A REPORT SUBMITTED AS A PART OF EXPERIENTIAL LEARNING ON SEAM

TOPIC: Employee Management Systems

Under the Supervision of Professor
Mrs. Chandan Kumar Panda

Submitted By

GROUP-10

NAME	REGD. NO
AAKASH BURNWAL	2101020279
SHREAYA KUMARI	2101020282
SAURAV KUMAR	2101020305
RANA SUJIT KUMAR	2101020306
MAHENDRA SINGH	2101020451

DECLARATION

This is to certify that the case study and title dash submitted by the team number 10 for partial fulfilment of degree of B Tech under CV Raman Global University under the supervision of Chandrakumar Panda sir Sir from cse dept dash The further declare we we further declare this work is not submitted to any other university for any purpose

Warm regards,

Signature

AAKASH BURNWAL

SHREAYA KUMARI

SAURAV KUMAR

RANA SUJEET KUMAR

MAHENDRA SINGH

Signature of HOD (CSE)

Signature of supervisor

ACKNOWLEDGEMENT

We express our sincere gratitude to all who contributed to the development and implementation of the **Employee Management System**. First and foremost, we extend our appreciation to our dedicated team whose relentless efforts made this project a reality. Their expertise and commitment were instrumental in every phase of the system's creation, from conceptualization to execution.

We are also thankful to our management for their unwavering support and guidance throughout this journey. Their vision and encouragement propelled us forward, ensuring the project's success.

Furthermore, we acknowledge the invaluable feedback and suggestions provided by our end-users, whose insights helped refine the system to better meet their needs.

Lastly, we extend our thanks to all stakeholders and partners involved, whose collaboration and cooperation facilitated a smooth development process. Together, we have created a robust Employee Management System that will streamline operations and enhance efficiency across our organization.

CONTENT

INTRODUCTION

The Employee Management System (EMS) stands as a testament to the ever-evolving landscape of organizational efficiency and workforce optimization. In a modern era characterized by dynamic workplaces and shifting paradigms, the need for streamlined and user-friendly solutions to manage human resources has never been more crucial. With its comprehensive suite of features and intuitive interface, the EMS emerges as a beacon of innovation, offering organizations a powerful tool to navigate the complexities of employee management with ease and precision.

At its core, the EMS is engineered to revolutionize the way organizations approach the management of their most valuable asset: their people. Gone are the days of cumbersome spreadsheets and disjointed systems. The EMS centralizes all employee-related functions into a single, cohesive platform, providing a holistic solution that empowers HR professionals and managers to efficiently oversee every aspect of the employee lifecycle.

From the moment a new hire steps through the door, the EMS facilitates a seamless onboarding process that sets the stage for success. Gone are the manual paperwork and administrative headaches that often accompany traditional onboarding methods. With the EMS, new employees can complete essential forms, review company policies, and access training materials—all within a digital environment that is both user-friendly and secure.

Once onboarded, employees can leverage the EMS to clock in and out, track their attendance, and manage their schedules with unparalleled ease. Whether they're working in the office or remotely, the EMS ensures that timekeeping is accurate and transparent, providing both employees and managers with real-time insights into productivity and attendance patterns.
But the EMS is not just about tracking hours; it's about empowering employees to thrive and grow within the organization. Through integrated performance evaluation tools, managers can set goals, provide feedback, and conduct performance reviews in a structured and transparent manner. By aligning individual performance with organizational objectives, the EMS fosters a culture of accountability and continuous improvement, driving employee engagement and retention.

Of course, no discussion of employee management would be complete without addressing the intricacies of payroll administration. Here again, the EMS shines, offering a robust suite of payroll management features that streamline the entire process from start to finish. From calculating wages and deductions to generating tax forms and direct deposit payments, the EMS automates the most time-consuming aspects of payroll processing, freeing HR professionals to focus on more strategic initiatives.

Beyond its core functionalities, the EMS is continuously evolving to meet the ever-changing needs of modern organizations. With regular updates and enhancements based on user feedback and industry best practices, the EMS remains at the forefront of innovation, driving efficiency, and productivity for businesses of all sizes and industries.

