



“Capstone Project – Attendance and Leave Management System ”

Submitted by: Bammidi Vihar, Mayur, Arshad

Submission type : Group 6

Name's: Bammidi Vihar, Mayur, Arshad

Batch : 12

LMS Id : viharbammidi@gmail.com

Program : Axis Fintech

Date: 13/07/2023

Git Hub Link:

https://github.com/Vihar1179/LAMS_1

Drive Link:

<https://drive.google.com/drive/folders/19H9sFQaz3HJv1cSFwC4uU7Ck-15pRXaN?usp=sharing>

Table of contents

1. **Introduction**
2. **Project Overview**
3. **Architecture Design**
4. **User Interface Design**
5. **ER – Diagram**
6. **Front-end Development**
7. **Testing**
8. **Integration with axios**
9. **Conclusion**
10. **Future Work**
11. **References**

What is in it?

- ❑ Attendance and leave management System is developed in order for the employees to mark their attendance and apply leaves on a regular bases
- ❑ There are three roles Hr, Employee, Manager
- ❑ HR is treated as Admin he can add employees and assign managers to the employees
- ❑ Traditional attendance and leave management systems rely on manual processes, such as paper-based timesheets and leave applications, leading to a significant waste of time and resources. Automating these processes eliminates the need for manual data entry and streamlines the overall workflow.
- ❑ Without a proper attendance and leave management system, employees may not have clear visibility into their leave balances, accruals, and overall attendance records. This can result in misunderstandings, disputes, and a lack of trust between employees and management.

Attendance & Leave Management Systems



Features:

- ☐ HR, Manager, Employee – 3 Login types
- ☐ **HR:(Admin)**
- ❖ Add employee through single sheet, Bulk
- ❖ Can see and edit profile and can see all employee and edit their work information as well as change status and reset leave balance
- ❖ Can add Departments
- ❖ Can see the status of Attendance and leave history
- ❖ can mark attendance for single and multiple days
- ❖ Will get mail for all the things he need to do
- ❖ Can apply for leave and check his leave balance
- ❖ Can add holidays, bulk add holidays can delete holidays and see the calendar
- ❖ Can see and upload roster in Calendar for all hybrid employees
- ❖ Can send, upload, delete and open the documents



Vihar

Personal Info.

ID: 1
Name: Vihar Bammidi
Gender: Male
Email: viharbammidi@gmail.com
Username: vihar123
Marriage Status: single
Country: India
Permanent Address: AP
Contact Number: 999999999
Emergency Contact Number: 88888888
Date of Birth: 2001-12-17

Work Information

Status: Active
Designation: HR
Job Title: Admin
Employee Type: Hybrid
Roster Type: WFH
Department: HR
Reporting Manager: Vihar Bammidi
Role: HR



Project Overview

❑ Manager

- ❖ Can see and edit profile
- ❖ Can apply for attendance and leave and see their balance leaves
- ❖ Can approve or reject attendance and leaves
- ❖ Can see the holidays posted by hr in calendar
- ❖ Can see their roster if Hybrid employee
- ❖ Can open and download documents by HR

❑ Employee

- ❖ Can see and edit profile
- ❖ Can apply for attendance and leave and see their balance leaves
- ❖ Can see their leaves and attendance history
- ❖ Can see their roster if Hybrid employee
- ❖ Can open and download documents by HR
- ❖ Can check holidays posted by HR

| Today | Back | Next | July 2023 | | | | Month | Week | Day | Agenda |
|-------|------|------|-----------|-----|-----|-----|-------|------|-----|--------|
| Sun | Mon | Tue | Wed | Thu | Fri | Sat | | | | |
| 25 | 26 | 27 | 28 | 29 | 30 | 01 | | | | |
| 02 | 03 | 04 | 05 | 06 | 07 | 08 | | | | |
| 09 | 10 | 11 | 12 | 13 | 14 | 15 | | | | |
| 16 | 17 | 18 | 19 | 20 | 21 | 22 | | | | |
| | WFO | WFO | WFH | WFH | WFH | | | | | |
| 23 | 24 | 25 | 26 | 27 | 28 | 29 | | | | |
| | WFO | WFO | WFH | WFH | | | | | | |
| 30 | 31 | 01 | 02 | 03 | 04 | 05 | | | | |

