



12-B Status from UGC

REAL-ESTATE BOOKING WEBSITE

Guided By:-

**Mr.Govind Jha Sir
CEA Department**

Submitted to:-

**Department of Computer Engineering and Applications
GLA University, Mathura**

Submitted By:-

Name of Candidate: Devansh Dixit

Univ Roll No.:2215000576



**Department of Computer Engineering
and Applications GLA University, 17 km.
Stone NH#2, Mathura-Delhi Road,
Mathura – 281406 U.P (India)**

DECLARATION

We hereby declare that the project work entitled "**Real-Estate Website**" submitted to the GLA University, is a record of an original work done by Devansh under the guidance of **Mr. Govind Jha Sir**, and this project work is submitted in the partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Computer Science & Engineering. The results embodied in this thesis have not been submitted to any other University or Institute for the award of any degree or diploma.

Name of Candidate: Devansh Dixit

Univ Roll No.:2215000576



**Department of Computer Engineering and Applications
GLA University, 17 km. Stone NH#2, Mathura-Delhi Road,
Mathura – 281406 U.P (India)**

CERTIFICATE

This is to certify that the project entitled “***Real-Estate Website***”, carried out in Emersion Project , work by **Devansh Dixit** submitted in partial fulfillment of the requirements for the award of the degree Bachelor of Technology (Computer Science & Engineering).

Signature of Supervisor:

Name of Supervisor: Mr. Govind Jha

Date: 08-07-2025



**Department of Computer Engineering
and Applications GLA University, 17 km.
Stone NH#2, Mathura-Delhi Road,
Mathura – 281406 U.P (India)**

ACKNOWLEDGMENT

It gives us a great sense of pleasure to present the synopsis of the **B.Tech** Project undertaken during final Year. This project is going to be an acknowledgement to the inspiration, drive and technical assistance will be contributed to it by many individuals. We owe special debt of gratitude to **Mr. Govind Jha, Full Stack Trainer** for providing us with an encouraging platform to develop this project, which thus helped us in shaping our abilities towards a constructive goal and for his constant support and guidance to our work.

His sincerity, thoroughness and perseverance has been a constant source of inspiration for us. We believe that he will shower us with all his extensively experienced ideas and insightful comments at different stages of the project & also taught us about the latest industry-oriented technologies. We also do not like miss the opportunity to acknowledge the contribution of all faculty members of the department for their kind guidance and co-operation.

Devansh Dixit (2215000576)

ABSTRACT

The Real Estate Website project report presents the development of a web-based application designed to facilitate the search and booking of real estate properties. The objective of the project is to create a user-friendly platform that enables users to browse through property listings, view detailed information, and make bookings securely and conveniently.

The project utilizes technologies such as HTML, CSS, JavaScript, and a backend framework to construct an intuitive and responsive user interface. The backend implementation includes database management and integration of essential functionalities such as property search, property listing management, user registration and authentication, and secure booking processing.

Throughout the project, emphasis is placed on the effective gathering of user requirements, system design, and implementation based on industry best practices. The user interface is designed to enhance the user experience, providing intuitive navigation, clear property information presentation, and interactive features for property search and selection.

The Real Estate Website project report highlights the challenges faced during the development process, including database design, data management, and security considerations. The report also details the testing procedures conducted to ensure the application's functionality, usability, and performance.

By completing the Real Estate Website project, students acquire valuable practical skills in web development, database management, and user interface design. They gain hands-on experience in project planning, requirement analysis, system design, implementation, and testing. The project report demonstrates the students' ability to apply their technical knowledge to develop a functional and user-friendly real estate website.

The Real Estate Website project serves as a stepping stone for students to further explore opportunities in the real estate industry and web development domain. It equips them with the necessary skills and knowledge to contribute to the advancement of real estate technology, providing a foundation for future projects and career prospects in the field.

CONTENTS

Introduction	(7)
Motivation	(8)
Literature Review.....	(9)
Problem Statement	(10)
Hardware and Software Requirement	(11)
Basic Introduction-MERN	(12)
Project Description	(13-15)
Working.....	(16-18)
Implementation.....	(19-23)
User Interface	(24-31)
Conclusion.....	(32)
Future Works.....	(33)
References	(34)

INTRODUCTION:

The real estate industry has witnessed a significant transformation with the advent of technology and the internet. Property buyers and renters now heavily rely on online platforms to search for and book properties. The Real Estate Website project aims to develop a dynamic and user-friendly website that caters to the needs of property seekers and facilitates property bookings.

The objective of this project is to design and develop a comprehensive real estate website that offers an intuitive and efficient platform for users to explore, compare, and reserve properties. The website will act as a bridge between property owners and potential buyers or renters, simplifying the property search and booking process.

The real estate website will encompass a range of features and functionalities to enhance the user experience. It will include a user-friendly interface, an advanced search system with filters for location, property type, price range, and amenities, detailed property listings with high-quality images and descriptions, and a secure booking system that enables users to reserve properties with ease.

In addition, the website will provide valuable information about the properties, such as their proximity to essential amenities, transportation options, and neighborhood details. It will also incorporate interactive maps, virtual tours, and user reviews to assist users in making informed decisions.

For property owners, the real estate website will offer a platform to showcase their properties and attract potential buyers or renters. They will have the ability to create and manage their property listings, update availability and pricing, and communicate with interested parties.

By working on this project, students will have the opportunity to gain hands-on experience in web development, database management, user interface design, and system integration. They will apply their programming skills and knowledge of technologies such as HTML, CSS, JavaScript, and backend frameworks to create a functional and visually appealing real estate website.

Moreover, this project will provide students with insights into the real estate industry, its dynamics, and the challenges faced by property buyers and sellers. It will enable students to understand the importance of user experience, data security, and effective communication in the context of a real estate website.