Key Features of Employee Management Systems

Employee Management Systems offer a wide array of features designed to meet the diverse needs of modern organizations. Some of the key features include:

Payroll Processing: Automated calculation and disbursement of employee salaries, taxes, and benefits.
Time and Attendance Tracking: Monitoring employee attendance, leave, and working hours to ensure accuracy and compliance.

Employee Self-Service Portals: Empowering employees to access and manage their personal information, submit leave requests, and view pay stubs.

Performance Evaluation and Management: Conducting performance reviews, setting goals, and tracking employee progress to facilitate professional growth and development.

Benefits Administration: Administering employee benefits such as healthcare, retirement plans, and insurance.

Compliance Management: Ensuring adherence to labor laws, regulations, and company policies.
Reporting and Analytics Tools: Generating insights and reports on various HR metrics to support data-driven decision-making.

Increased Efficiency: Automation of routine HR tasks reduces manual effort and minimizes errors, leading to greater operational efficiency.

Cost Savings: By streamlining processes and eliminating paper-based workflows, EMS help reduce administrative costs and improve resource utilization.

METHODOLOGY

The development and implementation of an Employee Management System (EMS) involve a systematic approach to ensure its effectiveness and reliability. The methodology typically encompasses several key stages:

Analysis: This phase involves gathering requirements from stakeholders, understanding existing processes, and identifying challenges and opportunities for improvement.

Design: Based on the analysis, the system architecture, database structure, user interface, and functionalities are designed, keeping usability and scalability in mind.

Development: The actual coding and programming of the EMS take place in this stage, adhering to industry standards and best practices.

Testing: Rigorous testing is conducted to validate the system's functionality, performance, security, and usability, identifying and resolving any issues or bugs.

Deployment: The EMS is deployed into the production environment, ensuring smooth transition and minimal disruption to operations.

Training and Support: Training sessions are conducted to familiarize users with the EMS, and ongoing support is provided to address any issues and ensure optimal utilization.

CASE STUDY FINDING

1.SYSTEM REQUIREMENT SPECIFICATION

An Employee Management System (EMS) is a comprehensive software solution designed to streamline various aspects of workforce management within an organization. At its core, the EMS serves as a centralized platform for handling employee-related tasks, data, and processes efficiently.

The key components of an EMS typically include:

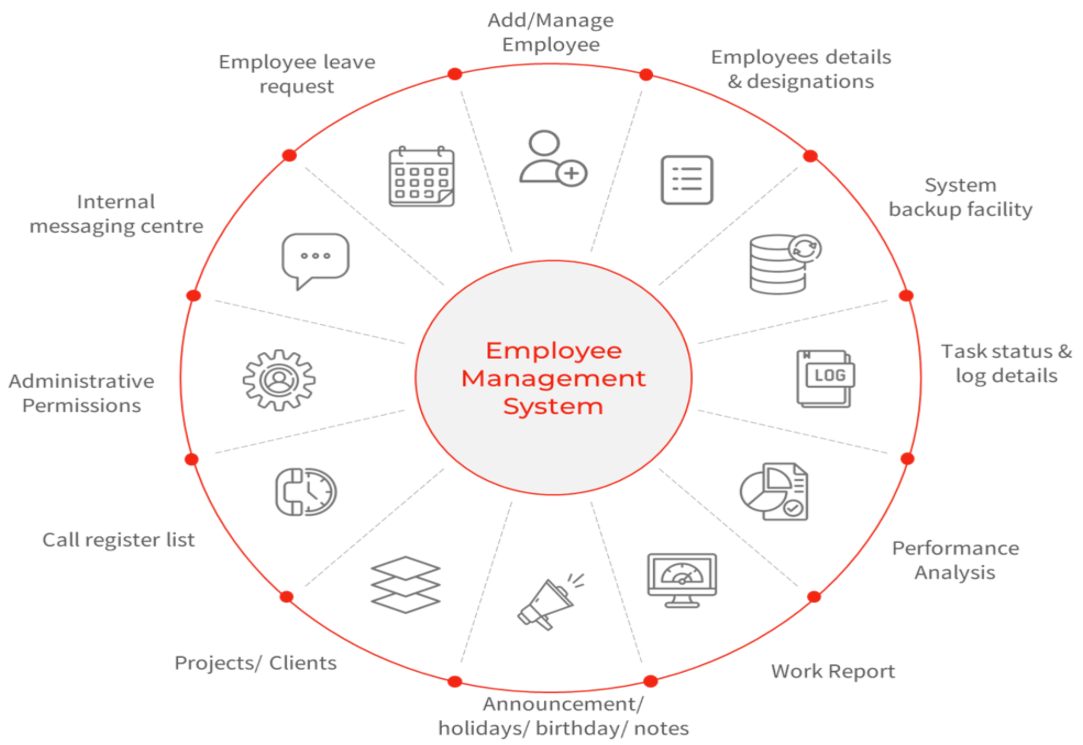
Employee Information Management: The system stores and manages essential employee information such as personal details, contact information, employment history, and qualifications in a secure and organized manner.
Attendance Tracking: EMS enables accurate tracking of employee attendance, including clock-in/out times, leave requests, and absence patterns. This feature helps in monitoring workforce availability and ensuring compliance with attendance policies.

