

Project Title: Elite Property Management System (Housing)

Team:

Srikeerthana Reddy Bandi

Section 1: Web App Architecture

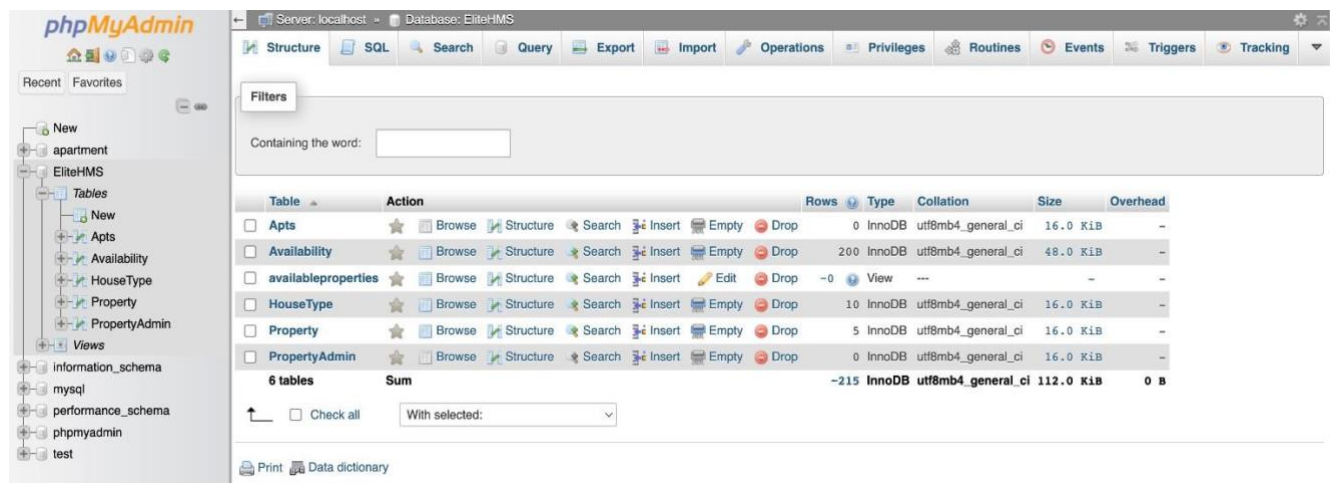
Model: We are using MySQL relational database management system for storing our data.

View: We are using HTML, CSS (Bootstrap framework), and JavaScript to create and customize web pages.

Controller: We are using PHP for the backend to perform all the CRUD operations.

Where is your data stored?

We are using MySQL to store our housing data. We have imported the SQL query file into phpMyAdmin to manage the database and establish connections easily.



What languages will be used to build back-end?

We are using PHP which is a server-side scripting language. The main reason to choose PHP is that it has built-in support for MySQL database, which makes it easy to interact with the data and build dynamic web applications. We used Visual Studio code to write our PHP code and used XAMPP for testing our application locally before deploying it to a live web server.

How you will be accessing the database: what connections and how secure?

We have established a connection between PHP and MySQL using the MySQLi extension, which is a PHP extension used to communicate with the MySQL database.

```
// Database credentials
$host = "localhost";
$user = "root";
$password = "";
$database = "EliteHMS";

// Create a database connection
$conn = mysqli_connect($host, $user, $password, $database);

// Check if the connection was successful
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}
```

To connect to the MySQL database, we have passed the host, user, password, and database credentials to the `mysqli_connect()` function. This function establishes a connection to the MySQL server. Once the connection is established, we can execute queries on the database.

To prevent the data from being modified by anyone, we will be implementing database security techniques, such as role-based access control, and establishing authentication i.e., setting username and password for admins, as these are essential for developing secure and reliable applications.

What will you use to create a front-end layout?

We are using HTML, CSS (Bootstrap Framework), JavaScript to create the front-end layout of our web application. HTML is used to define the structure and content of a web page, while CSS is used to style and format the page. With PHP as the server-side scripting language we were able to create dynamic layouts that are interactive and engaging.

Where is your application deployed?

We will be using MR Cloud hosting services to deploy our application. We chose MR Cloud as their servers are configured to ensure optimal performance for PHP-based applications, ensuring that the application runs smoothly and efficiently. Also, by using the latest hardware and software technologies they provide high-speed servers which ensure that the application is always available to the users.