Leave Balance

Sick: 15

Leave Balance

Casual: 15

Leave Balance

Paternity: 10

Leave Balance

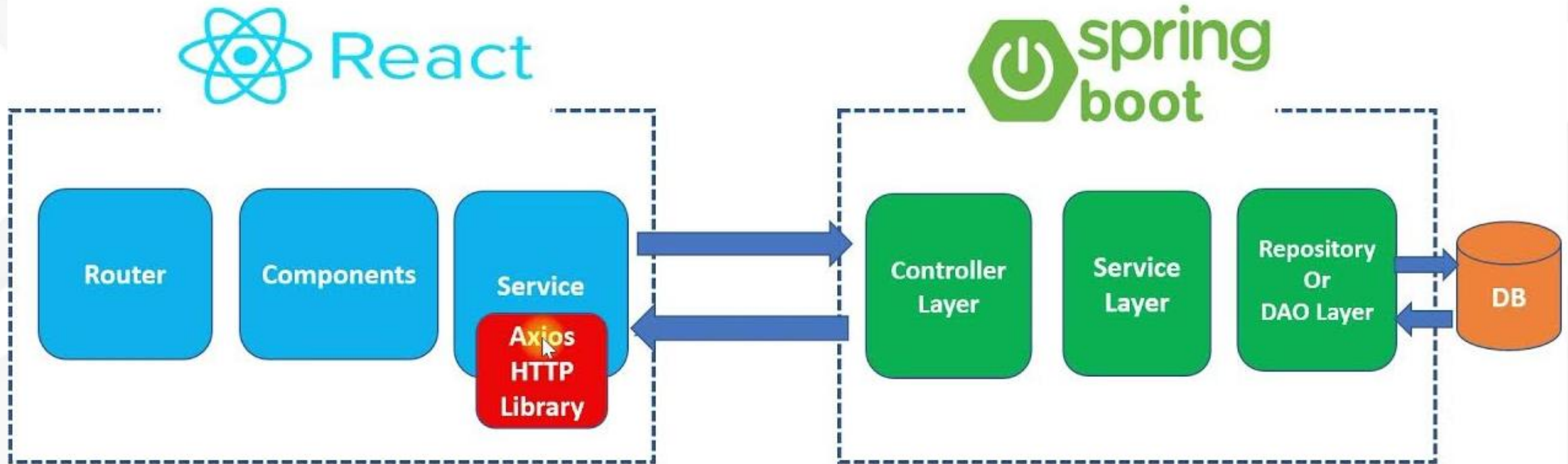
Marriage: 15

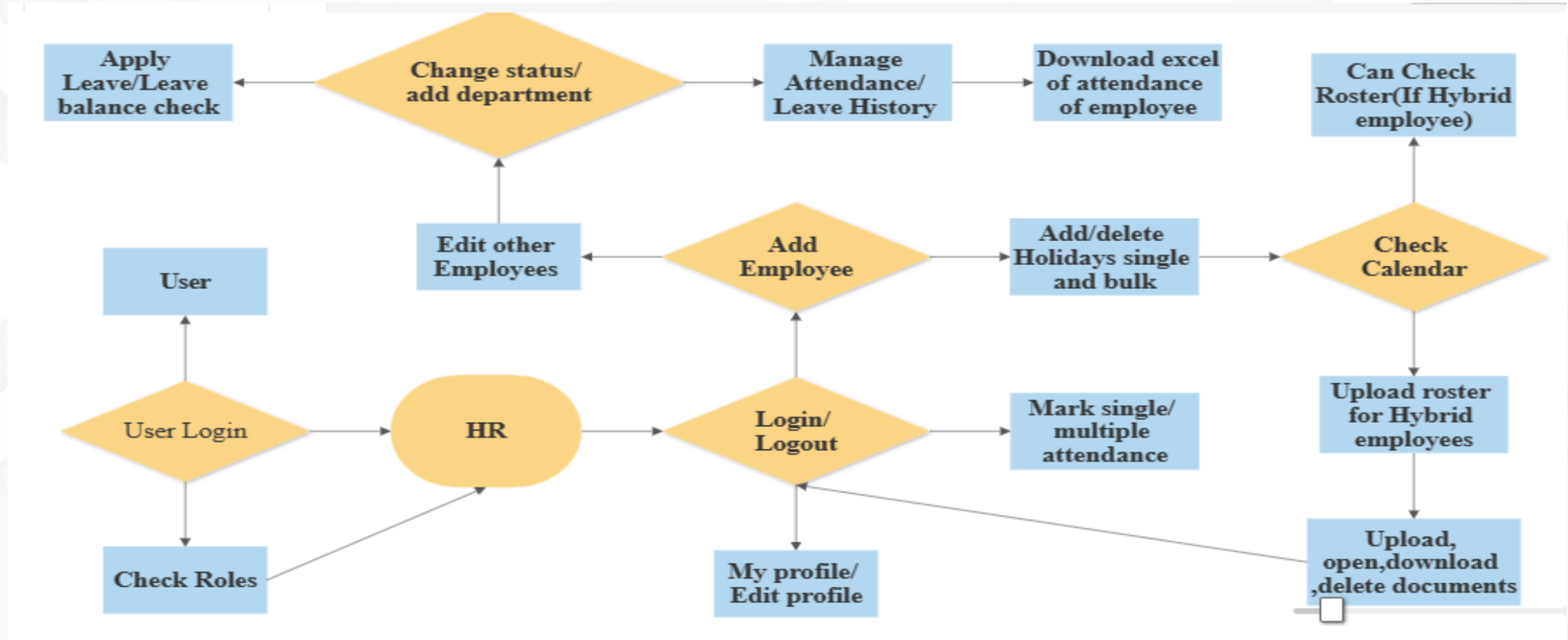
Technology Used

| Front end | Back end | Database |
|---------------------------|--------------------------|----------------------|
| Axios | Java | My SQL |
| React JS (Java Script) | Spring-boot | MySQL-connector-java |
| Chakra UI | Maven | |
| Components like Calendars | JWT, Mail-sender, Lombok | |
| | JPA, Hibernate | |
| | Bycrypt | |

