add {
      font-size: large;
      align-items: center;
      text-align: center;
      padding-left: 18%;
      text-decoration: underline;
      font-weight: 700;
    }
    .pay {
      padding-left: 20%;
      text-decoration: underline;
      font-size: larger;
      font-weight: 700;
    }
    .rhs {
      width: 40%;
      margin-top: 1%;
    }
    .buttonn {
      background-color: red;
     padding: 10px;
     border-radius: 16px;
     text-decoration: none;
     color:white
    }
  </style>
  <link rel="stylesheet" href="/admin.css" />
</head>
<body>
  <div id="navbar">
    <div>
      <img
        src="https://www.rentomojo.com/public/images/Home/new/ham__icon.svg"
        id="dropdown_menu_open"
        alt=""
      /><img
```

```
          src="https://www.rentomojo.com/public/images/Home/new/logo.svg"
          alt=""
        />
      </div>
      <div onclick="goToHome()"><img src="rentomarket (3).png" alt="" /></div>
      <div>
        <div id="location">
          <!-- <p id="change_location"><b>WELCOME TO THE ADMIN PAGE</b></p>-->
        </div>
      </div>
    </div>
    <h2 id="total"></h2>
    <h3 class="mainhead">Enter Your Address And Payment Details</h3>
    <div class="maincon">
      <div class="lhs">
        <form action="">
          <label for="" class="add">Address Details</label>
          <input type="text" placeholder="Name" class="s1" /><br />

          <input type="number" placeholder="Phone No." class="s2" /><br
/><input
            type="text"
            placeholder="Address(Area and Street)"
            class="s3"
          /><br /><input
            type="text"
            placeholder="City/District/Town"
            class="s4"
          /><br />
          <input type="number" placeholder="Pin-Code" class="s5" /><br />
          <input
            type="number"
            placeholder="Alternate No. (Optional)"
            class="s6"
          /><br />
          <!-- //<input type="submit"> -->
        </form>
      </div>
      <div class="rhs">
        <form action="" id="rhsform">
          <br /><br /><br />
          <label for="" class="pay">Payment Details</label>
          <input
            type="text"
            placeholder="Name of Account Holder"
            class="s7"
          /><br />
          <input type="number" placeholder="Card No." class="s8" /><br
/><input
```

```
            type="number"
            placeholder="Cvv"
            minlength="3"
            maxlength="3"
            min="111"
            max="999"
            class="s9"
          /><br />
          <input
            type="number"
            min="1"
            max="12"
            placeholder="Exp Month"
            class="s10"
          />
          <input type="number" placeholder="Exp Year" min="1995" class="s11"
/>
          <div style="margin: 10%;"><a  class="buttonn"
href="thankyou.html">Submit</a></div>
        </form>
      </div>
    </div>
  </body>
</html>
<script>
  data_item = JSON.parse(localStorage.getItem("cartpage"));
  console.log(data_item);