Payroll Management: The system automates payroll processes by calculating salaries, deductions, taxes, and other financial aspects accurately. It ensures timely and error-free payroll processing while adhering to legal regulations and company policies.

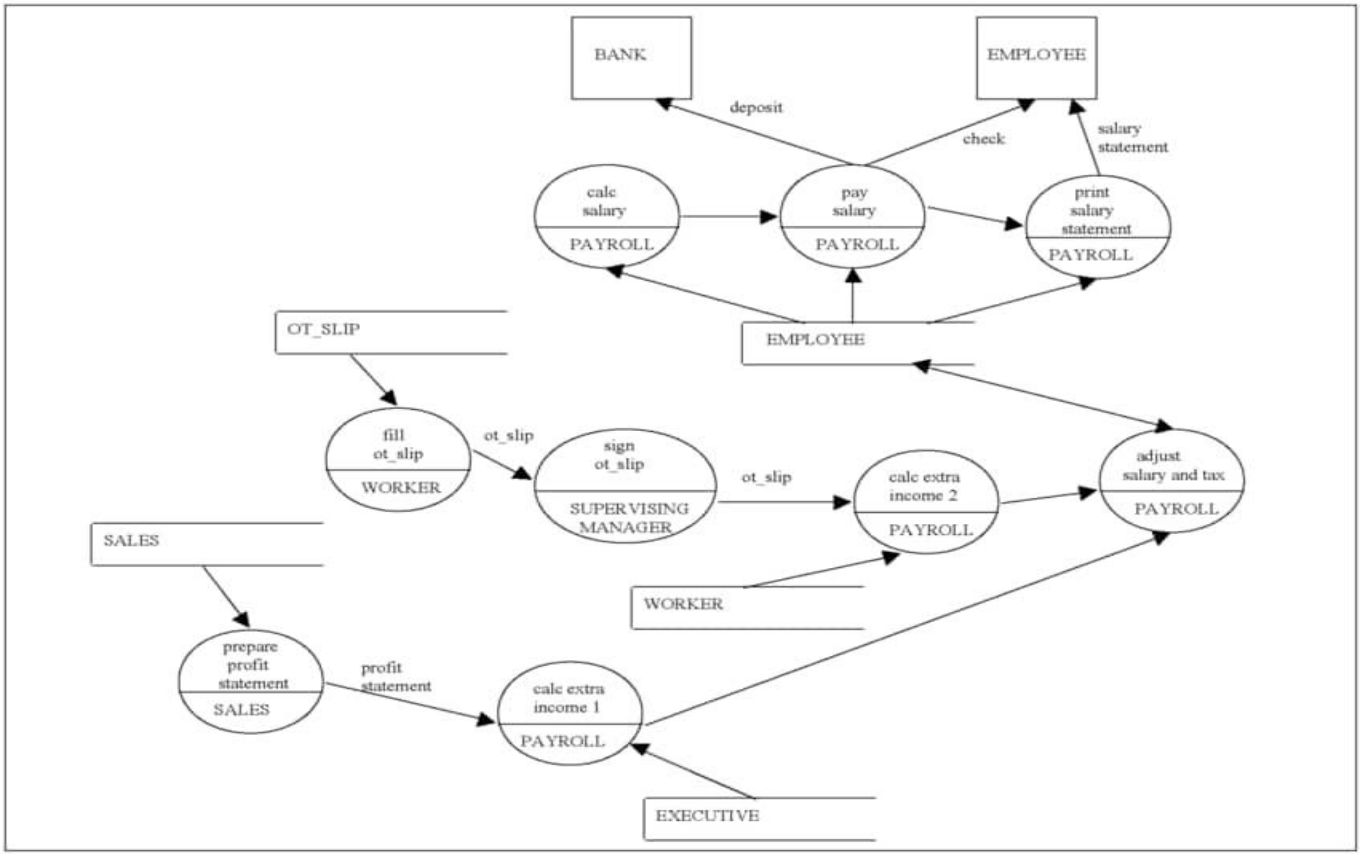
Performance Evaluation: EMS facilitates performance appraisal processes by providing tools for setting goals, conducting evaluations, and providing feedback to employees. It helps in assessing employee performance, identifying areas for improvement, and rewarding achievements
.
Training and Development: The system manages employee training programs, schedules, and certifications, allowing organizations to track employees' skill development and ensure compliance with training requirements.