How will you provide interactivity for your app?

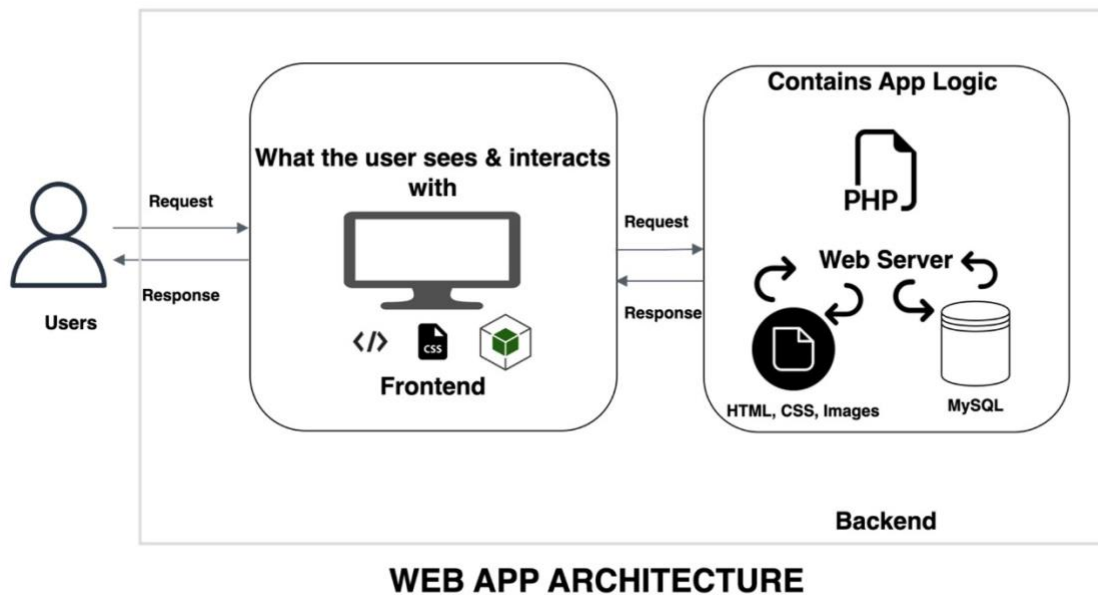
For our generic users, the application will allow them to browse different properties and select the one that best suits their needs. Once a property is selected, the user will be taken to a new page that showcases the different house types available within that property. This will allow them to easily compare and choose the right house type for them.

For our admins, the application provides a comprehensive dashboard that allows them to manage different house types within each property. They will be able to add or delete house types and

update the availability of each type as needed. Any changes made by the admin in the dashboard will be automatically reflected in the user portal.

Overall, our application is designed to provide a seamless and interactive experience for both our generic users and admins.

Web App Architecture:



Section 2: Web App Layout

What is the Initial Layout?

