HOSTEL MANAGEMENT

OVERVIEW :

This is a one stop app for hostel management. A hostel ecosystem comprises a few very important entities namely; students,hostel wards, hostels blocks/rooms, mess. We have made sure that we covered all aspects of the ecosystem making it a very handy and accessible application.
In our app we have distinct sections dedicated to the wardens, students and mess admins. The wardens can allocate rooms and post updates for the students(regarding general hostel related updates and complaint status). The students can look for their room allocations, post complaints, view updates posted both by the wardens and mess admins as well as post ratings for the mess food. The mess admin posts the mess menu and views the rating given by the students for a day's food.

FRONTEND :
Login/Signup
HostelAdmin(warden)
       AdminLanding Page
            Hostel Allotment
            Hostel complaints
            Post Updates
StudentLanding
       Updates
       Hostel complaints
       Mess feedback
MessAdmin
       Post Updates
       FeedBack

All of the above are individual pages in our application. The pages are written with indentation to give a sense of hierarchy and flow of the pages. Below is a brief description of the pages.

Login/Signup page : This page helps users to access the part of the application dedicated to them
HostelAdmin: This is a landing page for the Hostel Admin (warden). The page is a buffer that helps navigate to the Hostel Allotment, Hostel complaints, Post Updates pages.
Hostel Allotment: The warden allocates rooms to the students in this page.
Hostel complaints: The warden can read and get updated on complaints given by the students through this page.
Post Updates: This page allows the warden to post updates to the students.

StudentLanding: This a static page with a navbar for the students to navigates to its children pages.
Updates: This page carries the updates posted by the wardens for the students to view.
Hostel complaints: This page is for posting hostel complaints that will be visible to the warden.
Mess feedback: This page like the above page is for posting feedback for the mess food and will be visible to the mess admin.
MessAdmin: This is the landing page for the mess admins and has a navbar that takes the admin to its underwritten children pages.
Post Updates: This page is used by the Mess admin to post the mess menu and any updates related to the mess.
FeedBack: This page gives the mess admin the feedback given by the student for any given time food was served.

BACKEND :

Here we are going to write the API requests from controllers and map them through the routes to find their endpoints.
auth.js ⇒ /auth
hst_complaints.js ⇒ /hst
smesspage.js ⇒ /smess
supdates.js ⇒ /supd
whst_complaints.js ⇒ /whst
wupdates.js ⇒ /wupd

DATABASE :

The following are the schemas for our database:

```
hst_allocation
Emp_Id: { type: String, required:  true  },
   Sroom: { type: String, required: true },
   Sblock: { type: String, required: true },
   Empty_rooms: { type: String, required:  true  },
   Filled_rooms: { type: String, required:  true  },
```

```
hst_complaints
```

```
    Emp_Id: { type: String, required:  true  },
      Sroom: { type: String, required: true },
      Sblock: { type: String, required: true },
      Type: { type: String, required: true },
      Comments: { type: String, required: true },
      Complained_Date : {type : String , required : true},
      is_Solved: { type: Boolean},
```

```
hstdetailsSchema
    Emp_Id: { type: String, required:  true  },
      Staff_Name: { type: String, required: true },
      Batch: { type: String, required: true },
      Block : {type :String } ,
      Room : {type :String }
```

```
MessRemarksSchema
    Emp_Id: { type: String, required:  true  },
      Rating: { type:String, required: true },
      Remarks: { type: String, required: true },
      noon : {type:String ,required : true},
      Date_Of_Issue: { type: String, required: true },
```

```
MessSchema
    Emp_Id: { type: String, required:  true  },
      Date_Of_Issue: { type: String, required: true },
      Remarks: { type: String, required: true },
```

```
staffSchema
    Emp_Id: { type: String, required:  true  },
      Staff_Name: { type: String, required: true },
      Contact: { type: String, required: true },
      Batch: { type: String },
      Vaccination_Status: { type: Boolean},
      Email: { type: String, required: true },
      Type: { type: String, required: true },
      Password: { type: String, required: true },
      Reset_Token: { type: String, required: true },
      Reset_Expire: { type: Boolean, default:false ,required: true },
```

```
WardenUpdatesSchema
Emp_Id: { type: String, required:  true  },
    Comments: { type: String, required: true },
    Date_Of_Issue: { type: String, required: true },
```