AY 2024-25

III Semester

CSL306

A PROJECT REPORT
ON

Home Ease:A Comprehensive Home Rental Platform

Bachelor of Technology
in
School of Computing

By
Sajal Kumar Baranal(23151)

SCHOOL OF COMPUTING

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY UNA
HIMACHAL PRADESH

DECEMBER 2024

BONAFIDE CERTIFICATE

This is to certify that the project titled “Home Ease: A Comprehensive Home Rentals Platform” is a bonafide record of the work done by

SAJAL KUMAR BARANWAL (23151)

in partial fulfilment of the requirements for the award of the degree of Bachelor of Technology in COMPUTER SCIENCE AND ENGINEERING of the INDIAN INSTITUTE OF INFORMATION TECHNOLOGY UNA, HIMACHAL PRADESH, during the year 2024 - 2025.

under the guidance of

Dr. NIKUNJ GOYAL

Project viva-voce held on: _____

Internal Examiner External Examiner

ORIGINALITY / NO PLAGIARISM DECLARATION

I certify that this project report is our original report and no part of it is copied from any published reports, papers, books, articles, etc. We certify that all the contents in this report are based on our personal findings and research and we have cited all the relevant sources which have been required in the preparation of this project report, whether they be books, articles, reports, lecture notes, and any other kind of document. We also certify that this report has not previously been submitted partially or as whole for the award of degree in any other university in India and/or abroad.

I hereby declare that I am fully aware of what constitutes plagiarism and understand that if it is found at a later stage to contain any instance of plagiarism, my degree may be cancelled.

SAJAL KUMAR BARANWAL (23151)

ABSTRACT

HomeEase is a modern web platform designed to revolutionize the rental process by offering an efficient, secure, and user-friendly experience for property hosts and tenants. The platform enables seamless property listings, secure booking management, and an intuitive wishlist feature that allows users to effortlessly add or remove desired properties.

By focusing on clear and transparent communication between hosts and tenants, HomeEase fosters trust and ensures a smooth rental experience. With its robust infrastructure and innovative features, the platform addresses the growing demand for reliable and accessible online rental services. HomeEase aims to simplify the complexities of property rentals, creating a safe and efficient environment that caters to the needs of its users.

***Keywords*:** Online Rentals , Property Listings , web

ACKNOWLEDGEMENT

I would like to thank the following people for their support and guidance without whom the completion of this project in fruition would not be possible.

I would like to express our sincere gratitude and heartfelt thanks to **Dr. Nikunj Goyal** for their unflinching support and guidance, valuable suggestions and expert advice. Their words of wisdom and expertise in subject matter were of immense help throughout the duration of this project.

I also take the opportunity to thank our Director and all the faculty of School of Computing and Electronics, IIIT Una for helping me by providing necessary knowledge base and resources.

I would also like to thank my parents and friends for their constant support.

SAJAL KUMAR BARANWAL (23151)