  let sum = 0;
  data_item.map((item) => {
    console.log(item.price);
    sum = sum + Number(item.price);
  });
  document.querySelector("#total").innerText =
    "Your Cart Amount " + (sum || 0) + " " + "₹";
  document.querySelector("#rhsform").addEventListener("submit", (event) => {
    event.preventDefault();
    let data1 = document.querySelector(".s1");
    let data2 = document.querySelector(".s2");
    let data3 = document.querySelector(".s3");
    let data4 = document.querySelector(".s4");
    let data5 = document.querySelector(".s5");
    let data6 = document.querySelector(".s6");
    let data7 = document.querySelector(".s7");
    let data8 = document.querySelector(".s8");
    let data9 = document.querySelector(".s9");
    let data10 = document.querySelector(".s10");
    let data11 = document.querySelector(".s11");
    data9.value.length = 3;
    if (data1.value === "") {
```

```
      alert("Please enter the Name");
    } else if (data2.value == "") {
      alert("Please enter the Phone.No");
    } else if (data3.value == "") {
      alert("Please enter the Address");
    } else if (data4.value == "") {
      alert("Please enter the City");
    } else if (data5.value == "") {
      alert("Please enter the Pin-Code");
    } else if (data8.value == "") {
      alert("Please enter the Account Holder Name");
    } else if (data8.value == "") {
      alert("Please enter the Card No.");
    } else if (data8.value.length < 11 || data8.innerText.value > 11) {
      alert("Please check the Card No.");
    } else if (data9.value == "") {
      alert("Please enter the Phone.No");
    } else if (data9.value.length < 3 || data9.value.length > 3) {
      alert("Please enter correct Cvv");
    } else if (data10.value < 1 && data10.value > 12) {
      alert("Please Check Exp Month");
    } else if (data11.value == "") {
      alert("Please Enter Exp year");
    } else if (data11.value < 2022) {
      alert("Your Card is Expired");
    } else {
      console.log("S");
      window.location.href = "cvv.html";
    }
    console.log("s");
  });
  function submitted() {
    // alert("Payment Successfull");
    window.location.href="thankyou.html";
  }
</script>
```

5.2 Experimental Results

5.2.1 Login/Signup page

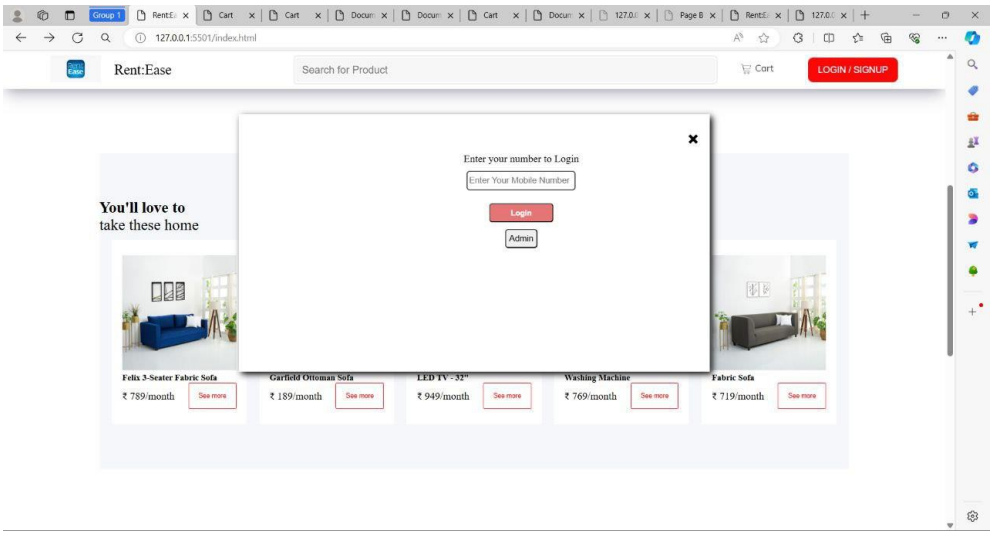

Fig 5.1 Login/Signup Page

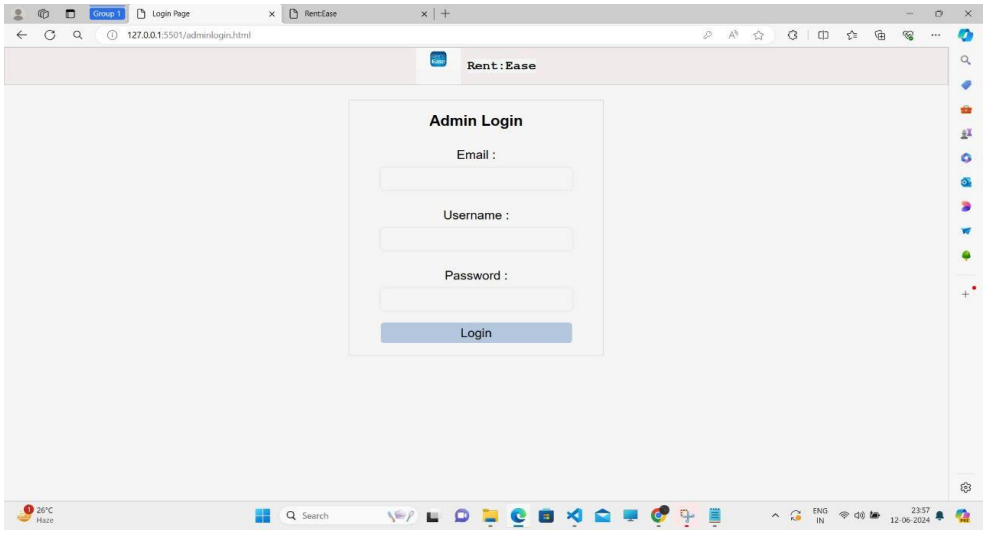

Fig 5.2 Admin Login Page

- Users can enter their mobile number to receive an OTP for login verification.
- The user login form accepts the admin's email, username, and password.
- Upon successful login, the system authenticates the admin and grants access to the admin dashboard.

5.2.2 Incorrect Credentials

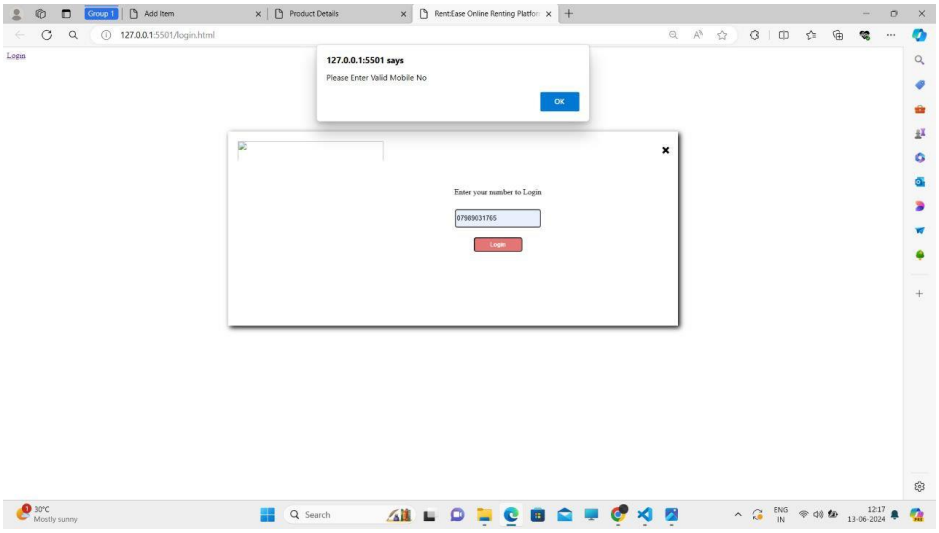

Fig 5.3 Invalid Mobile Number

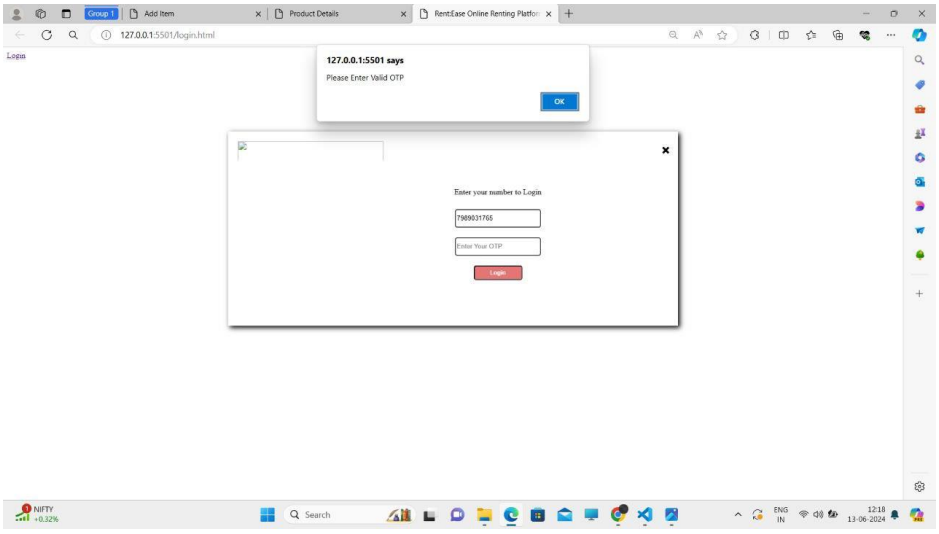

Fig 5.4 Invalid OTP

- The accompanying JavaScript handles the display of the login form, validates the mobile number, shows the OTP field upon entering a valid mobile number, and verifies the OTP to complete the login process, displaying a welcome message upon successful login.

