**Home Rent Management System**

Group Members:

|  |  |
| --- | --- |
| SEIP ID | Name |
|  | MINHAZUR RAHAMAN |
|  | MD. NAZMUL HASAN |
|  | MD.SHEKH FARID |
|  | MD. HUMAYAN KABIR |

Objective:

**General objective**: The main objective of our project is to develop online home rental management system for a busy and crowded city area.

**Specific objectives:**

* To avail our project goal our specific topics given below:
* Prepare an online home rental system for the home finders and home owner as well.
* Customer can easily register at any point of time sitting in their homes to rent the home.
* To avoid face to face conversation which is time consuming and provide a fully functional automated home Rental Management System that will be an online system.
* To ensure system secure payment system.
* To provide a complete organized and reliable system with least possibility of any errors.
* To ensure best rent to the customer by providing facility to compare the room price with details information.
* Anybody can confirm his/her flat booking, from any place
* Both home owner and tenant will get payment report.

Short Overview:

|  |  |
| --- | --- |
| Roles | Description |
| Super Admin | A super admin can see apartment, rate the rooms also comment in any apartment details. Super admin can log in to the system and can manage categories, apartment, tenant, tenant request, notification, email, profile, comments, message, user role. Super admin has all every power for maintain the website also can manage some of the statistics data of the system about apartments, categories, tags, tenant, home owner. Super admin can see statics data with graph. Can manage all admins. |
| Admin | An admin can see apartment, rate the rooms also comment in any apartment details. Admin can log in to the system and can manage categories, apartment, tenant, tenant request, notification, email, profile, comments, message, user role. Admin can manage some of the statistics data of the system about apartments, categories, tags, tenant, home owner. Super admin can see statics data with graph. |
| Home Owner | A home owner can see apartment, rate the rooms also reply the comment in any apartment details. Home owner can log in to the system and can manage own apartment, profile home owner can see of some statistics data of the system about his own apartments, tenant. |
| Tenant | A tenant can see apartment, rate the rooms also comment and review in any apartment details. tenant can log in to the system and can manage own apartment, profile home owner can see of some statistics data of the system about his own apartments, tenant. |

Modules:

|  |  |
| --- | --- |
| **Common Modules (M1)** | |
| Function | Features |
| M1F1 | There will be a login page with 2 input fields one will take the users email address and another will take input of the user’s password. There will also be a login button which will be used to submit form data for successful login to the system using a post request. |
| M1F2 | A backend form validation will occur which will validate the user’s credentials submitted if they are secure to pass to the database for validation.   * Validations includes: * Valid email syntax * Null email and password field validation. * Password length min. 8 characters * Password contains alphanumeric characters.   If any invalid input is given then error message should be shown to the view page accordingly. |
| M1F3 | Upon passing the validation steps the user must be searched by the email address in the database and get the users email, username and password. Then the password must be checked with the user submitted password if it matches with the database password for validation. If it matches then necessary credentials such as user type, user name should be saved in session of cookie storage and redirected to the home page of the user using get request. Upon not finding the user form database or password mismatch an error message must be shown to the view. |
| M1F4 | Users can register the system from a registration page where there will be some input fields including name, email, password, confirm password. Users need to fill up the form and make a post request with the help of a button named sign up. |
| M1F5 | Upon submission of the registration post request a backend validation will take place that validates if all the inputs are valid and match the database data and types.  Validations include:   * Full name length must be between 3 to 30 characters. * Full name must be alphabetic * Email length must be between 10 to 50 characters. * Email address should be a valid email syntax. * Password must be between 8 to 20 characters and it must be alphanumeric. * Password and Confirm Password field must match their input data. * Full name, email, user name, password, confirm password, phone number. cannot be empty. |

|  |  |
| --- | --- |
| **Super Admin Modules (M2)** | |
| Function | Features |
| M2F1 | A Dashboard should be developed in the system in the route “dashboard/superadmin” were multiple types of information or overview would be Present such as:   * Showing number of total admin, owner, tenant in the card * Showing of top rent room list for current date and last seven days * Showing of top rent room list for current date and last seven days * Showing of earning current date and last seven days |
| M2F2 | Super Admin will be inserted from seeder. He can edit his own profile |
| M2F3 | Roles can be inserted, updated and deleted by Super Admin |
| M2F4 | Users can be inserted, updated and deleted by Super Admin |
| M2F5 | Apartments can be inserted, updated and deleted by Super Admin |
| M2F6 | Tags can be inserted, updated and deleted by Super Admin |
| M2F7 | Can accept or reject order |

|  |  |
| --- | --- |
| **Admin Modules (M3)** | |
| Function | Features |
| M3F1 | A Dashboard should be developed in the system in the route “dashboard/admin” were multiple types of information or overview would be Present such as:   * Showing number of total owners, tenant in the card * Showing of top rent room list for current date and last seven days * Showing of top rent room list for current date and last seven days * Showing of earning current date and last seven days |
| M3F1 | Users can be inserted, updated by Admin |
| M3F2 | Apartments can be inserted, updated by Admin |
| M3F3 | Tags can be inserted, updated by Admin |
| M3F4 | Can accept or reject order |
| M3F5 | He updates his own profile |

|  |  |
| --- | --- |
| **Home Owner Modules (M4)** | |
| Function | Features |
| M4F1 | A Dashboard should be developed in the system in the route “dashboard/homeowner” were multiple types of information or overview would be Present such as:     * Showing number of total own tenant in the card * Showing number total comment in own apartment post * Showing of earning current date and last seven days |
| M4F2 | He updates his own profile |
| M4F3 | Can insert, update and soft-delete room information |
| M4F4 | Can reply in the review in the room details |

|  |  |
| --- | --- |
| **Tenant Modules (M5)** | |
| Function | Features |
| M5F1 | A Dashboard should be developed in the system in the route “dashboard/homeowner” were multiple types of information or overview would be Present such as:     * Showing of expense of current date and last seven days |
| M5F2 | Can book/order or remove any preferred room |
| M5F3 | Can Review and comment in the room details |

Master Data:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Users | Profiles | Roles | Apartments | Apartment Description |