MOTIVATION:

The MERN stack (MongoDB, Express.js, React.js, and Node.js) is a popular and powerful combination of technologies for building web applications. the MERN stack offers a powerful and versatile platform for building Real-Estate-Booking websites that are robust, scalable, customizable, and secure. These benefits provide strong motivation for businesses and developers looking to build successful e-commerce applications.

LITERATURE REVIEW:

The evolution of real estate booking websites has witnessed a transformative journey, progressing from basic property listings to sophisticated platforms offering immersive experiences. Studies underscore the pivotal role of user experience and interface design in these platforms, emphasizing intuitive navigation, categorized property listings, and high-quality visuals as key drivers of user engagement and satisfaction. Moreover, research highlights the integration of cutting-edge technologies like augmented reality (AR), virtual reality (VR), and artificial intelligence (AI) as catalysts for enhancing property visualization and aiding decision-making for potential buyers or renters.

Trust and credibility emerge as crucial factors in the success of real estate booking websites. Scholarly work emphasizes the significance of authentic property information, user reviews, secure payment gateways, and transparent policies in building user trust. Additionally, the exponential rise in mobile app usage for real estate transactions garners attention, with studies analyzing the impact of mobile accessibility, responsive design, and app features on user behavior and conversion rates.

Beyond user-centric aspects, literature explores the broader market impact of these platforms, investigating their influence on consumer behavior, market dynamics, and the traditional real estate industry. Challenges such as data security concerns, regulatory complexities, and market saturation are identified, prompting discussions on potential solutions and future trends. Furthermore, case studies and success stories of innovative platforms showcase the effective implementation of features, marketing strategies, and user engagement tactics, illustrating the transformative potential of real estate booking websites in shaping the industry landscape. This comprehensive literature review synthesizes multifaceted insights into the realm of real estate booking websites, encompassing technological advancements, user experiences, market impact, challenges, and future directions in this dynamic sector.

PROBLEM STATEMENT:

Despite the rise in popularity of online real-estate booking services, a platform that offers customers a thorough, user-friendly, and customized booking experience is still required. Many currently available property booking websites have a complex structure and below-average user- experience, and using them can be cumbersome and annoying. Additionally, some users can find it difficult to make judgments due to the abundance of information provided.

Goals and Objectives

- 1. Providing a comprehensive database of properties:** The primary goal of a real estate website is to offer a comprehensive database of properties for sale or rent. The website should have a user-friendly interface that allows visitors to easily search for properties based on their criteria such as location and property name.
- 2. Enhancing User Experience:** Real estate website offer a seamless user experience, making it easy for visitors to navigate and search for properties. This can be achieved by having clear and detailed property listings, high-quality images, virtual tours, and interactive maps.
- 3. Building Trust and Credibility:** Property finder websites should aim to build trust and credibility with their visitors by providing accurate and up-to-date property information, and by offering excellent customer service and support.
- 4. Generating Leads:** Property finder websites can generate leads by capturing visitor information and providing them with relevant property information. This information can be used to send targeted marketing campaigns and follow-ups.
- 5. Generating Revenue:** The ultimate goal of most property finder websites is to generate revenue through the sale or rental of properties. This can be achieved by charging fees to property owners or real estate agents for listing their properties on the site, or by earning a commission on successful sales or rentals.

6. **Secure and Convenient Booking System:** The website should incorporate a secure and user-friendly booking system that allows users to reserve properties easily. The objective is to provide a seamless booking experience with clear instructions, availability calendars, and secure payment processing.

HARDWARE AND SOFTWARE REQUIREMENTS:

Hardware Requirement

Processor	: i3/Intel or above
Operating System	: Windows 7 +
RAM	: 2GB (min)
Hardware Devices	: Desktops, Tablets, Mobiles
Hard Disk	: 128GB (min)
Keyword	: Standard

Software Requirement

Technology implemented:	Full-Stack Web Development
Language Used	: HTML, CSS, JavaScript, React
Database	: MongoDB Atlas /Compass
Cloud Storage	: Cloudinary
VCS	: Git/GitHub
Web Browser	: All Existing Browser
IDE	: VsCode

What is MERN STACK:

MERN stack is a framework used for creating websites (web app development). MongoDB, ExpressJS, ReactJS, and NodeJS make up its functional components. The specific role of each of these elements while creating a web application are listed below:

- **MongoDB:** The application data is stored in this document-oriented, No-SQL database.
- **NodeJS:** This is the JavaScript runtime environment that is used to run the JavaScript code on the machine itself, instead of a browser.
- **ExpressJS:** It is a framework that sits atop NodeJS and is used to create a website's backend using NodeJS functions and structures. NodeJS was created to run JavaScript on computers, not to create websites, so ExpressJS was created to fill that gap.
- **ReactJS:** It is a library that Facebook built. It is used to build the UI elements that go into a single page web application's user interface. The user interacts with the ReactJS UI components in the front-end of the application, which is situated in the browser. The backend of this application, which is located on a server, is served by ExpressJS, which is built upon NodeJS.

A request to change data is sent to the Express server, which is built on NodeJS, after any interaction. When necessary, Express fetches information from the MongoDB database and sends it to the application's front end, where it is shown to the user.