Communication and Collaboration: EMS often includes features for internal communication, such as messaging, announcements, and document sharing, fostering collaboration and enhancing transparency across the organization.

Analytics and Reporting: The system generates comprehensive reports and analytics on various HR metrics, including employee turnover, attendance trends, performance ratings, and training effectiveness. These insights enable data-driven decision-making and strategic workforce planning.


Employee Management System
Add/Manage Employee
Employees details & designations
System backup facility
Task status & log details
Performance Analysis
Work Report
Announcement/ holidays/ birthday/ notes
Projects/ Clients
Call register list
Administrative Permissions
Internal messaging centre
Employee leave request
LOG

2.DATA FLOW DIAGRAM



3.STRUCTURED ANALYSIS

Structured Analysis involves breaking down the Hotel Automation Software systematically using tools like Data Flow Diagrams (DFD) and a Data Dictionary. These tools visually represent data flow, elements, and relationships, providing a concise understanding of the system's data processes. This section lays the foundation for a detailed exploration of the software's design and functionality.

The structured design of an Employee Management System (EMS) involves a systematic approach to ensure the system's functionality, scalability, and maintainability. Here's an overview of the structured design process for an EMS:

Requirements Analysis: The first step is to gather and analyze requirements from stakeholders, including HR personnel, managers, and end-users. This involves identifying the key functionalities, data requirements, user roles, and system constraints.

System Architecture Design: Based on the requirements analysis, the system architecture is designed, outlining the high-level structure, components, and interactions. This may include defining modules for employee information management, attendance tracking, payroll processing, etc.

Database Design: A relational database schema is designed to store and manage employee data efficiently. This involves defining tables, relationships, constraints, and indexing strategies to ensure data integrity and optimize query performance.

User Interface Design: The user interface (UI) design focuses on creating intuitive and user-friendly interfaces for interacting with the EMS. This includes designing forms, screens, menus, and navigation flows to facilitate smooth user interactions and enhance usability.