5.2.3 Homepage/Index Page

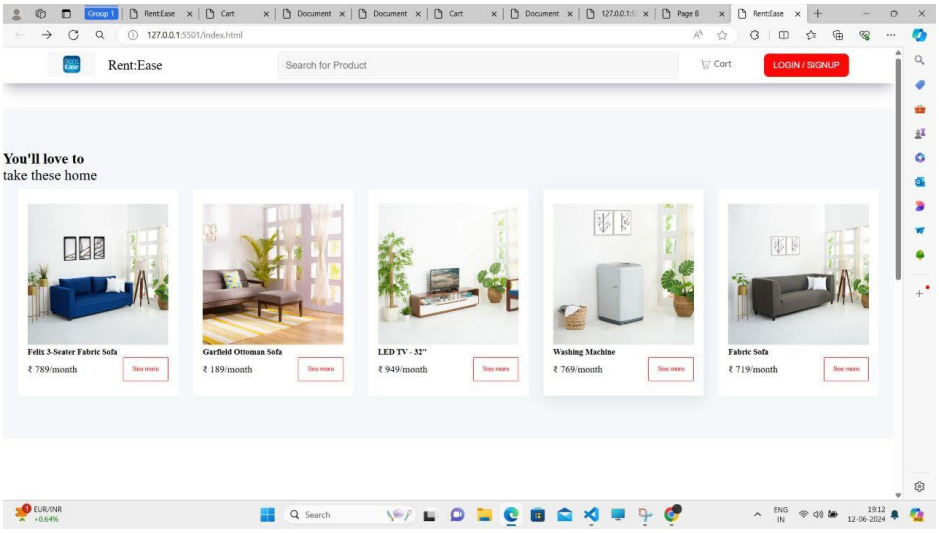

Fig 5.5 Homepage

- This features a navigation bar, a pop-up login form, product listings with images and pricing, and a section highlighting additional services such as free maintenance and easy returns. The JavaScript functions handle the login process and navigation to product and cart pages.

5.2.4 Furniture Index Page

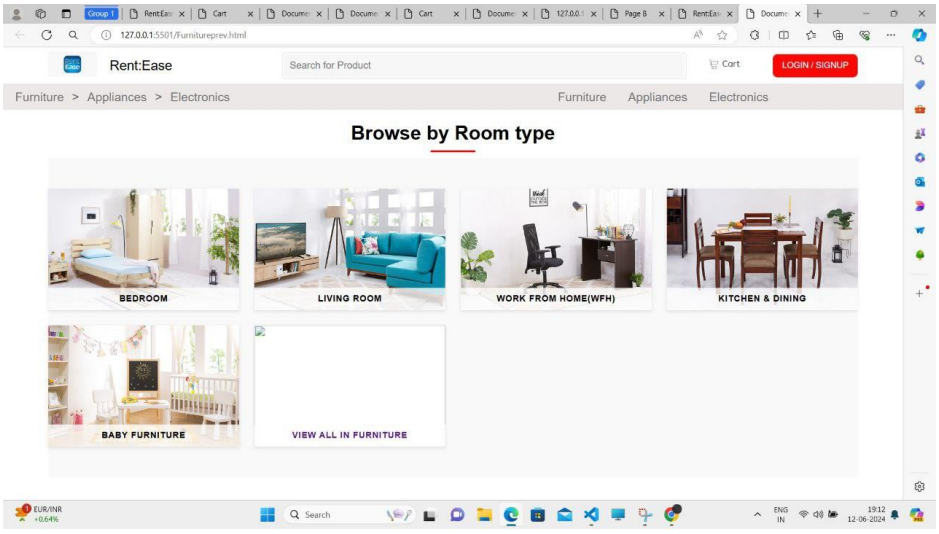

Fig 5.6 Furniture index page

- A webpage with a navbar, a header featuring navigation links, and a main section displaying categories of furniture for rent.
- It includes images and links for browsing different room types, such as the bedroom,

living room, and kitchen, with a final card that redirects users to view all furniture products.