PROJECT DESCRIPTION:

Modules And Functionalities:

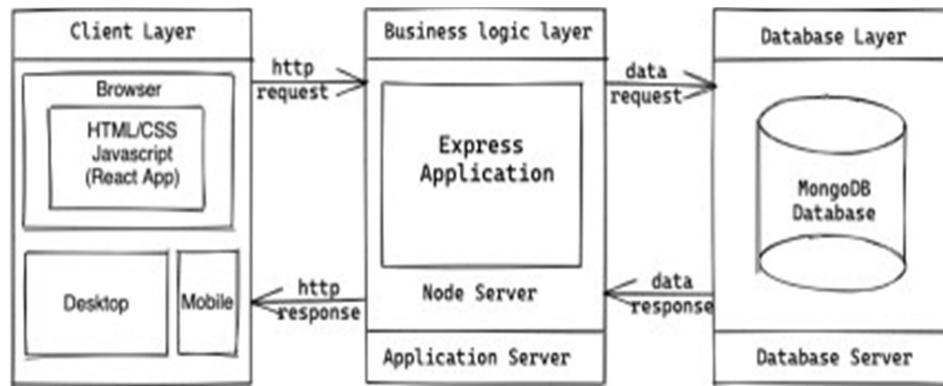


Figure 1: Various Layers of Mern Stack

The project is divided into 3 modules: -

Front end Module:

Client-side rendering, often known as front-end development, is a new style of site rendering that is employed in contemporary apps. JavaScript, which is now the de facto standard web language, is used to render the content on your computer as opposed to a distant web server in client side rendering. In actuality, this indicates that a browser is responsible for generating the HTML output of the web application and that a server is only needed to provide the raw web application. Additionally, it shows that a piece of the presentation logic—the reasoning used to create a web page and display it to the user on the screen—is handled on the client-side. With the introduction of JavaScript libraries like Angular, React, and Vue, client-side rendering became more common.

Backend Module:

Building websites and web apps has always been done using server-side rendering, also referred to as back-end web development. When we access a page, we send a request for data to the server, which processes it and sends back a response to the

browser. All the activities required to build an HTML page that the web browser can understand are carried out on the remote server that houses the website or web application when a website renders server-side. This entails processing any required logic as well as information queries from databases for that web application. While it waits for the distant server to finish processing the request and provide the response, the web browser on the other end sits idle. When a response is sent, web browsers interpret it and show the material on the screen.

Database Module:

A web database is a system for storing information that can then be accessed via a website. For example, an online community may have a database that stores the username, password, and other details of all its members. At its most simple level, a web database is a set of one or more tables that contain data.

TABLES USED IN REAL-ESTATE DATABASE:

Users:

Table 1: user Table

SNO	NAME	TYPE	DESCRIPTION
1	Id	objectId	Auto-generated id as an object
2	Name	String	Identify unique Users
3	E-mail	String	For security authentication
4	Image	String	For security authentication
5	BookedVisits	Json	Show Residencies Which you have Booked
6	Fav Residencies	String	List out Favourite Residencies

Residency:

Table 2: Residency Table

SNO	NAME	TYPE	DESCRIPTION
1	Id	String	Autogenerated id as an object
2	Title	String	Title of the Property
3	Description	String	Description of Property
4	Price	Int	Price of the Property
5	Address	String	Give the Description of Property Address
6	City	String	Tell About the City Where Property is Located
7	Country	String	Tells about the country
8	Image	String	
10	Facilities	json	Tells about the facilities
11	userEmail	String	Gives User Mail id

WORKING:

A real estate booking website typically operates through a series of steps to facilitate property transactions between buyers, sellers, and real estate agents. Here's a simplified breakdown of how it might work:

1. **Property Listing:** Real estate agents or property owners list their properties on the website. They provide details such as property type (apartment, house, commercial space), location, size, amenities, price, and photos.
2. **Search and Filters:** Users (potential buyers or renters) browse the website and use search filters (like location, etc.) to find properties that match their preferences.
3. **Property Details:** Users can view detailed information about the properties, including descriptions, photos, floor plans, amenities, nearby facilities (schools, hospitals, etc.), and contact details of the agent or owner.
4. **Booking or Inquiry:** Interested users can inquire about the property or, in some cases, directly book a viewing or make an appointment to visit the property. They may also have the option to schedule virtual tours.
5. **Communication:** The website facilitates communication between buyers/renters and sellers/agents. It may include messaging systems, contact forms, or direct contact information exchange to arrange property viewings, negotiate terms, or ask questions.
6. **Booking Process:** If the buyer decides to proceed, they might have the option to book the property online, paying a reservation fee or deposit through secure payment gateways. Contracts or agreements might be initiated or signed digitally.

Block Diagram:

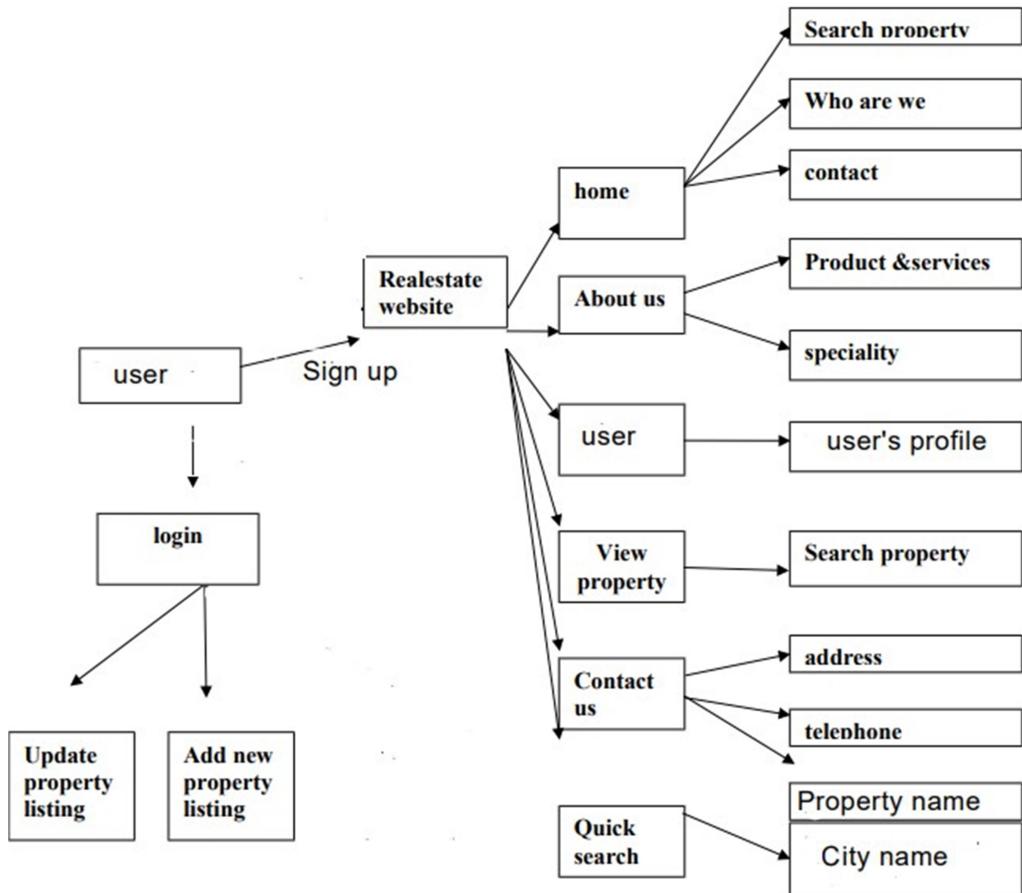


Figure 2:Block Diagram of Real Estate Website

DESCRIPTION OF THE PROCESSES

Following are the 5 main generalized processes involved in this information system:

1. user Login process
2. Property Searching Process
- 3 .Registering New User Process
4. Add Property Process

5. Book Property Visit process

1. **User Login Process:** Users access the website by creating an account or logging in. They provide their credentials (username/email and password) to gain access to personalized features like saving favorite properties, , or managing their profile information.
2. **Property Searching Process:** Users can search for properties using various filters such as location, etc. The website displays a list of properties matching the specified criteria, allowing users to view details, photos, and contact information for each property.
3. **Registering New User Process:** New users can register by providing necessary details like name, email address, and creating a password. Some websites might require additional information for verification purposes. After registration, users gain access to the website's functionalities.
4. **Add Property Process:** Property owners or real estate agents can add their properties to the website by providing detailed information about the property. This includes property type, location, size, photos, amenities, price, and any other relevant information. They might also upload documents like property deeds or floor plans.
5. **Book Property Visit Process:** Users interested in a particular property can schedule a visit or viewing through the website. They may have the option to select a convenient date for the visit. The website might facilitate communication between the user and the property owner/agent to confirm the visit and provide any additional details.

IMPLEMENTATION:

Java script is a scripting language used to enhance the functionality of the browser. Java script is integrated with HTML and CSS. Java script facilitates the developer with properties related to document windows, frames, loaded documents and links.

React is a popular JavaScript library used for building user interfaces. It is used for frontend due to its flexibility, reusability, and ease of use.

For the Back-end part we will use Node.js which can generate dynamic page content. It also helps to open , read, write and close files on the server and collect Login Authentication data. Node.js can add, delete, modify data in your database.

For Data Storage, we use MongoDB(Atlas, Compass), it is a popular NoSQL database it can handle large amounts of data and high traffic loads.

index.js:

```
import express from 'express';
import dotenv from 'dotenv';
import cookieParser from 'cookie-parser';
import cors from 'cors';
import { userRoute } from './routes/userRoute.js';
import { residencyRoute } from './routes/residencyRoute.js';

dotenv.config()

const app = express();

const PORT = process.env.PORT || 3000;

app.use(express.json())
app.use(cookieParser())
app.use(cors())

app.listen(PORT, ()=> {
    console.log(`Server is running on port ${PORT}`);
});

app.use('/api/user', userRoute)
app.use("/api/residency", residencyRoute)
```

index.css :

```
:root {  
    --primary: #1f3e72;  
    --secondary: rgba(255, 255, 255, 0.78);  
    --black: #131110;  
    --blue-gradient: linear-gradient(97.05deg, #4066ff 3.76%, #2949c6 100%);  
    --orange-gradient: linear-gradient(270deg, #ffb978 0%, #ff922d 100%);  
    --blue: #4066ff;  
    --lightBlue: #eeeeff;  
    --shadow: 0px 23px 21px -8px rgba(136, 160, 255, 0.25);  
}  
  
* {  
    margin: 0;  
    padding: 0;  
    box-sizing: border-box;  
    font-family: "Poppins", sans-serif;  
    scroll-behavior: smooth;  
}  
  
.wrapper{  
    background-color: white;  
}  
  
.a {  
    color: inherit;  
    text-decoration: none;  
}  
  
.paddings {  
    padding: 1.5rem;  
}  
  
.innerWidth {  
    width: 100%;  
}  
  
.flexCenter {  
    display: flex;  
    row-gap: 2rem;  
    justify-content: center;  
    align-items: center;  
    flex-wrap: wrap;  
}  
  
.flexStart {  
    display: flex;  
    justify-content: flex-start;  
    align-items: center;  
}
```

```
.flexEnd {
  display: flex;
  justify-content: flex-end;
  align-items: center;
}

.flexColCenter {
  display: flex;
  justify-content: center;
  align-items: center;
  flex-direction: column;
}
.flexColStart {
  display: flex;
  flex-direction: column;
  justify-content: center;
  align-items: flex-start;
}
.flexColEnd {
  display: flex;
```

index.html :

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <link rel="icon" type="image/svg+xml" href="/vite.svg" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <link rel="preconnect" href="https://fonts.googleapis.com" />
    <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin />
    <link
      href="https://fonts.googleapis.com/css2?family=Poppins:wght@400;500;600;700&display=swap"
      rel="stylesheet"
    />
    <title>Real Estate</title>
    <script
      src="https://upload-widget.cloudinary.com/global/all.js"
      type="text/javascript"
    ></script>
  </head>
  <body>
    <div id="root"></div>
    <script type="module" src="/src/main.jsx"></script>
  </body>
</html>
```

Schema.prisma :