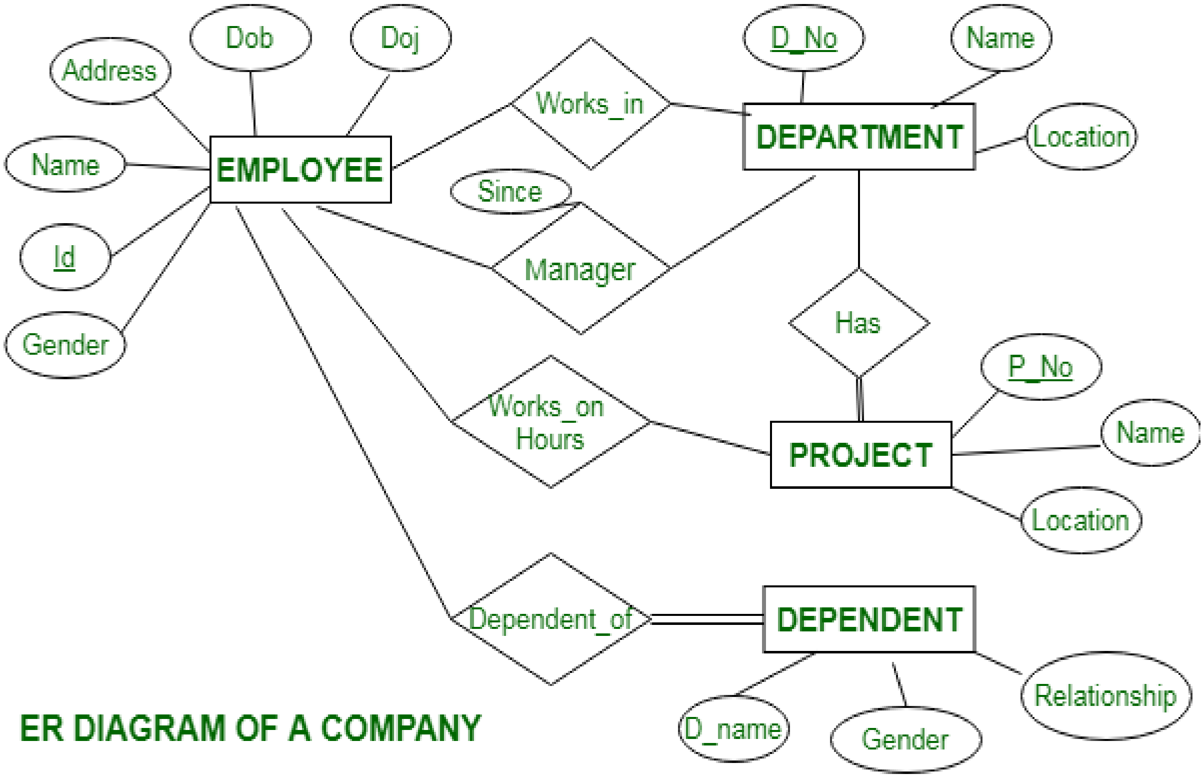
Functional Design: Detailed specifications are developed for each functional module of the EMS, outlining the input/output requirements, processing logic, error handling, and validation rules. This ensures that each module performs its intended tasks accurately and reliably.

Security Design: Security measures are integrated into the design to protect sensitive employee data and prevent unauthorized access. This may include user authentication, role-based access control, encryption, and audit trails to ensure compliance with data protection regulations.

Scalability and Performance Design: The design incorporates strategies for scalability and performance optimization to accommodate growing user bases and workload demands. This may involve distributed architecture, caching mechanisms, and load balancing techniques to ensure optimal system performance.

Error Handling and Recovery Design: Robust error handling mechanisms are designed to detect and handle exceptions gracefully, ensuring system stability and data integrity. This includes logging, error notifications, and recovery procedures to minimize downtime and data loss.

4.ER MODEL



Implementation

When implementing an Employee Management System, organizations should consider several factors to ensure successful deployment and adoption:

Scalability: The EMS should be capable of accommodating the organization's current needs and future growth. Integration Capabilities: Seamless integration with existing systems, such as accounting software and HRIS (Human Resource Information System), is essential to avoid data silos and streamline processes.

Customization Options: The ability to customize the EMS to align with the organization's unique workflows and requirements enhances usability and effectiveness.

User-Friendliness: Intuitive user interfaces and comprehensive training programs are crucial for ensuring smooth adoption and acceptance by employees.

Security Measures: Robust data security measures, including encryption, access controls, and regular audits, are necessary to safeguard sensitive employee information and ensure compliance with data protection regulations.

SOURCE CODE:

Frontend(html):

```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <link rel="icon" href="%PUBLIC_URL%/favicon.ico" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <meta name="theme-color" content="#000000" />
    <meta
      name="description"
      content="Web site created using create-react-app"
    />
    <link rel="apple-touch-icon" href="%PUBLIC_URL%/logo192.png" />

    <link rel="manifest" href="%PUBLIC_URL%/manifest.json" />
    <title>Employee Management System</title>
  </head>
  <body>
    <noscript>You need to enable JavaScript to run this app.</noscript>
    <div id="root"></div>
  </body>
</html>
```