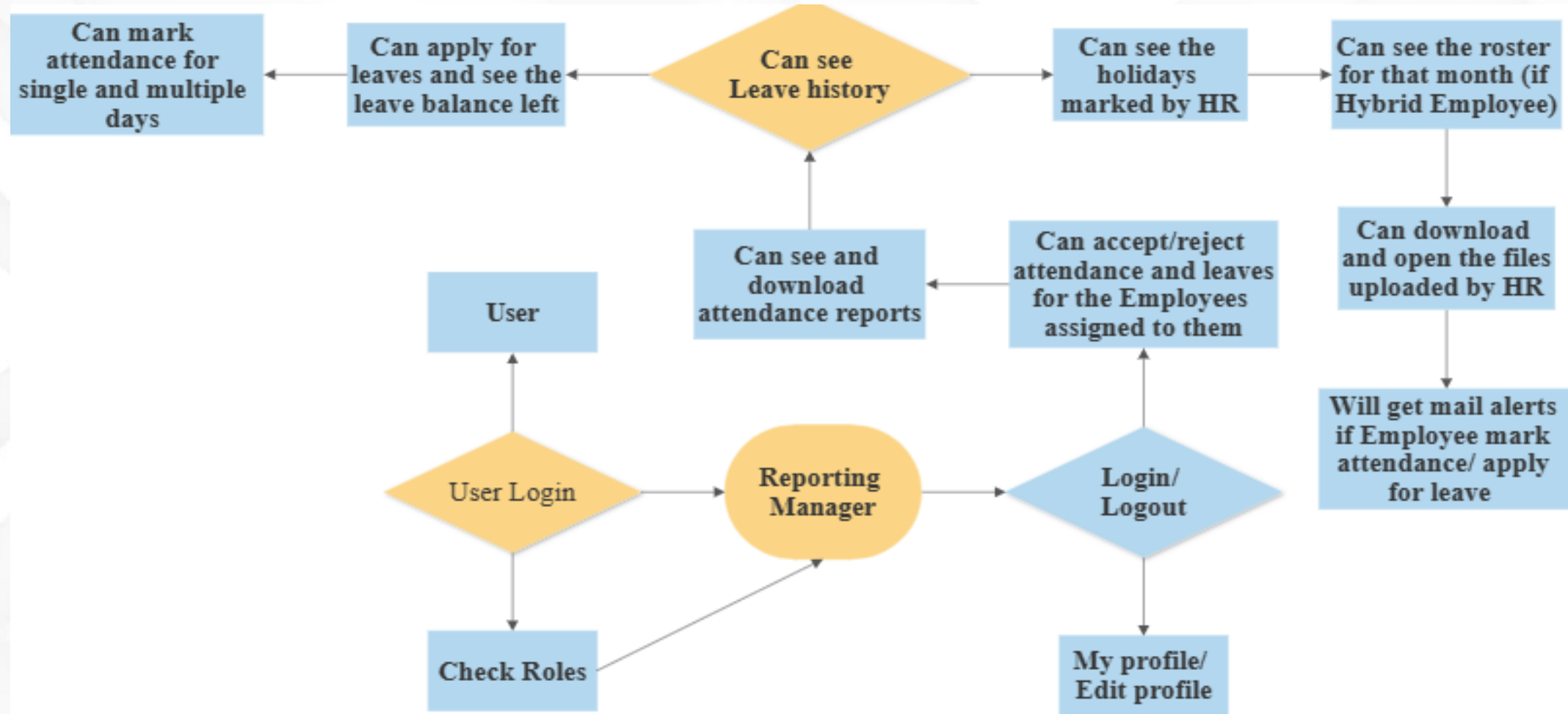
- ❑ React – big – calendar, XlXS – For excel, Date picker for blocking calendar, Chakra UI Icons for Icons
- ❑ Tools: Visual Studio, IntelliJ, My sql- command-line, Postman