```
model User {
    id      String    @id @default(auto()) @map("_id") @db.ObjectId
    name    String?
    email   String    @unique
    image   String?
    bookedVisits Json[]
    favResidenciesID String[]    @db.ObjectId
    ownedResidencies Residency[] @relation("Owner")
}

model Residency {
    id      String    @id @default(auto()) @map("_id") @db.ObjectId
    title   String
    description String
    price    Int
    address   String
    city     String
    country   String
    image    String
    facilities Json
    userEmail String
    owner    User      @relation("Owner", fields: [userEmail], references: [email])
    createdAt DateTime @default(now())
    updatedAt DateTime @updatedAt
}
```

UserRoute.js :

```
import express from "express";
import {
    bookVisit,
    cancelBooking,
    createUser,
    getAllBookings,
    getAllFavorites,
    toFav,
} from "../controllers/userCntrl.js";
import jwtCheck from "../config/auth0Config.js";
const router = express.Router();

router.post("/register", jwtCheck, createUser);
router.post("/bookVisit/:id", jwtCheck, bookVisit);
router.post("/allBookings", getAllBookings);
router.post("/removeBooking/:id", jwtCheck, cancelBooking);
router.post("/toFav/:rid", jwtCheck, toFav);
router.post("/allFav/", jwtCheck, getAllFavorites);
export { router as userRoute };
```

Package.json