TABLE OF CONTENTS

Title **Page No.**

ABSTRACT……iii

ACKNOWLEDGEMENT……..……iv

TABLE OF CONTENTS……v

LIST OF ACRONYMS……..vi

LIST OF TABLES……vii

LIST OF FIGURES……..viii

1 Introduction……...1

1.1 Background……...……..1

1.1.1 Current Challenges in the Rental Industry.……..…1

1.1.2 Growing Need for Online Rental Platforms…….2

1.1.3 Gaps Addressed by HomeEase……3

2 Review of Literature ……4

2.1 Existing Rental Platforms……...4

2.2 Technological Framework ……5

2.3 Trends in Online Rental Services…….5

3 Motivation and Problem Definition……6

3.1 Problem Definition……..6

3.1.1 Objective of Problem Definition:.....……...…6

3.2 Motivation……...……6

4 Proposed Methodology ……...8

4.1 System Architecture…….…..8

4.2 Detailed WebPages…….…9

5 Conclusion and Future Work……11

5.1 Project Summary…….……...11

5.2 Concluding Remarks……..…11

5.3 Future Enhancement……11

References……..…..12

Appendices…….3

LIST OF ACRONYMS

UI User Interface

UX User Experience

CRUD Create, Read, Update, Delete

API Application Programming Interface

MERN MongoDB, Express.js, React.js, Node.js

SSL Secure Sockets Layer

DB Database

JSON JavaScript Object Notation

SEO Search Engine Optimization

HTTPS HyperText Transfer Protocol Secure

CMS Content Management System

NoSQL Not Only SQL

JWT JSON Web Token

UI/UX User Interface/User Experience

LIST OF TABLES

4.1 Project technologies and Tools……9

LIST OF FIGURES

2.1 Market Analyzation……..……..2

2.2 Block diagram of website working…..…….8

Chapter 1

Introduction

1.1 Background

1.1.1 Current Challenges in the Rental Industry

The rental industry faces several challenges that hinder efficient transactions and create friction for both property hosts and tenants. These issues include:

1. **Inefficient Property Discovery**
   Tenants often struggle to find properties that match their preferences due to disorganized listings, lack of filtering options, and insufficient property details. Hosts face difficulties in attracting suitable tenants because of poor visibility and limited reach.
2. **Lack of Transparency**
   Trust is a significant concern in the rental process. Tenants often encounter incomplete or misleading property descriptions, while hosts face uncertainty about the reliability of prospective tenants.
3. **Complex Communication**
   Effective communication between hosts and tenants is often fragmented, leading to misunderstandings, delays, and dissatisfaction. Traditional communication channels, such as emails or phone calls, lack the efficiency and immediacy needed in a fast-paced rental market.
4. **Security Concerns**
   Both hosts and tenants are wary of security issues, such as fraudulent transactions, identity theft, or unauthorized bookings. The lack of a secure, centralized system exacerbates these risks.

5. **Manual and Time-Consuming Processes**

   Traditional rental methods involve tedious paperwork, manual agreement processes, and prolonged negotiations, making the entire experience time-intensive and inefficient

1.1.2 Growing Need for Online Rental Platforms

The shift towards digital transformation in various industries has also extended to the rental market. Online rental platforms have emerged as a solution to these challenges, offering:

- **Convenience**: Tenants can browse and book properties from anywhere, while hosts can manage listings effortlessly.
- **Streamlined Processes**: Automation reduces the time required for tasks like property discovery, communication, and payment handling.
- **Enhanced Trust**: Verified user profiles, detailed property descriptions, and secure payment gateways build confidence among users.
- **Broader Reach**: Online platforms enable hosts to showcase their properties to a wider audience, increasing their chances of finding tenants.

Fig 1.1 Market Analyzation

1.1.3 Gaps Addressed by HomeEase

Despite the advancements of existing platforms, several gaps persist that **HomeEase** aims to address:

1. **Enhanced User Experience**: HomeEase provides an intuitive, user-friendly interface that simplifies the property search, booking, and management processes.
2. **Dynamic Wishlist Management**: The ability to easily add or remove properties from a wishlist ensures a personalized and organized experience for tenants.
3. **Secure Booking System**: With robust encryption and payment gateways, HomeEase ensures safe transactions, protecting users from fraud.
4. **Efficient Communication Tools**: HomeEase integrates streamlined communication features, fostering better understanding and faster resolutions between hosts and tenants.
5. **Transparent Listings**: The platform emphasizes verified listings with comprehensive property details, ensuring trust and reliability for users.

By addressing these challenges, HomeEase is well-positioned to transform the rental landscape, providing a seamless, secure, and efficient experience for both hosts and tenants.

Chapter 2
Review of Literature

The literature review provides an analysis of existing research, platforms, and technologies relevant to **HomeEase**. It explores the current state of online rental services, their challenges, and how HomeEase fits into the evolving landscape of digital property management.

2.1 Existing Rental Platforms

Several online rental platforms, such as Airbnb, Zillow, and Booking.com, have transformed the way users interact with property rentals. These platforms offer features like property listings, online payments, and communication tools. However, they are not without limitations:

1. Generic User Experience: Many platforms focus on a one-size-fits-all approach, which limits personalization and flexibility.
2. High Service Fees: Hosts and tenants often face steep fees, reducing the cost-effectiveness of these platforms.
3. Limited Local Reach: Platforms tend to cater more to global audiences, often overlooking local rental needs and preferences.