5.2.5 Appliances Index Page

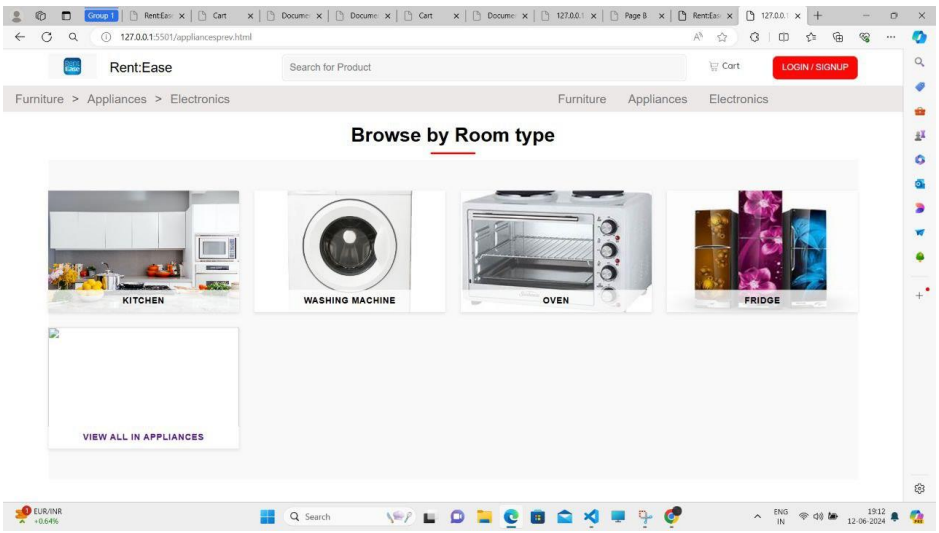

Fig 5.7 Appliances index page

- A website with a navbar, a header providing navigation links, and a primary segment for surfing distinct sorts of cameras for rent.
- It consists of photo playing cards with classes like Simple, Cinematic, DSLR, and Action Cameras, together with a hyperlink to view all digital digicam products.

5.2.6 Camera Index Page

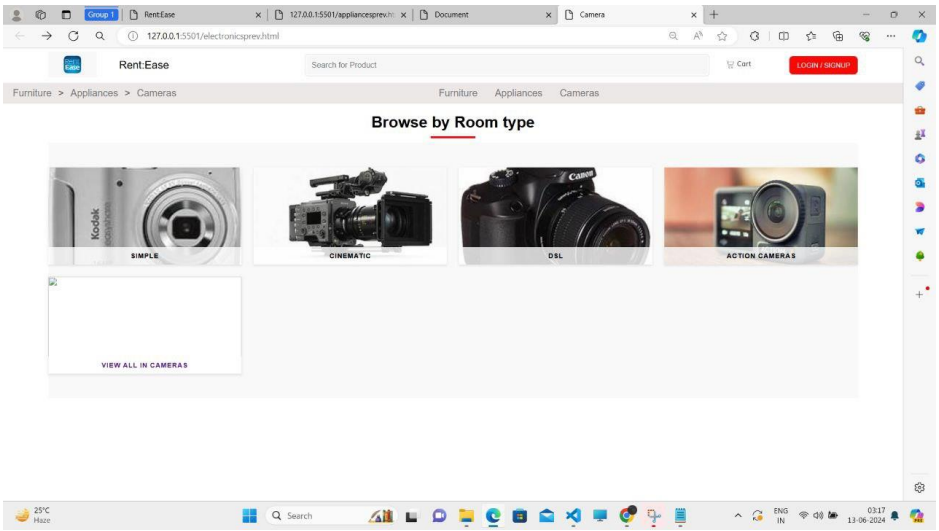

Fig 5.8 Camera index page

- Webpage that consists of a navbar, a header with navigation hyperlinks, and a primary phase showcasing exceptional classes of home equipment to be had for rent.
- Each category, together with Kitchen, Washing Machine, Oven, and Fridge, is displayed with a photo and description, and there`s a choice to view all home equipment.
- The code additionally consists of navigation hyperlinks to different sections like Furniture and Cameras.

5.2.7 Category Selection Page

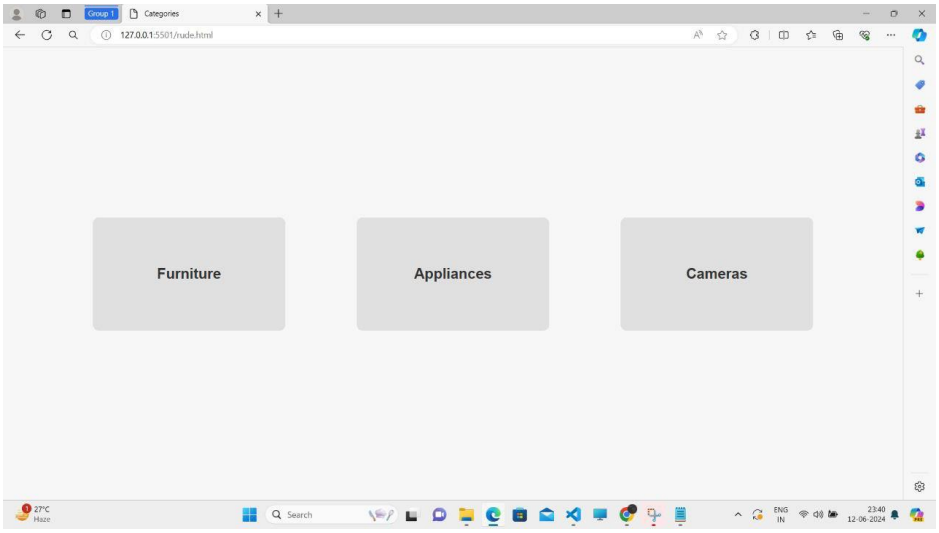

Fig 5.9 Category-wise selection

- A smooth and responsive interface for deciding on classes together with Furniture, Appliances, and Cameras. When a class is clicked, it navigates to the corresponding HTML page.