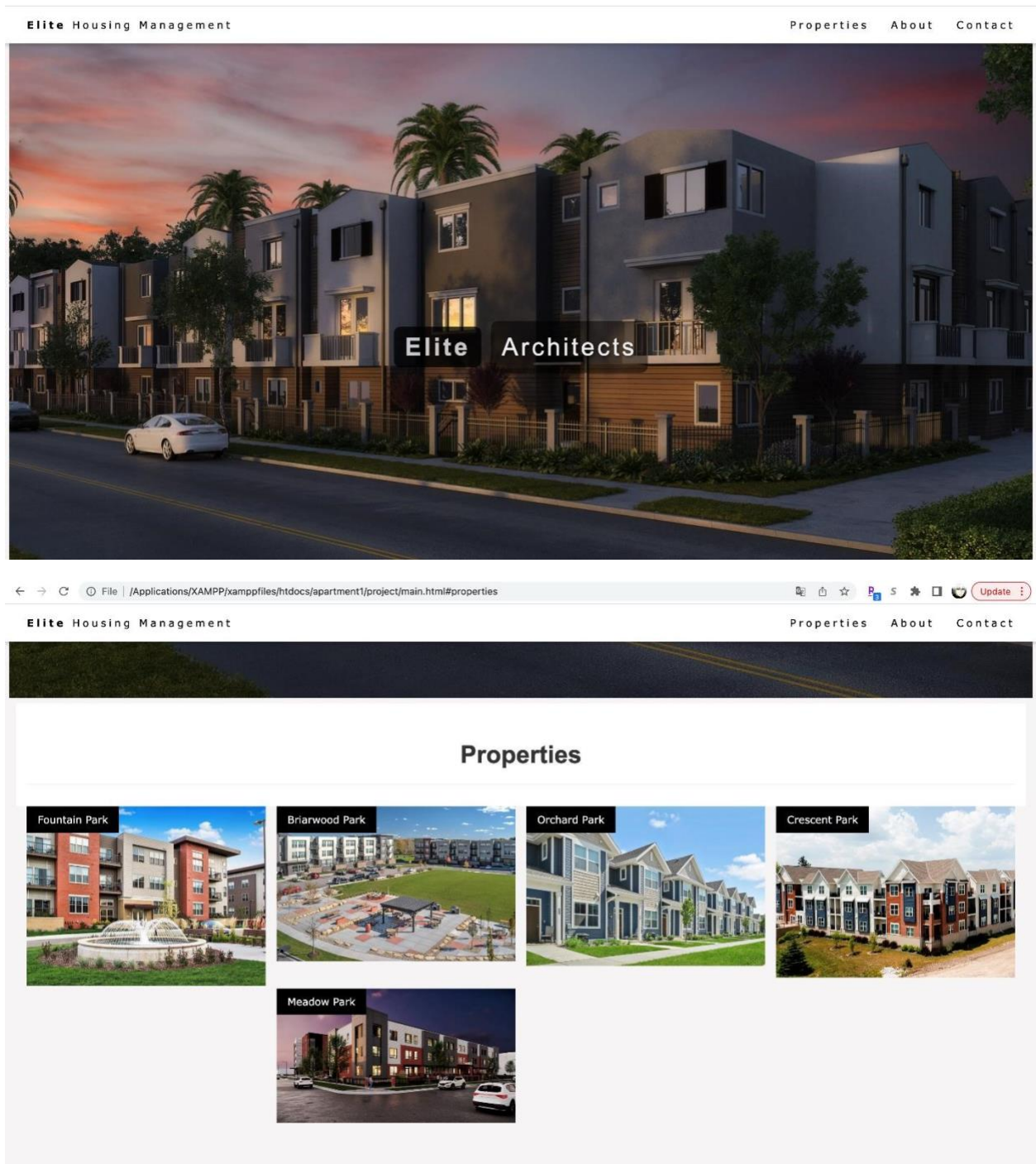
Our application caters to two types of users:

- a) Generic Users (those who are looking to either lease or buy a house)
- b) Admins (those who will be performing administrative tasks/ CRUD operations)