HomeEase aims to overcome these limitations by providing a cost-effective, user-friendly solution tailored to both global and local markets.

2.2 Technological Framework

The development of web-based platforms like HomeEase relies heavily on modern technologies. Key components include:

- **Frontend Technologies**: React is widely used for creating intuitive and dynamic user interfaces. Its reusable components make it ideal for platforms like HomeEase.
- **Backend Technologies**: Node.js, coupled with Express.js, ensures scalable and efficient server-side operations.
- **Database Systems**: MongoDB, a NoSQL database, allows for flexible and scalable data storage, making it suitable for managing property details, user information, and transactions.

These technologies provide a robust foundation for delivering a seamless user experience.

2.3 Trends in Online Rental Services

The rental industry is experiencing significant growth, driven by increasing urbanization and a preference for short-term rentals. Research highlights key trends:

- **Shift to Digital Platforms**: More users are moving from offline methods to online platforms for convenience and efficiency.
- **Demand for Transparency**: Users prioritize platforms that provide verified property details and secure payment methods.
- **Personalization**: Features like dynamic wishlists and tailored search results are becoming critical for user satisfaction.

HomeEase addresses these trends by integrating transparent, secure, and personalized features.

Chapter 3

Motivation and Problem Definition

3.1 Problem Definition

Develop a home rentals platform featuring property listings,property booking, booking management, user reviews, and secure contact between landlords and tenants.

3.1.1 Objective of Problem Definition:

To address these challenges, HomeEase aims to provide:

- A user-friendly platform for seamless property discovery and management.
- Secure and transparent transactions for both hosts and tenants.
- Real-time communication tools to enhance clarity and reduce delays.
- Dynamic wishlist functionality for personalized user experiences.
- A scalable and reliable digital solution to modernize the rental process.

3.2 Motivation

The global rental industry is undergoing a digital transformation, driven by the need for more convenient, secure, and efficient processes. Traditional rental methods, such as manual property searches and in-person negotiations, are often time-consuming and unreliable. Online platforms have emerged as a solution, yet many fail to fully address critical challenges like transparency, trust, and user experience.

Motivating Factors for HomeEase Development:

- Growing Demand for Reliable Online Platforms:
  With increasing urbanization and a shift toward digital solutions, users need a platform that simplifies property rental management while ensuring security and efficiency.
- Bridging the Communication Gap:

Miscommunication between hosts and tenants often leads to misunderstandings and dissatisfaction. A platform with integrated communication tools can mitigate these issues.

- Personalized and User-Centric Solutions:
  Many existing platforms lack features like dynamic wishlists or tailored search options, leaving users frustrated. There is a strong demand for tools that provide a personalized experience.
- Trust and Transparency:
  Security concerns, such as fraudulent listings and payment fraud, create hesitation in adopting online rental services. A trusted system with verified listings and secure payment gateways can address these fears.