5.2.8 Category Database

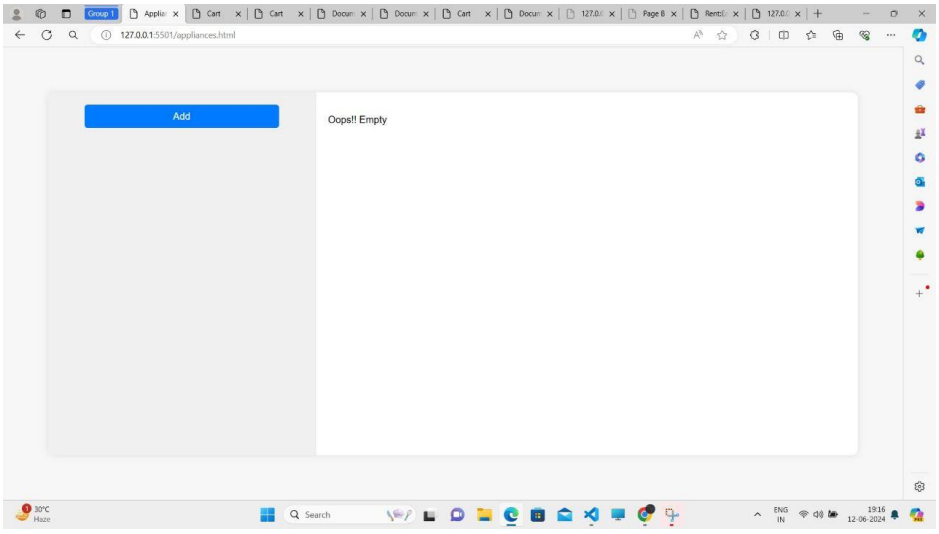

Fig 5.10 Category Database

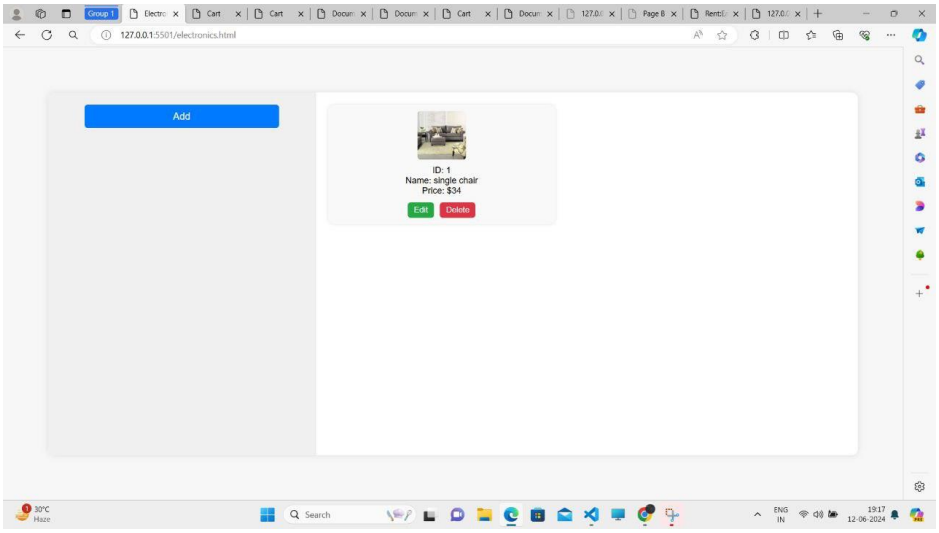

Fig 5.11 Category Database with One item

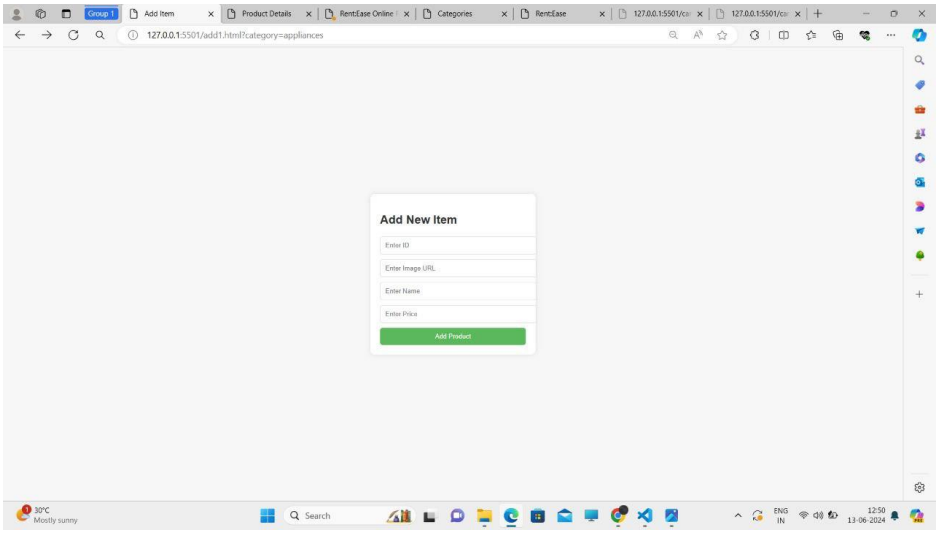

Fig 5.12 Entering details to add a New Item

- It lets in customers to add a new object to a selected class via way of means of filling out a form. When the form is submitted, the new object is stored withinside the browser`s nearby garage below the suitable class.