```
  "name": "server",
  "version": "1.0.0",
  "main": "index.js",
  "license": "MIT",
  "type": "module",
  ▷ Debug
  "scripts": {
    "start": "nodemon index.js",
    "postinstall": "prisma generate"
  },
  "dependencies": [
    "@prisma/client": "^5.6.0",
    "bcryptjs": "^2.4.3",
    "cookie-parser": "^1.4.6",
    "cors": "^2.8.5",
    "dotenv": "^16.3.1",
    "express": "^4.18.2",
    "express-async-handler": "^1.2.0",
    "express-oauth2-jwt-bearer": "^1.6.0",
    "mongoose": "^8.0.1",
    "nodemon": "^3.0.1",
    "prisma": "^5.6.0",
    "react-icons": "^4.12.0"
  ]
```

USER INTERFACE:

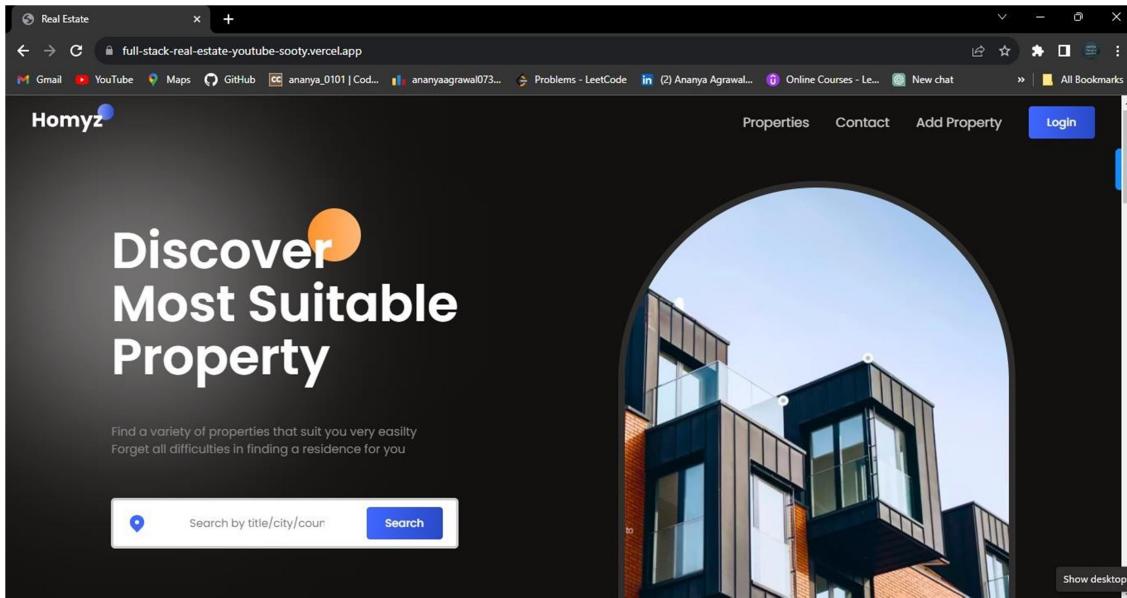


Figure 3: Website Interface

LOGIN OR SIGN UP:-

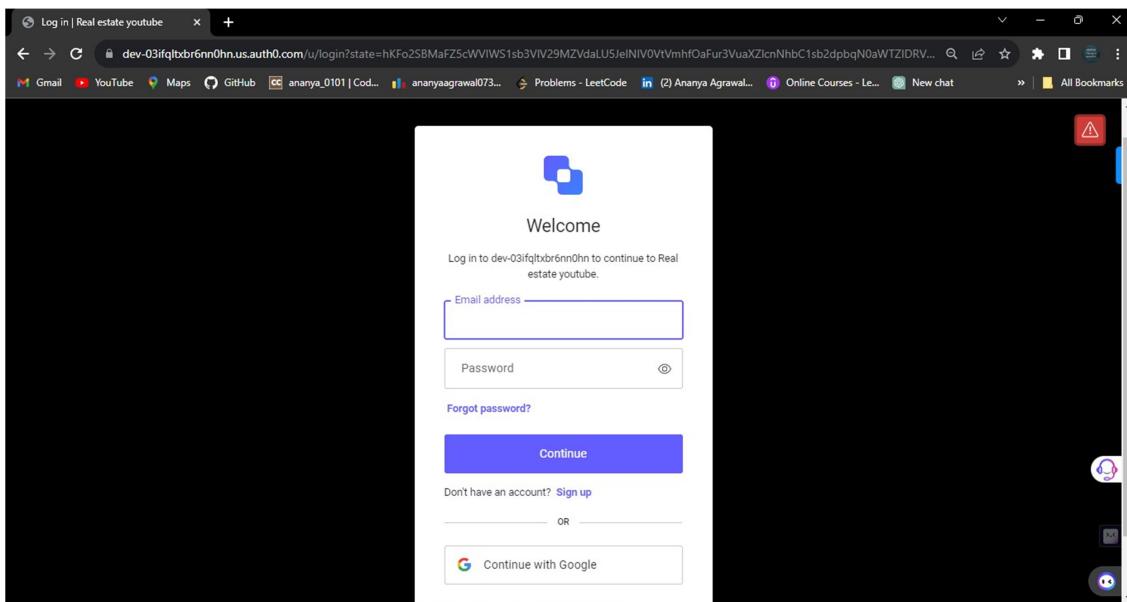


Figure 4: Login and Sign up Page

HOME PAGE:-

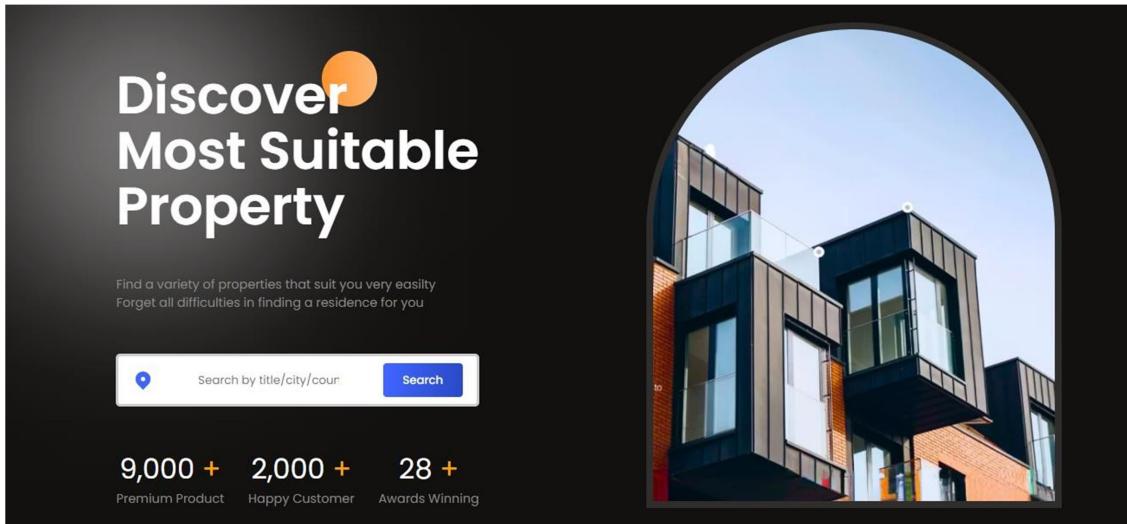


Figure 5:Home Page

A screenshot of a real estate website's "Our Value" section. On the left is a large circular inset image of a modern house at night, illuminated from within. To the right, the section is titled "Our Value" in yellow and "Value We Give to You" in dark blue. Below this is a sub-headline: "We always ready to help by providing the best services for you. We believe a good place to live can make your life better". Three service offerings are listed in boxes with dropdown arrows: "Best interest rates on the market", "Prevent unstable prices", and "Best price on the market". A small explanatory text follows: "Exercitation in fugiat est ut ad ea cupidatat ut in cupidatat occaecat ut occaecat consequat est minim minim esse tempor laborum consequat esse adipisicing eu reprehenderit enim."

Figure 6:Home Page

FEATURED PROPERTIES: -



\$10000

Beautiful House

This is a good home to live



\$5000

ZKC Villa

Its an amazing house to live



\$5000

The Johnson ...

This is an amczing place to live with all facilities around you



\$8000

Jansheen villan

Villa is a marvelous piece of luxury and Art. Decorated with two helipads, wi..



\$2000

Modern Marve...

Massive opportunity to build your dream



\$2000

Cedar Ridge ...

Massive opportunity to build your dream



\$2000

Crimson Peak...

Massive opportunity to build your dream



\$2000

Summerhill E...

Massive opportunity to build your dream

Figure 7:Property Listing Page