(CSS Page):

```css
.card {
    box-shadow: rgba(17, 17, 26, 0.1) 0px 0px 20px, rgba(17, 17, 26, 0.05) 0px
10px 32px;
    border-radius: 1.1rem;
    border: none;
}

.card-title {
    text-align: left !important;
}

.card-title h6{
    color: var(--secondary-font);
    font-size: 1rem !important;
}
```

```css
.leave-section h3 {
    color: var(--main-header);
    font-weight: 700;
}

.leave-section {
    height: 12rem;
    overflow-x: auto;
}

.leave-section::-webkit-scrollbar {
    padding-left: 0.1rem;
    width: 10px;
    height: 10px;
}

.leave-section::-webkit-scrollbar-thumb {
    background: linear-gradient(to bottom right, #ff4c52 0%, rgb(255, 139, 139)
100%);
    border-radius: 5px;
}

.leave-section::-webkit-scrollbar-track {
    background-color: #ddddddbe;
    border: 1px solid #ccc;
    border-radius: 1rem;
}

.leave-req {
    color: var(--main-header);
    font-weight: 600;
}

.leave-text {
    font-size: 1.15em;
    font-weight: 600;
    /* color: var(--secondary-font); */
}

.sticky-head {
    position: sticky;
    position: -webkit-sticky;
    top: 0;
    background-color: #fff !important;
    z-index: 5;
}

.welcome-section {
    height: 12rem;
}

.welcome-section h2 {
    color: var(--main-header);
    font-weight: 700;
}

.welcome-section h6 {
    font-style: italic;
    color: var(--secondary-font);
}

.dashboard-img{
    margin-top: -4.5rem;
    margin-left: 2rem;
    height: auto;
    width: 20rem;
}

@media screen and (max-width: 576px) {
    .dashboard-img{
        margin-top: 0rem;
        width: 12rem;
    }
    .welcome-section {
        height: 15rem;
    }
}
```

Backend(Using React):

Home Page:

```
import React from "react";
import img from "../images/dashboard-img.png";
import "./WelcomeUI.css";

const WelcomeUI = ({ employee }) => {
  const getGreetings = () => {
    let timeNow = new Date().getHours();
    let greeting =
      timeNow >= 5 && timeNow < 12
        ? "Good Morning"
        : timeNow >= 12 && timeNow < 18
        ? "Good Afternoon"
        : "Good Evening";
    return greeting;
  };

  return (
    <div className="row welcome-section">
      <div className="col-md-7">
        {employee && (
          <h3>
            Hello, <span className="fw-bold">{employee.name}</span>
          </h3>
        )}
        <h2>{getGreetings()}!</h2>
        <h6>Have a Good Day !</h6>
      </div>
      <div className="col-md-5">
        <img src={img} className="dashboard-img" alt="dashboard"/>
      </div>
    </div>
  );
};

export default WelcomeUI;
```

Login Employee Page:

```
import React, { useContext } from "react";
import userContext from "../../context/userContext";
import { useForm } from "react-hook-form";
import { Form, Button } from "react-bootstrap";
import { useNavigate } from "react-router-dom";
import { message } from "antd";
import axios from "axios";

const FormGroup = (value) => {
  return (
    <Form.Group className="mb-3" controlId={value.id}>
      <>
        <Form.Label className="fw-bold">{value.label}</Form.Label>
        <Form.Control
          type={value.type}
          placeholder={value.placeholder}
          {...value.register(value.id, {
            required: {
              value: value.required,
              message: "This field is required",
            },
            pattern: {
              value: value.pattern,
              message: value.message,
            },
          })}
        />
      </>
      {value.errors?.message && (
        <Form.Text className="text-danger">{value.errors?.message}</Form.Text>
      )}
    </Form.Group>
  );
};

const LoginUser = () => {
  const authUser = useContext(userContext);
  const navigate = useNavigate();
  const {
    register,
    handleSubmit,
    formState: { errors },
  } = useForm();

  const onSubmit = async (data) => {
    try {
      const response = await axios.post("https://employee-management-system-
ujnj.onrender.com/api/superuser/login", {
        email: data.email,
        password: data.password,
      });
      const { _id, isSuperUser } = response.data.user;
      console.log(response.data.message);
      message.success("Login Successfull!!");
      authUser.login(response.data.token, _id, isSuperUser);
      navigate("/");
    } catch (error) {
      console.log(error);
      message.warning("Invalid Email or Password");
    }
  };
  return (
    <div className="mt-4">
      <h2 className="text-center profile-detail-heading">Log In User</h2>
      <div className="d-flex justify-content-center my-4">
        <Form onSubmit={handleSubmit(onSubmit)} className="px-5 py-4">
          <FormGroup
            register={register}
            errors={errors.email}
            type="email"
            placeholder="abc@gmail.com"
            label="Email Address"
            id="email"
            required={true}
            pattern={/^[\w-.]+@([\w-]+.)+[\w-]{2,4}$/}
            message="please enter valid email"
          />
          <FormGroup
            register={register}
            errors={errors.password}
            type="password"
            placeholder="password"
            label="Password"
            id="password"
            required={true}
            pattern={/^([a-zA-Z0-9@*#$%^&*!]{6,15})$/}
            message="password should contain atleast 8 characters"
          />
          <Button
            variant=""
            type="submit"
            className="my-2 custom-button w-100 p-2 fw-bold"
          >
            Submit
          </Button>
        </Form>
      </div>
    </div>
  );
};

export default LoginUser;
```