- If the form fields aren't well crammed out or if the category cannot be determined, suitable indicators are shown.

5.2.9 Items parsing Through Database

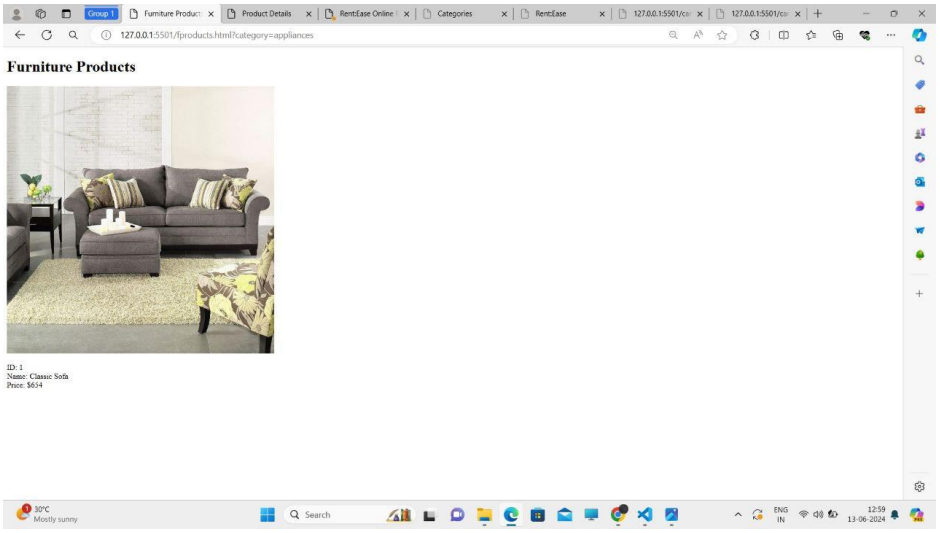

Fig 5.13 Furniture Products parsing Through Database

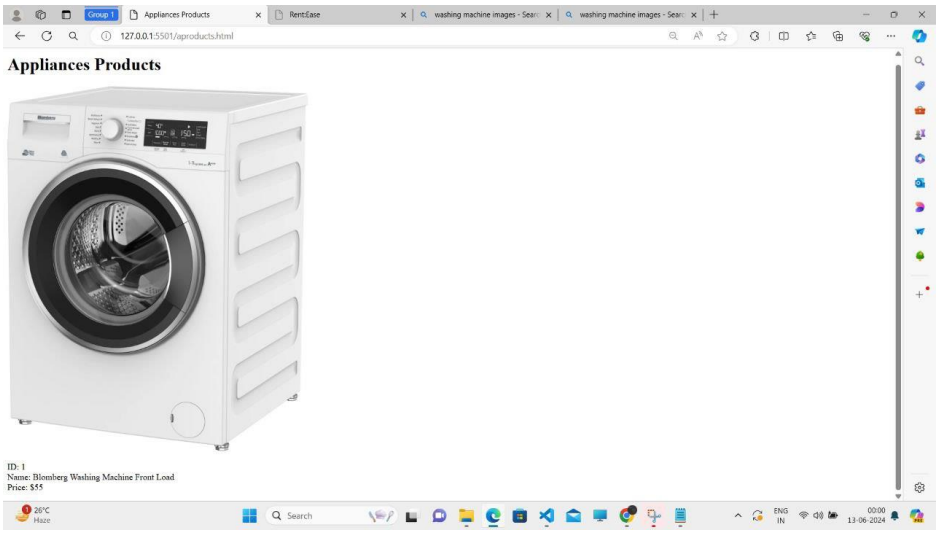

Fig 5.14 Appliances Products parsing Through Database

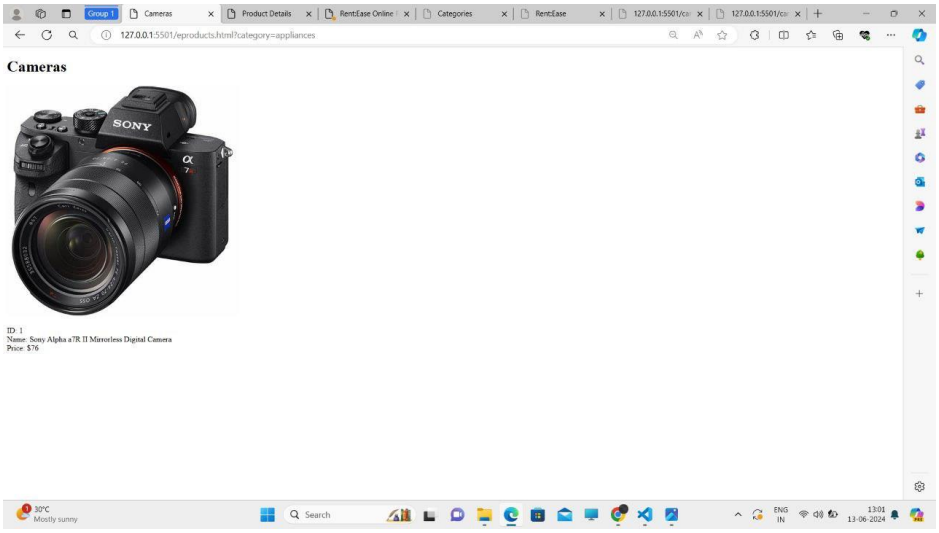

Fig 5.15 Cameras parsing Through Database

- It dynamically shows products labeled as Furniture, Electronics, and Appliances respectively, fetching records from local storage and supplying them in a grid format with clickable images to designated product pages.