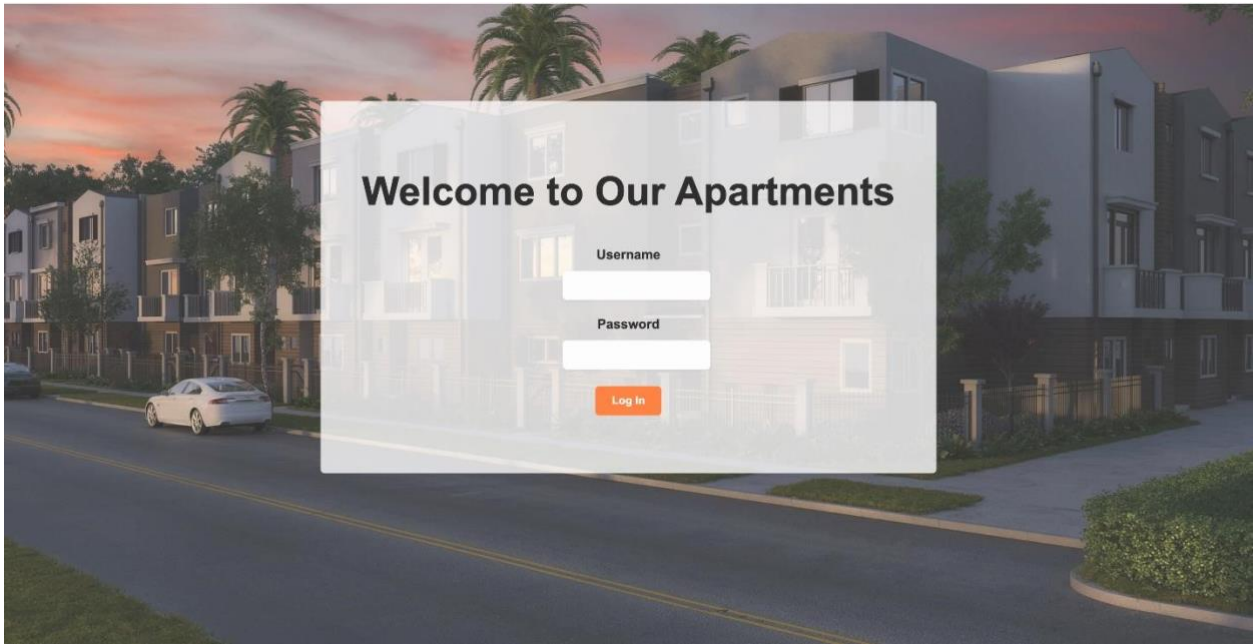
Depending on the type of user, the landing page will appear differently. For generic users, the landing page will provide a user-friendly interface that showcases different properties or communities available for either leasing or buying. This will allow them to easily navigate through the application and find the property that best suits their needs.

On the other hand, admins will have a different landing screen that is specifically designed for their role. They will need to login to the application using their credentials and once logged in, admins will see a Dashboard where they can perform CRUD (create, read, update, delete) operations. This will include the ability to add or remove properties, update property details.

a) User View:



b) Admin View:



The below screen is displayed once the admin logs in

EliteHMS

Dashboard

Logout

Welcome admin

Add / Edit Availability

Property Name

House Type

Availability

Add

Availability

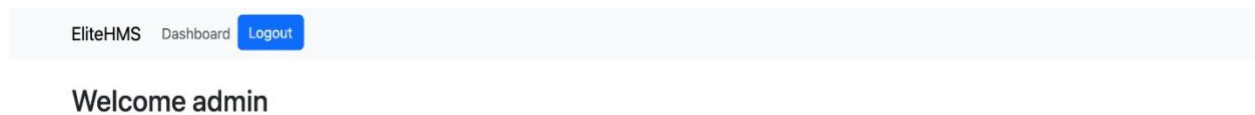
ID	Property Name	House Type	Availability	Actions
	Fountain Park	Cozy Loft	Yes	<div>EditDelete</div>
	Fountain Park	Cozy Loft	Yes	<div>EditDelete</div>
	Fountain Park	Cozy Loft	Yes	<div>EditDelete</div>

The menu panel is placed on top of the screen for both the user types.



If a user clicks on the 'Properties', the page will automatically scroll down to the Properties section on the same page. This function operates in the same way for the 'About' and 'Contact' sections.

The menu panel for admins will include two options, namely Dashboard and Logout, as shown below.



How many pages do you need? Or will you be using Tabs?

Our application is designed to have a simple and intuitive interface, with a total of four pages that are tailored to the needs of our two types of users.

What each page or Tab will display?

For our generic users, we have two pages that they will interact with. The first page is the landing page, which provides a user-friendly interface that allows users to browse different properties and select the one that best suits their needs. Once a property is selected, the user will be taken to the second page, which showcases the different house types available within that property.

For our admins, we also have two pages that they will interact with. The first page is the login page, where they will need to enter their credentials to access the application. Once logged in, they will be taken to the second page, which is the dashboard. Here, admins will be able to perform CRUD operations, including adding or removing properties and managing availability status.

What is the color schema?

To ensure clarity and ease of use, we have chosen to use a yellow color for the Edit button and a red color for the Delete button in our application.

What functionalities will be available and how users will access them?

Currently, we have Buttons (Add, Edit, Delete, Login, Logout) which allow the users to perform specific actions, such as adding a new property, deleting an existing property, or updating house types. We are planning to implement a Search Box or a Drop-Down menu in the Admin UI so that they can select particular result (apartment) to either update, delete or read.