**INTRODUCTION**

This is an Employee Management System. The main purpose to build this project is to manage Employees and their details. Also, there is a feature for Employees, by this they can see their Holiday List.

And all these programs are managed by the admins. They can add/remove Employees and also add/remove holidays in the Holiday List.

**PARTS OF THE PROJECT**

**BACK-END *(P-2 to P-4)***

**FRONT-END *(P-5 to P-6)***

**BACK-END**

**Technologies**

To make the Server side portion of our Project we used several technologies to make it happen. The main technology we are work in is Node.js and the must needed packages of it. We used Express.js, nodemon, cors, mongoose, mongodb, jsonwebtoken,bcrypt, helmet, morgan. And for storing our data we used MongoDB as our Database**.**

**Admin Actions**

• Manage other admins & employees:

o Add

o Remove

o View

o Update

**Back-end APIs**

• Employee Management:

Add Employee API,

Show all Employee API,

Update Employee API,

Delete Employee API

• Employees Authentication:

Login API

• Holiday Lists:

API for fetching all Holiday Lists

API for Add a new Holiday,

API for update a Holiday,

API for delete an existing Holiday

**INFORMATION we Need**

All information is handled by the Admin. There are several Information needed for Employees and Holidays, Such as,

**Employee**:

Employee ID,

Employee Name,

Employee Designation,

Employee Date of Joining

**Holidays:**

Holiday Type,

Holiday Name,

Holiday Date,

Holiday Day

**DATABASE MODELS**

The project will have the following Database Models

**Employee**:

id: => Integer

name:=>String

desigation:=>String

joined\_on:=> String

is\_admin:=>boolean

**Holiday List**:

type:=>String,

name:=>string,

date:=>date,

day:=>string

**HANDLE DATA IN BACK END**

**Phase – 1**

We need to make APIs for Employee Part and Holiday List.

In our case we need five APIs for achieve our Goal.

**First Task**, need a POST API. By this admin can add employees.

**Second Task**, need a GET API. By this admin can view all employee details.

**Third Task**, need a PUT API. By this admin can update any employee details.

**Fourth Task**, need a DELETE API. By this admin can remove any employee from DB.

**Fifth Task**, need a POST API. By this employee can login into portal.

**Phase - 2**

Once we get data from admins, now we can make APIs for Holiday List.

In our case we need four APIs for achieve our Goal.

**First Task**, need a POST API. By this admin can add Holidays.

**Second Task**, need a GET API. By this admin and employees can view all holiday details.

**Third Task**, need a PUT API. By this admin can update any holiday details.

**Fourth Task**, need a DELETE API. By this admin can remove any holiday list from DB.

**FRONT-END**

**TECHNOLOGIES**

To make the Client-side portion of our Project we are going to used several technologies to make it happen. The Client-side of this project is based on React.JS and obviously html and CSS. And also, we need library/ libraries of React.JS to make it of the mark, like react-router-dom.

**PAGES**

1. Login Component
2. Navbar
3. Dashboard Component
4. Add user page
5. Profile page
6. Update Profile page
7. Add Holiday page
8. View Holiday lists page
9. Update Holiday page
10. View all Users page

*\*\*Number of pages/components may be update later depending upon requirements.*

**APIs WE NEED**

* User Login API
* Get all users API
* Add user API
* Update single user API
* Get single user API
* Update Password API
* Remove user API
* Add Holiday API
* Get Holiday List API
* Update Holiday API
* Remove Holiday API

**PROCCESS & PLANS**

**To make our work easier we divide our work in two phases.**

Phase – 1

* Make Login page first and integrate require back-end API with it.
* After successfully logged in user should redirected to dashboard.
* Make a Navbar consists of User, Holiday and Profile tab and a logout button.
* If the user is admin he can have the access of user, profile and holiday tab and if our user is an employee then he will get access of holiday and profile tab in Navbar.
* In User tab (only access Admins) admin can show all users list. They can also add, update, delete uses if needed.
* In Holiday tab, holiday lists are available for both employees and Admins. But only Admins can modify them.
* In Profile tab Both Employees and Admins can see their Profile details.
* Employees and Admins can update their personal infos and password.
* While user clicked on Log out Button they will redirected to login Page.

**.**