Chapter 4

Proposed Methodology

4.1 System Architecture

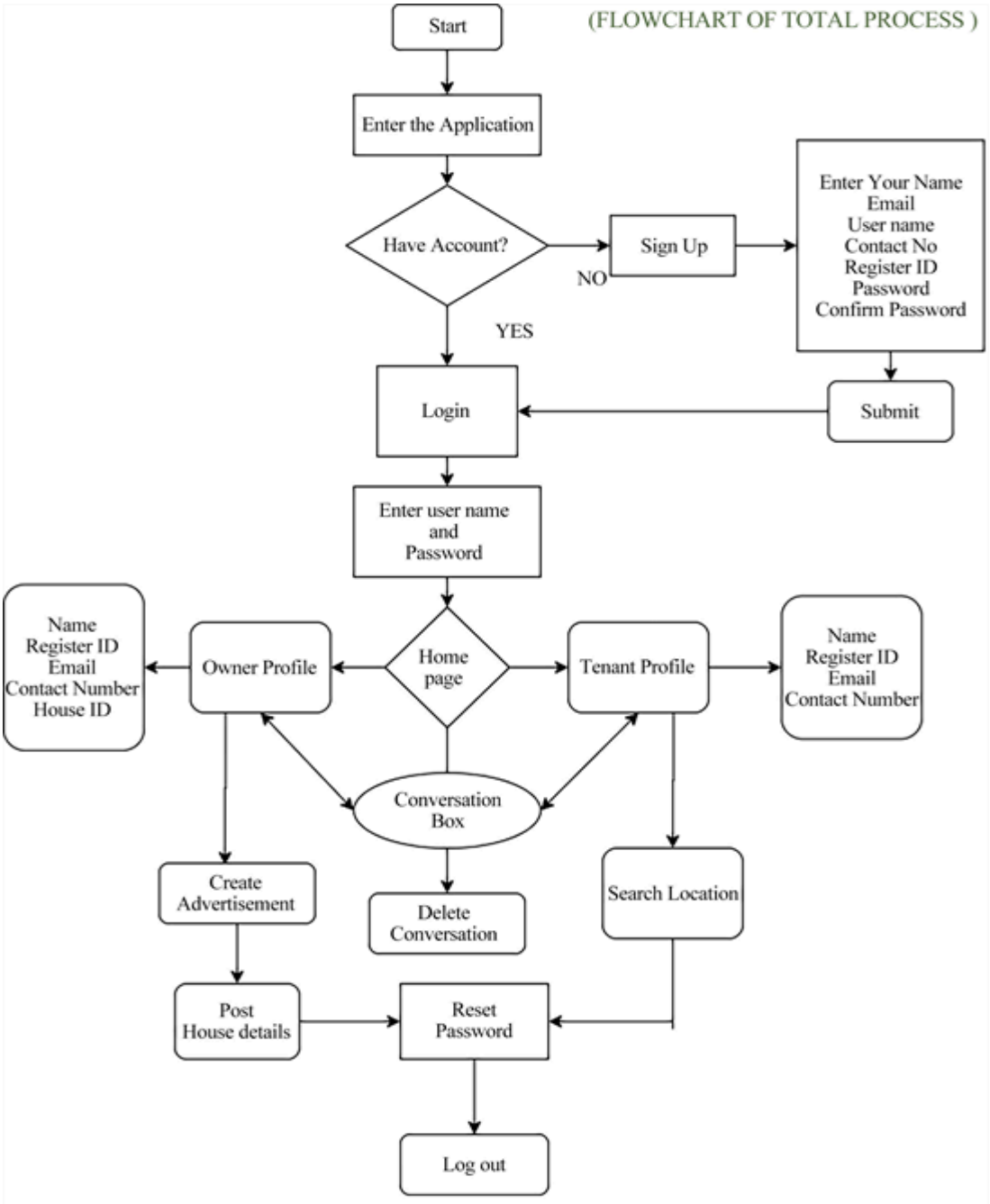

Fig 4.1 Block diagram of website working

Technology	Purpose
HTML	structure and format content on a web page
CSS	controls the visual presentation of a web page
Javascript	to add interactive elements to web pages and other applications
Bcrypt js	to securely store passwords in Node.js applications
JWS	to send information between websites
MongoDB	used to store and process data in various forms
Redux	to manage and update the global state of an application in a predictable way
React	build apps and websites faster and with less code than using plain JavaScript

Table 4.1 Project technologies and Tools

4.2 Detailed Webpages

1.Home page:

Showcased featured properties ,search bar and quick filters for easy navigation.Highlights latest listings, popular cities, and user testimonials.

2.Register page:

Offers registration for landlords and tenants , including personal information, email verification and password setup ,user friendly form with clear guidance.

3. Login page:

Allow users to securely access their accounts using email and password. features a “forgot password” recovery option and links to register for new users.

4.Create listing page:

Landlords can Now add new properties by uploading photos ,entering description , setting price ,and specifying amenities.Including preview and listing management

options.

5.Listing details:

Displays detailed property information, including photos, descriptions, pricing , availability, and amenities ,includes contact details, booking options using calendar.