Spring Boot + React Full Stack Application Architecture

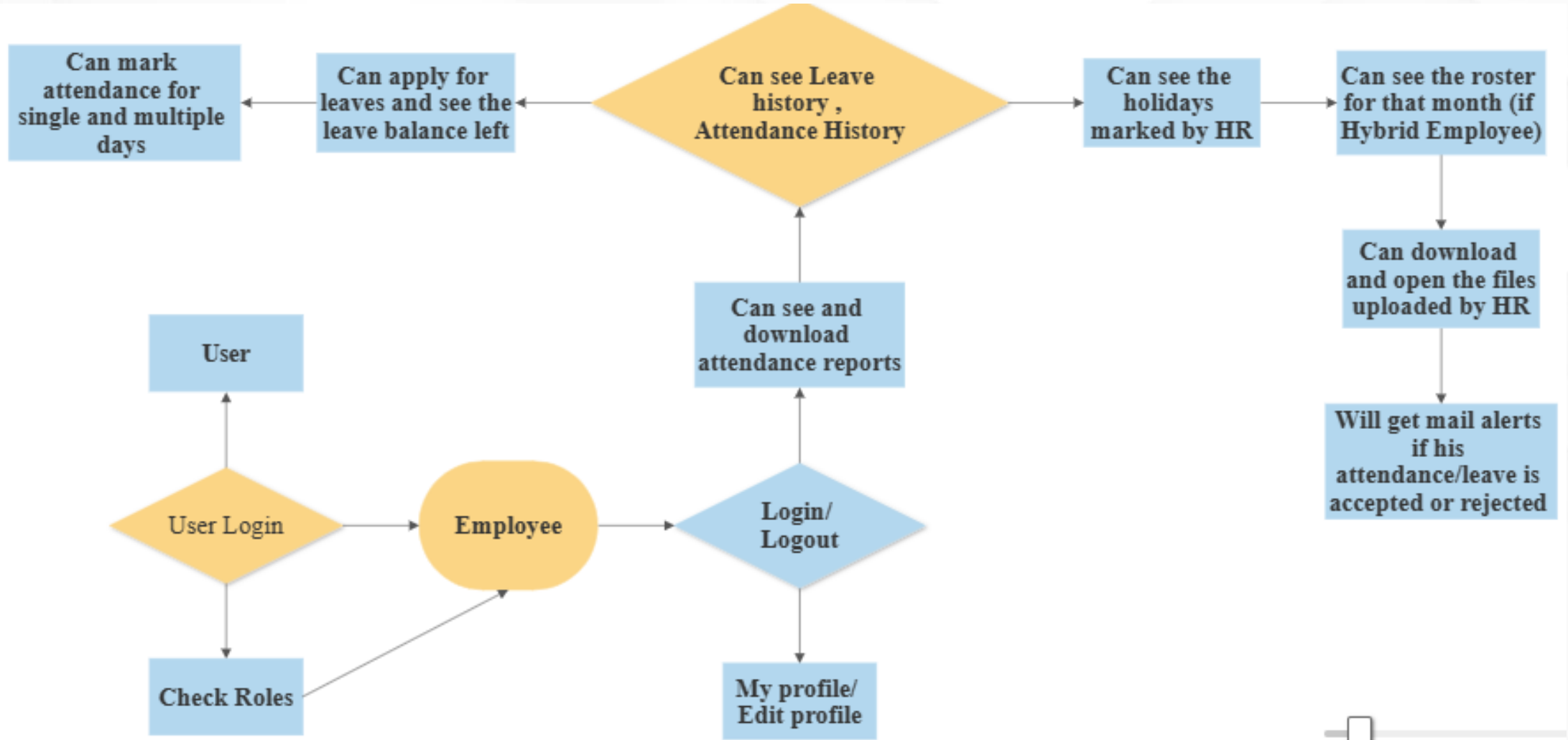




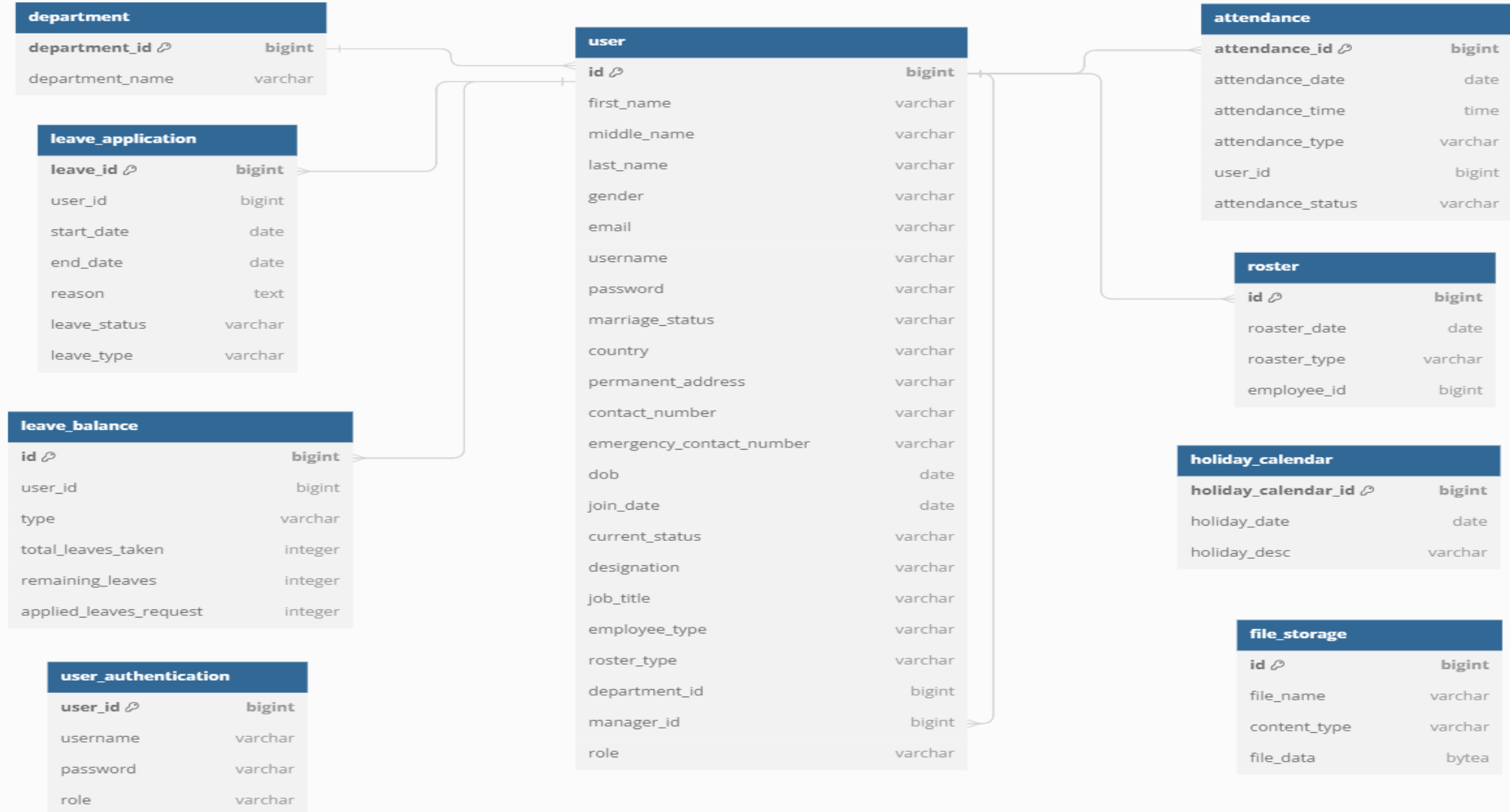
User Interface Design – Manager



User Interface Design - Employee



ER – Diagram



❑ Languages and Framework used:

- Chakra UI To make the UI page Interactive
- Java-script and React-JS used react-JS due to virtual Dom

❑ Front-end components

- Responsive Frontend With different resolution
- Added Search bar for Employee list, Attendance & Leave History to make it User friendly
- And for Everything you will be having a dropdown will registering to make it User friendly
- We kept Working people and resign people in separate tabs to make the responsive faster when we see the active people
- Used axios For connecting end points

Employee List

[Active Employees](#)[Inactive Employees](#)

Employee ID: 10



Name: alex aero

Email: vihar.bammidi17@gmail.com

Role: Manager

Department: R&D

Status: Active

Reporting Manger: Vihar Bammidi

[Leave Requests](#)[Leave History](#)