Crimson Peak Chalet

\$ 2000

3 Bathrooms Parking 4 Room/s

Massive opportunity to build your dream home at the base of Mummy Mountain in the 3 C's school district. Home is currently updated and very livable if your plans are to build at a later date.* Bonus * to live hillside without hillside restrictions in the town of PV. Run don't walk to capture this needle in a hay stack.



Figure 8:Property Description Page

SEARCH AND CATEGORIES:-

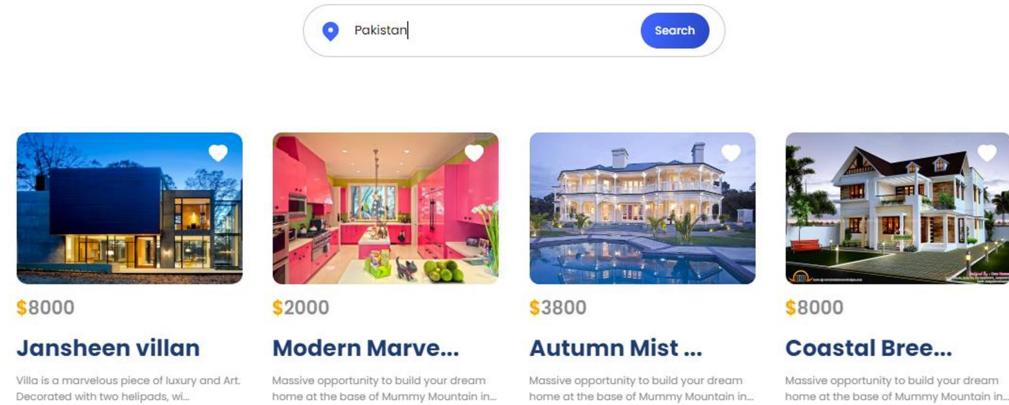


Figure 9:Search Property Page

ADD New Property :-

The screenshot shows a step-by-step form for adding a new property. Step 1: Location Address. It includes fields for Country (India IN), City (Gandhinagar), and Address (Shop no. 23 Near central Bank gandhinagar). To the right is a map of Gujarat, India, with Ahmedabad marked. A 'Next Step' button is at the bottom.

Figure 10:Add new Property Page

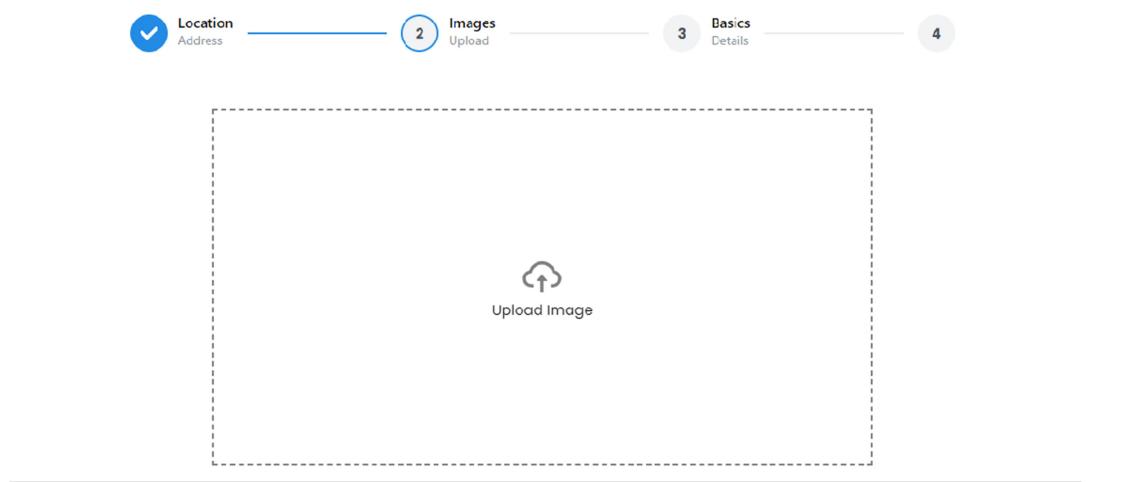


Figure 11: Add Image Section Page

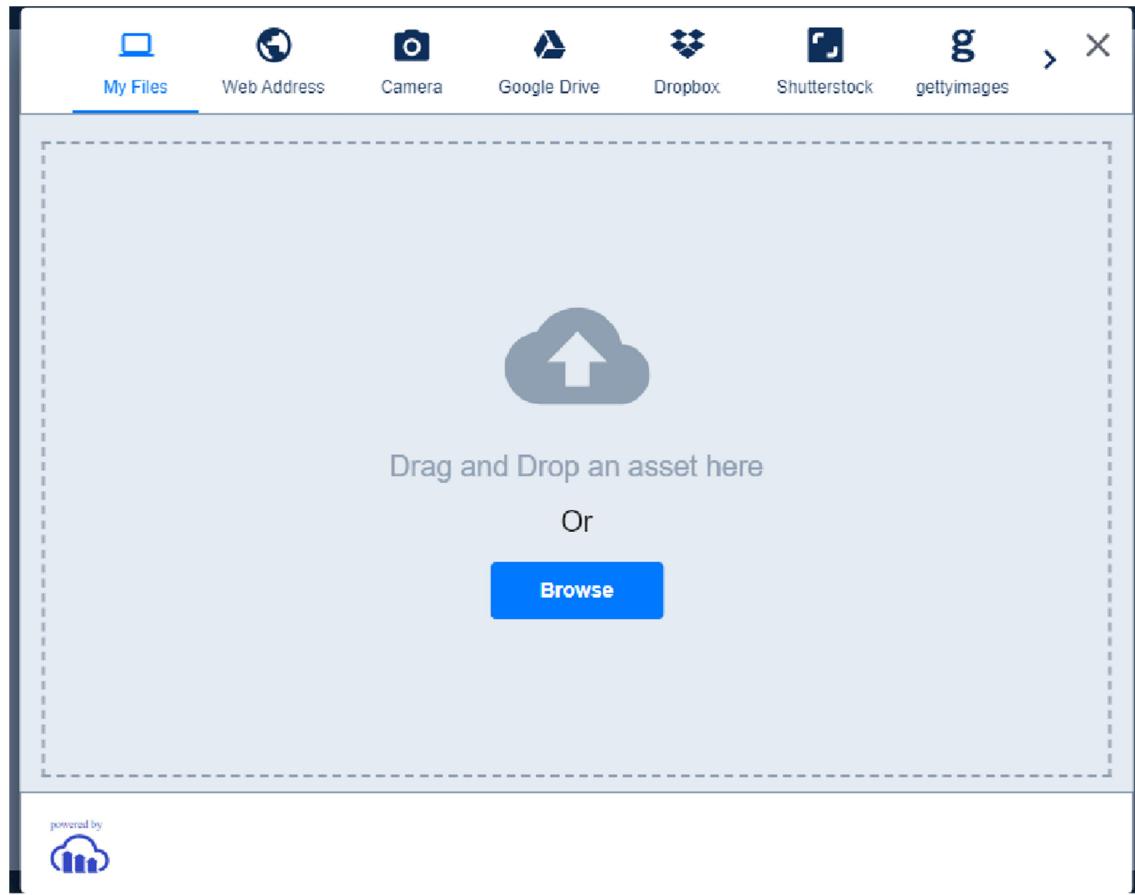


Figure 12: Upload Image section



Back

Next

Figure 13:Image to be added

Location Address Images Upload Basics Details 4

Title *
Joahan Beautiful Mansion

Description *
Massive opportunity to build your dream home at the base of Mummy Mountain in the 3 C's school district. Home is currently

Price *
100

Back Next step

Figure 14:Description of Property

Location Address

Images Upload

Basics Details

4

No of Bedrooms *

4

No of Parkings

1

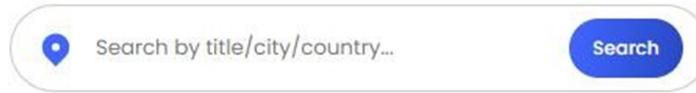
No of Bathrooms *

3

Back Add Property

Figure 15:Details of Property

Show Bookings You have Done



\$4000

Heaven On Earth

Very beautiful house

Figure 16:Bookings Page

Databases

Residency Database

The screenshot shows the MongoDB Compass interface for the 'Real_estate.Residency' database. At the top right, it displays '19 DOCUMENTS' and '2 INDEXES'. Below the header, there are tabs for 'Documents', 'Aggregations', 'Schema', 'Indexes', and 'Validation'. A search bar at the top says 'Type a query: { field: 'value' } or [Generate query](#)'. Below the search bar are buttons for 'Explain', 'Reset', 'Find', and 'Options'. There are also buttons for 'ADD DATA' and 'EXPORT DATA'. The main area shows a single document with the following fields and values:

```
_id: ObjectId('64392d39dfd90fb29e464fb7')
title: "Sunny Meadow Cottage"
description: "Massive opportunity to build your dream home at the base of Mummy Mountain"
price: 6000
address: "Street 1"
city: "Chicago"
country: "US"
image: "https://images.pexels.com/photos/7031406/pexels-photo-7031406.jpeg?auto=compress&cs=tinysrgb&h=350&w=350"
userEmail: "mukhtarhamza294@gmail.com"
createdAt: 2023-04-14T10:38:48.762+00:00
updatedAt: 2023-04-14T10:38:48.762+00:00
facilities: Object
```

Figure 17:Residency Database

User Database

The screenshot shows the MongoDB Compass interface for the 'Real_estate.User' database. At the top right, it displays '2 DOCUMENTS' and '2 INDEXES'. Below the header, there are tabs for 'Documents', 'Aggregations', 'Schema', 'Indexes', and 'Validation'. A search bar at the top says 'Type a query: { field: 'value' } or [Generate query](#)'. Below the search bar are buttons for 'Explain', 'Reset', 'Find', and 'Options'. There are also buttons for 'ADD DATA' and 'EXPORT DATA'. The main area shows a single document with the following fields and values:

```
_id: ObjectId('653dc27d9726e5a64c1bf9d3')
email: "ssss55@gmail.com"
```

Figure 18:User Database

CONCLUSION:

The development of a real estate website booking system brings significant value to the real estate industry by simplifying the property booking process and enhancing the overall user experience. By successfully implementing key features such as an intuitive user interface, advanced search functionality, detailed property listings, and a secure booking system, the booking system improves efficiency and convenience for property seekers and property owners alike.

Through the creation of a real estate website booking system, developers gain hands-on experience in web development, database management, user interface design, and system integration. They apply their technical skills and knowledge of technologies such as HTML, CSS, JavaScript, and backend frameworks to create a functional and user-friendly platform. Additionally, they learn about the importance of data security, privacy, and performance optimization in the context of a booking system.

The real estate website booking system project also provides valuable insights into the real estate industry, its dynamics, and the challenges faced by property buyers and sellers. Developers gain an understanding of the importance of effective communication, accurate property information, and a streamlined booking process. By implementing a secure and convenient booking system, they contribute to building trust and confidence among users.

In conclusion, the development of a real estate website booking system is a valuable project that equips developers with practical skills, knowledge, and insights in the fields of web development and the real estate industry. The project enhances problem-solving abilities, project management skills, and software development expertise, preparing developers for future endeavors in the technology and real estate sectors. With an efficient and secure booking system, users can easily reserve properties, while property owners can effectively manage bookings, ultimately facilitating smoother transactions and enhancing the overall real estate experience.

FUTURE WORK

While the real estate website project has provided a solid foundation, there are several avenues for future work and enhancements to further improve the website's functionality and user experience. Some potential areas for future work include:

Enhanced Property Filtering and Sorting: Implement more advanced filtering and sorting options for property search results. This could include filtering by additional criteria such as property size, number of rooms, proximity to amenities, and sorting options based on price, popularity, or user ratings.

Real-Time Booking Availability: Develop a real-time booking availability feature that allows users to see the current availability status of properties. This can help users make informed decisions and avoid conflicts in booking.

Integration with External APIs: Explore integration with external APIs to enhance the website's functionality. For example, integrate with a payment gateway API for seamless and secure payment processing, or integrate with property valuation APIs to provide estimated property values.

User Reviews and Ratings: Implement a user review and rating system where users can provide feedback and ratings for properties they have booked or visited. This can help future users make informed decisions and enhance transparency in the property booking process.

Social Media Integration: Integrate social media sharing capabilities, allowing users to easily share properties or their booking experiences on social platforms. This can increase the visibility of the website and attract a wider audience.

Responsive Design Optimization: Continuously optimize the website's responsive design to ensure seamless user experience across different devices and screen sizes. Regular testing and updates can help address any compatibility issues and improve performance.

Mobile Application Development: Consider expanding the project by developing a dedicated mobile application for iOS and Android platforms. This can provide users with a native and optimized experience on their mobile devices, leveraging device-specific features and capabilities.

REFERENCES: -

Websites

- i) <https://developer.mozilla.org/en-US/>
- ii) <https://www.w3schools.com/>
- iii) <https://css-tricks.com/>
- iv) <https://tailwindcss.com/>

Books

- i) React in Action by Mark Tielens Thomas.
- ii) The Full Stack Developer: Your Essential Guide to the Everyday Skills Expected of a Modern Full Stack Web Developer By Chris Northwood.

GitHub Repository Link

<https://github.com/dixitdevansh22/RealEstateWebsite>

Deployment Link

<https://full-stack-real-estate-youtube-sooty.vercel.app/>