6.Category page:

Organize properties into categories like apartments ,houses and vacation rentals.Features filter for location, price range and amenities to help users find suitable listings easily.

7. Search page:

Provides a dynamic search experience with filters for location, price , property type ,and amenities. Display results with a brief overview.

8.Booking list:

Shows a user upcoming and past bookings with details like property names,dates,status and total costs.Includes options to edit or cancel bookings.

9. Wishlist page :

Allows users to save and view favourite properties.displays saved listing with brief description and options to quickly access ,remove or book selected properties.

10. Property listing page:

Shows a grid or list of available properties with summaries including images, prices, location and key features.including sorting and filtering options.

11.Reservation listing page:

Displays all current and past reservation details such as property names, check/check out date, and total amount.

Chapter 5

Conclusion and Future Work

5.1 Project Summary

HomeEase is an innovative web platform designed to simplify and enhance the rental experience for both property hosts and tenants. By offering a user-friendly interface, it facilitates seamless property listing, secure booking management, and dynamic wishlist functionality, allowing users to effortlessly add or remove items.

The platform focuses on streamlining communication between hosts and tenants, ensuring transparency and fostering trust within the rental process. With robust features and a secure infrastructure, HomeEase addresses the increasing demand for reliable online rental services, providing a safe, efficient, and enjoyable experience for all users.

5.2 Concluding Remarks

The project has successfully accomplished its primary objective - **HomeEase** is a cutting-edge web platform created to streamline and elevate the rental experience for both property hosts and tenants. It provides an intuitive interface for easy property listing, secure booking management, and flexible wishlist features, enabling users to seamlessly add or remove desired properties.

The platform prioritizes transparent communication between hosts and tenants, building trust and confidence throughout the rental journey. With its comprehensive features and reliable infrastructure, HomeEase meets the growing demand for dependable online rental solutions, delivering a secure, efficient, and user-centric experience for everyone involved.

5.3 Future Enhancement

1. Integrating payment method.

2. Constructing a dedicated page for contacting landlords.

3. Making a chat option between landlords and tenants.

4. Implementation of advanced caching mechanisms for improved performance.

References

1. Problem debugging https://stackoverflow.com/
2. Redux documentation https://redux.js.org/
3. Material UI Documentation https://mui.com/material-ui/material-icons/
4. React Documentation https://react.dev/
5. MongoDB Documentation https://www.mongodb.com/

Appendices

Appendix A

Code Attachments

The following is the partial / subset of the code. Code of some module(s) have been wilfully suppressed.