Pending Leave Requests

Name: Vihar Bammidi



User ID: 1

Start Date: 2023-07-10

End Date: 2023-07-11

Reason: dsx

Status: PENDING

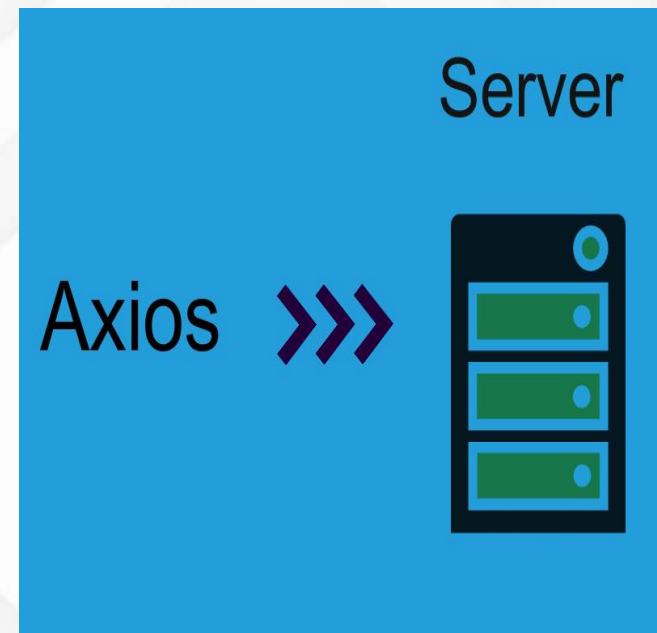
Leave Type: Sick

❑ Postman:

- We used Postman for testing the endpoints before we used to connect it to frontend

❑ Integration

- **Token and Role retrival:** In the getConfig function, the token is retrieved from the sessionStorage using sessionStorage.getItem("token"). Similarly, the role is retrieved from sessionStorage using sessionStorage.getItem("role"). These values are then used to construct the authorization header for subsequent API requests
- **Login Functionality:** In the submitHandler function, the login function is called with the usernameOrEmail and password as parameters. This function is responsible for making an HTTP request to the login endpoint of the backend API using Axios. If the login is successful, the returned accessToken and role are stored in sessionStorage using sessionStorage.setItem.
- **API Request:** In the register function, Axios is used to make an HTTP POST request to the /user/register endpoint of the backend API. The userData is sent as the request payload, and the getConfig function is used to include the authorization header with the request. The response is then returned.



❑ Integration:

- **Session Storage Usage:** After a successful login, the accessToken and role are stored in sessionStorage using sessionStorage.setItem. These values are later retrieved using sessionStorage.getItem when making subsequent API requests. This allows the frontend to send the authorization token along with the requests to the backend API.

```
// Store user info in local storage
sessionStorage.setItem("token", response.accessToken);
sessionStorage.setItem("login", true);
sessionStorage.setItem("role", response.role);

// Debugging statements
console.log("Token:", sessionStorage.getItem("token"));
console.log("Role:", sessionStorage.getItem("role"));
```

❑ Register Function

❑ Storing in session storage

```
const register = async (userData) => {
  try {
    const response = await axios.post(
      `${API_BASE_URL}/user/register`,
      userData,
      getConfig()
    );
    return response.data;
  } catch (error) {
    throw error.response.data;
  }
};
```

```
const getConfig = () => {
  const token = sessionStorage.getItem("token");
  //console.log("Retrieved token:", token); // L

  const config = {
    headers: {
      Authorization: "Bearer " + token,
    },
  };
  //console.log("Request headers:", config.headers);

  return config;
};
```