5.2.10 Cart Page

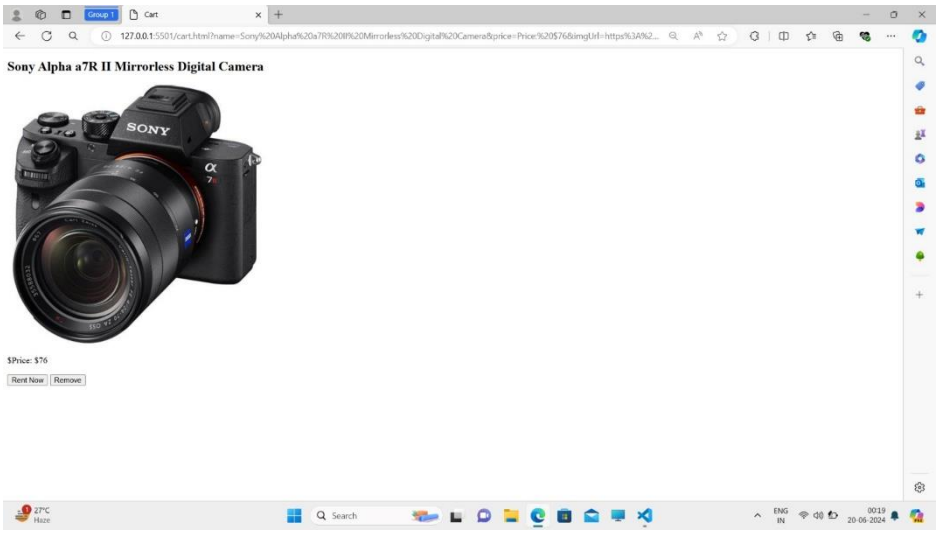

Fig 5.16 Items added to Cart

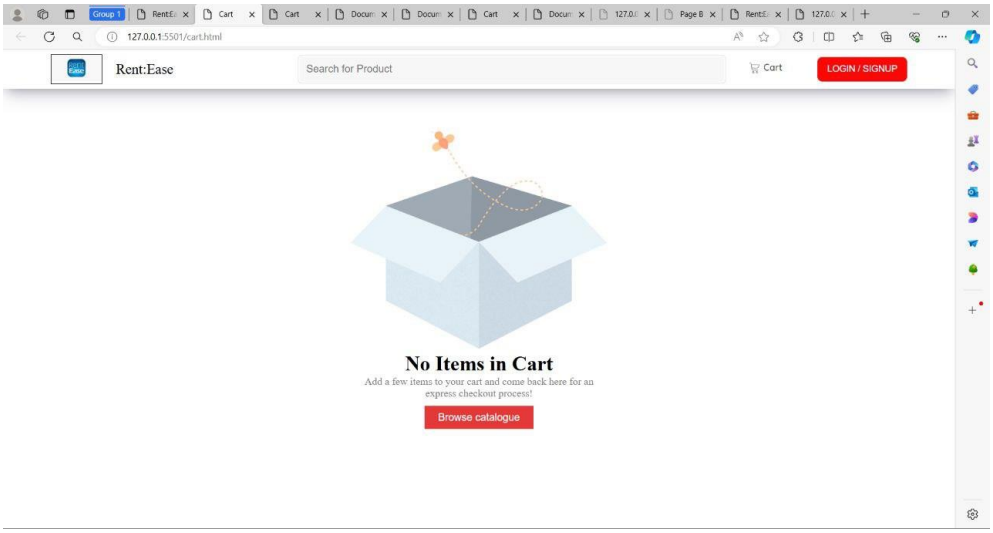

Fig 5.17 Empty Cart page

- Creating a purchasing cart web page showing products with images, descriptions, prices, and "Rent Now" buttons styled within a flexible container.

5.2.11 Transaction Page

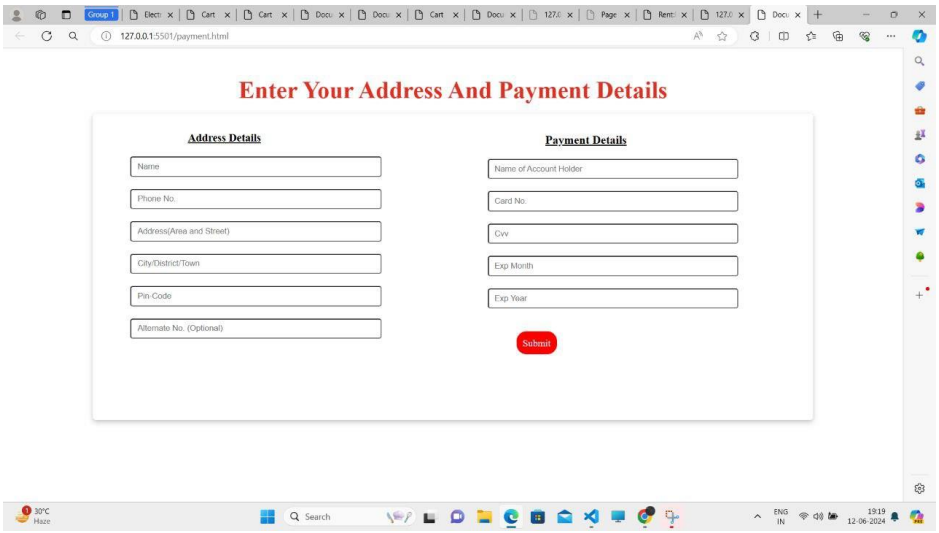

Fig 5.18 Transaction page

- Creates a checkout web page in which customers can input their address and payment details.