A.1 Home page

```
import { useEffect, useState } from "react";

import "../styles/ListingDetails.scss";

import { useNavigate, useParams } from "react-router-dom";

import { facilities } from "../data";

import "react-date-range/dist/styles.css";

import "react-date-range/dist/theme/default.css";

import { DateRange } from "react-date-range";

import Loader from "../components/Loader";

import Navbar from "../components/Navbar";

import { useSelector } from "react-redux";

import Footer from "../components/Footer";

const ListingDetails = () => {

  const [loading, setLoading] = useState(true);

  const [error, setError] = useState(""); // State for error message
```

```
const [dateRange, setDateRange] = useState([
  {
    startDate: new Date(),
    endDate: new Date(),
    key: "selection",
  },
]);
const { listingId } = useParams();
const [listing, setListing] = useState(null);
const getListingDetails = async () => {
  try {
    const response = await fetch(
      `http://localhost:3001/properties/${listingId}`,
      {
        method: "GET",
      }
    );
    const data = await response.json();
    setListing(data);
    setLoading(false);
```

```
    } catch (err) {
      console.log("Fetch Listing Details Failed", err.message);
    }
  };
  useEffect(() => {
    getListingDetails();
  }, [listingId]);
  /* BOOKING CALENDAR */
  const handleSelect = (ranges) => {
    setDateRange([ranges.selection]);
  };
  // Calculate the number of months
  const start = new Date(dateRange[0].startDate);
  const end = new Date(dateRange[0].endDate);
    const monthCount = end.getMonth() - start.getMonth() + (12 * (end.getFullYear() - start.getFullYear()));
  // Calculate the remaining days
   const remainingDays = (end.getTime() - start.getTime()) % (30 * 24 * 60 * 60 * 1000); // Assuming 30 days in a month
   const remainingDaysInMonths = Math.ceil(remainingDays / (30 * 24 * 60 * 60 * 1000)); // Convert remaining days to months
```

```
// Update monthCount if there are remaining days
const updatedMonthCount = monthCount + remainingDaysInMonths;
useEffect(() => {
  if (monthCount < 1) {
    setError("Bookings must be at least 1 month long.");
  } else {
    setError("");
  }
}, [dateRange]);

/* SUBMIT BOOKING */
const customerId = useSelector((state) => state?.user?._id);
const navigate = useNavigate();
const handleSubmit = async () => {
  if (updatedMonthCount < 1) return; // Prevent submission if invalid

  try {
    const bookingForm = {
      customerId,
      listingId,
```

```
    hostId: listing.creator._id,
    startDate: dateRange[0].startDate.toDateString(),
    endDate: dateRange[0].endDate.toDateString(),
    totalPrice: listing.price * updatedMonthCount,
  };

  const response = await fetch("http://localhost:3001/bookings/create", {
    method: "POST",
    headers: {
      "Content-Type": "application/json",
    },
    body: JSON.stringify(bookingForm),
  });

  if (response.ok) {
    navigate(`/${customerId}/trips`);
  }
} catch (err) {
  console.log("Submit Booking Failed.", err.message);
}
```

```
};

return loading ? (
  <Loader />
) : (
  <>
    <Navbar />

    <div className="listing-details">
      <div className="title">
        <h1>{listing.title}</h1>
        <div></div>
      </div>

      <div className="photos">
        {listing.listingPhotoPaths?.map((item) => (
          <img
            src={`http://localhost:3001/${item.replace("public", "")}`}
            alt="listing photo"
          />
```

```
  ))}
</div>

<h2>
  {listing.type} in {listing.city}, {listing.province},{" "}
  {listing.country}
</h2>
<p>
  {listing.guestCount} guests - {listing.bedroomCount} bedroom(s) -{" "}
  {listing.bedCount} bed(s) - {listing.bathroomCount} bathroom(s)
</p>
<hr />

<div className="profile">
  <img
    src={`http://localhost:3001/${listing.creator.profileImagePath.replace(
      "public",
      ""
    )}`}
  />
```

```
  <h3>
    Hosted by {listing.creator.firstName} {listing.creator.lastName}
  </h3>
</div>
<hr />

<h3>Description</h3>
<p>{listing.description}</p>
<hr />

<h3>{listing.highlight}</h3>
<p>{listing.highlightDesc}</p>
<hr />

<div className="booking">
  <div>
    <h2>What this place offers?</h2>
    <div className="amenities">
      {listing.amenities[0].split(",").map((item, index) => (
        <div className="facility" key={index}>
```

```
        <div className="facility_icon">
          {
            facilities.find((facility) => facility.name === item)
              ?.icon
          }
        </div>
        <p>{item}</p>
      </div>
    ))}
  </div>
</div>

<div>
  <h2>How long do you want to stay?</h2>
  <div className="date-range-calendar">
    <DateRange ranges={dateRange} onChange={handleSelect} />
    <h2>
      ₹{listing.price} x {updatedMonthCount}{" "}
      {updatedMonthCount > 1 ? "months" : "month"}
    </h2>
```

```
            <h2>Total price: ₹{listing.price * updatedMonthCount}</h2>
            <p>Start Date: {dateRange[0].startDate.toLocaleDateString()}</p>
            <p>End Date: {dateRange[0].endDate.toLocaleDateString()}</p>
                    {error && <p className="error-message">{error}</p>} {/* Display error message */}
            <button
           className="button"
           type="submit"
           onClick={handleSubmit}
           disabled={monthCount < 1} // Disable button if booking is invalid
         >Book Now</button>
          </div>
         </div>
        </div>
      </div>
     <Footer />
    </>
  );
};
export default ListingDetails;
```