❑ Get config

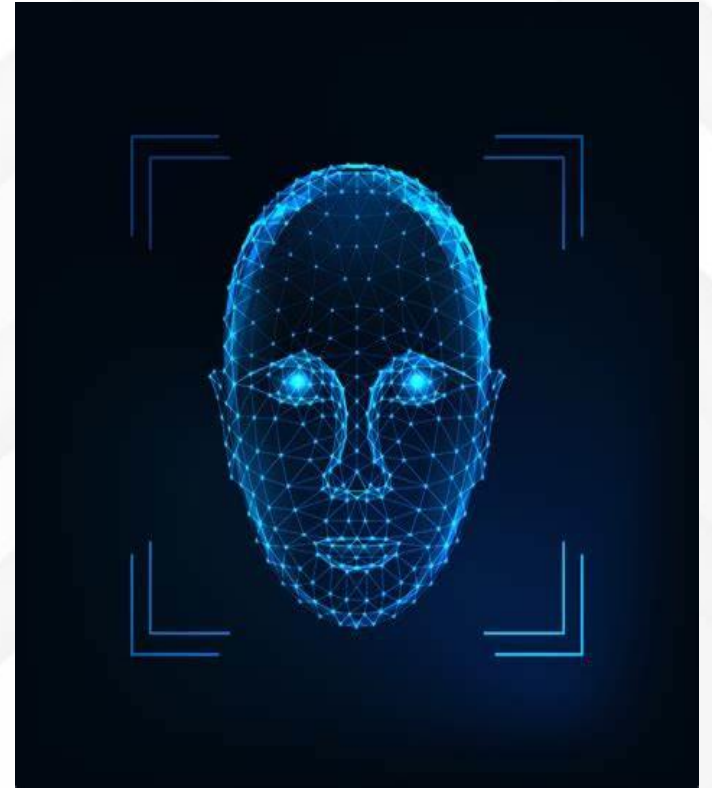
Conclusion:

- ❑ The implementation of an Attendance and Leave Management System is crucial for modern organizations to streamline their workforce management processes and optimize operational efficiency. By leveraging advanced technology, such a system allows for accurate tracking of employee attendance, simplifies leave requests and approvals, and enhances overall productivity.
- ❑ Through the course of this presentation, we have explored the key features and benefits of an Attendance and Leave Management System. We have seen how it automates timekeeping, eliminates manual errors, provides real-time insights into attendance patterns, and simplifies the leave management process. These advantages not only save time and effort for HR personnel but also empower employees to manage their attendance and leave effectively.



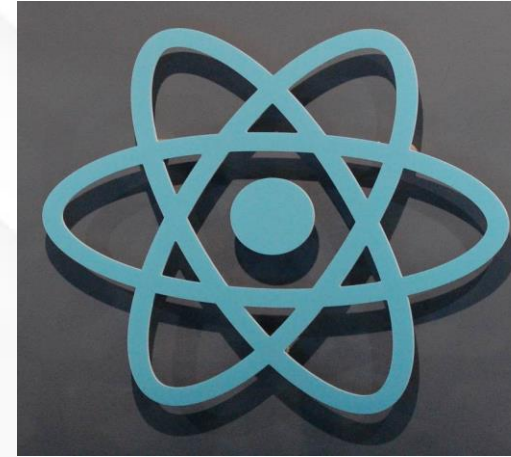
❑ Future Scope:

- **Integration with Time and Attendance Devices:** Explore the possibility of integrating the portal with advanced time and attendance devices such as facial recognition systems or fingerprint scanners to provide more secure and accurate attendance tracking.
- **Automated Workflow Management:** Implement an automated workflow system for leave requests and approvals, streamlining the process and reducing administrative overhead.
- **Employee Performance Tracking:** Integrate performance management features into the portal, allowing managers to track employee performance metrics alongside attendance records for a holistic view of employee productivity.
- **Integration with Collaboration Tools:** Explore integration with collaboration tools such as project management software, instant messaging platforms, or video conferencing tools to create a seamless work environment and enhance team collaboration.



❑ ReactJS – References

- React documentation : <https://reactjs.org/docs>
- React Tutorial : <https://reactjs.org/tutorial/tutorial.html>
- React Router: <https://reactrouter.com/web/guides/quick-start>
- React Hooks: <https://reactjs.org/docs/hooks-intro.html>



❑ Spring-boot – References

- Spring Boot Documentation: <https://docs.spring.io/spring-boot/docs/current/reference/html/>
- Spring Boot Tutorials by Baeldung: <https://www.baeldung.com/spring-boot>

❑ My SQL – References

- MySQL Documentation: <https://dev.mysql.com/doc/>
- MySQL Tutorial by Guru99: <https://www.guru99.com/mysql-tutorial.html>



Thank you