5.2.12 Order Confirmation Page

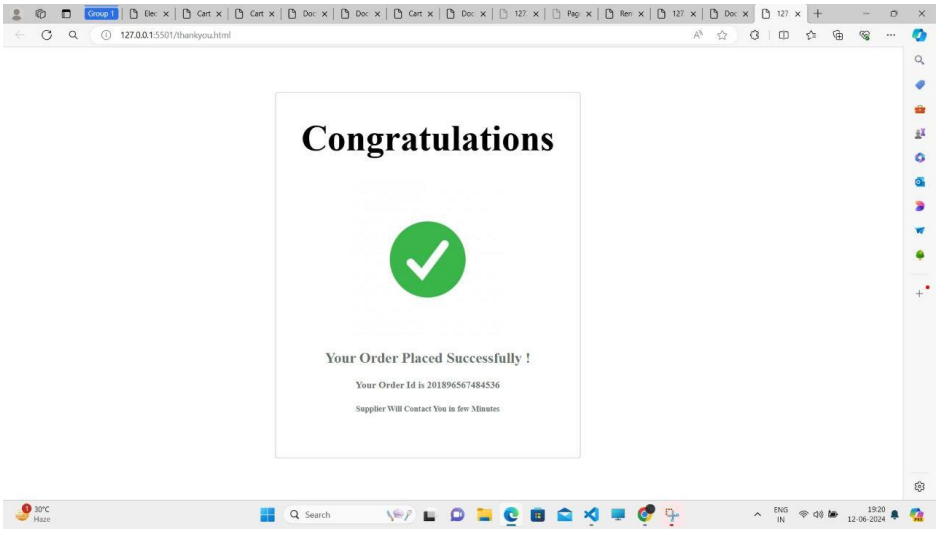

Fig 5.19 Order confirmation page

- The affirmation web page showing a congratulatory message that “Your Order Placed Successfully!”.

5.3 Test Cases

Sl. No	Input	Output
1.	If we click on Button Signup.	It opens the Admin Login Page.
2.	Onclick on admin details login.	Open the add edit delete options of product details.
3.	Onclick on Login.	Open customer login details.
4.	Onclick on customer admin details.	Redirects to index Page.
5.	Selecting Categories.	Opening separate category databases.
6.	Selecting products.	Opening of Product details.
7.	Click on cart.	Opens the product details in cart page and allows us to proceed further or kill the product there itself.

8.	Onclick on rent now.	Proceed for transaction details.
9.	Onclick on confirm.	Order has been successfully placed is displayed.

CHAPTER 06

CONCLUSION AND FUTURE SCOPE

Conclusion

To sum up, the RentEase Online Rental System has greatly changed how people rent things out by delivering on its core values that are enhancement of user experience, better efficiency in operations and presentation to data for informed decisions. RentEase has consolidated the rental process through an easy-to-use online platform which reduces administrative workloads while ensuring seamless transactions between tenants and landlords. According to users' comments, it is clear that this program has a friendly interface and excellent features like... Alternatively, advanced analytics have delivered useful market insights thereby enabling strategic direction and future-oriented moves. By carrying out successive improvements that were geared towards addressing issues including technical optimizations as well as user acceptance glitches, latest AI integration advancements alongside mobile accessibilities will boost Rent-Ease's capacities even further. In order to succeed in expanding their market presence on the whole or becoming one of the leaders in online rental management, they need to create strategic partnerships and continue innovating at all times. As a matter of fact, if RentEase can adapt to changing industry demands and surpass user expectations over time; it will forever maintain its impact in years ahead.

Future Scope

The destiny of on-line leasing offerings shines brightly with extra purchasers searching out less expensive and sustainable intake. This sample covers special regions consisting of fashion, electronics, furnishings and gear in which renting gives unbeatable flexibility and get right of entry to without the attendant obligations of ownership. Supporting this transition are upgrades in generation that consist of logistics and virtual systems that facilitate seamless transactions and customized experiences. Data evaluation improves carrier performance and stock control whilst the emergence of sharing economies highlights environmental benefits of aid performance. As those offerings expand subscription fashions and logo associations, they may be critical to figuring out destiny intake styles thereby fostering monetary efficacy in addition to environmental sustainability.

CHAPTER 07
REFERENCES

1. ONLINE RENTAL HOUSING Sahreen Afzal, Toiba Rouf, Sumaiya Qadir, Sahila Shah, Volume 8, Issue 11, JETIR November 2021.
2. HOME APPLIANCES FOR RENT M.NIREESHA, P.SRINIVASA REDDY, Volume 7, Issue 5, May 2020.

3. LeKeDe: Online Rental System Amika Mehta, Vedant Patil, Apurva Shinde Vol. 8 Issue 10, October-2019 •
4. Tiffin Services Application and Live Tracking, RiteshNimje, Aparna Gurjar Volume, 09 Issue, 7 March– 2018

5. R. D. Sari, "Building Application System Car Rental Reservation and Payment Online Web-Based (Case Study in The Rental Daras Corporation)," unpublished. Undergraduate Thesis. Bandung: Unikom, 2011.

6. Car Rental System, Amey Thakur, Department of Computer Engineering, University of Mumbai, Volume 9, Issue VII July 2021