Registration Page:

```
import React, { useContext } from "react";
import { useForm } from "react-hook-form";
import { Form, Button } from "react-bootstrap";
import { message } from "antd";
import axios from "axios";
import userContext from "../../context/userContext";

const FormGroup = (value) => {
  return (
    <Form.Group className="mb-3" controlId={value.id}>
      {value.id === "superuser" ? (
        <Form.Check
          type={value.type}
          label="Admin rights?"
          {...value.register(value.id)}
        />
      ) : (
        <>
          <Form.Label className="fw-bold">{value.label}</Form.Label>
          <Form.Control
            type={value.type}
            placeholder={value.placeholder}
            {...value.register(value.id, {
              required: {
                value: value.required,
                message: "This field is required",
              },
              pattern: {
                value: value.pattern,
                message: value.message,
              },
            })}
          />
        </>
      )}
      {value.errors?.message && (
        <Form.Text className="text-danger">{value.errors?.message}</Form.Text>
      )}
    </Form.Group>
  );
};

const NewUser = () => {
  const {
    register,
    handleSubmit,
    formState: { errors },
  } = useForm();

  const authUser = useContext(userContext)

  const onSubmit = async (data) => {
    console.log(data);
    try {
      const formData = {
        email: data.email,
        password: data.password,
        joiningDate: data.joinDate,
        position: data.position,
        name: data.name,
        aadhar: data.aadharNo,
        panNo: data.panNo,
        isSuperUser: data.superuser,
        address: data.address,
        dateOfBirth: data.dateOfBirth,
        githubId: data.githubId,
        linkedIn: data.linkedIn,
        phone: data.tel,
      };
      const response = await axios.post("https://employee-management-system-
ujnj.onrender.com/api/superuser/signup", formData, {
        headers: {
          Authorization: "Bearer " + authUser.token,
        },
      });
      console.log(response);
      message.success(response.data.message);
    } catch (error) {
      console.log(error);
      message.error(error);
    }
  };

  return (
    <div className="mt-4">
      <h2 className="text-center profile-detail-heading">Add new employee</h2>
      <div className="d-flex justify-content-center my-4">
        <Form onSubmit={handleSubmit(onSubmit)} className="px-5 py-4">
          <div className="row">
            <div className="col-6">
              <FormGroup
                register={register}
                errors={errors.name}
                type="text"
                placeholder="enter your name"
                label="Full Name"
                id="name"
                required={true}
                pattern=""
              />
              <FormGroup
  register={register}
  errors={errors.email}
  type="email"
  placeholder="abc@gmail.com"
  label="Email Address"
  id="email"
  required={true}
  pattern={/^[\w-.]+@([\w-]+.)+[\w-]{2,4}$/}
  message="please enter valid email"
/>
<FormGroup
  register={register}
  errors={errors.password}
  type="password"
  placeholder="password"
  label="Password"
  id="password"
  required={true}
  pattern={/^([a-zA-Z0-9@*#$%^&*!]{6,15})$/}
  message="password should contain atleast 8 characters"
/>
<FormGroup
  register={register}
  errors={errors.joinDate}
  type="date"
  placeholder="Joining Date"
  label="Joining Date"
  id="joinDate"
  required={true}
/>
<FormGroup
  register={register}
  errors={errors.position}
  type="text"
  placeholder="Position"
  label="Position"
  id="position"
  required={true}
/>
<FormGroup
  register={register}
```

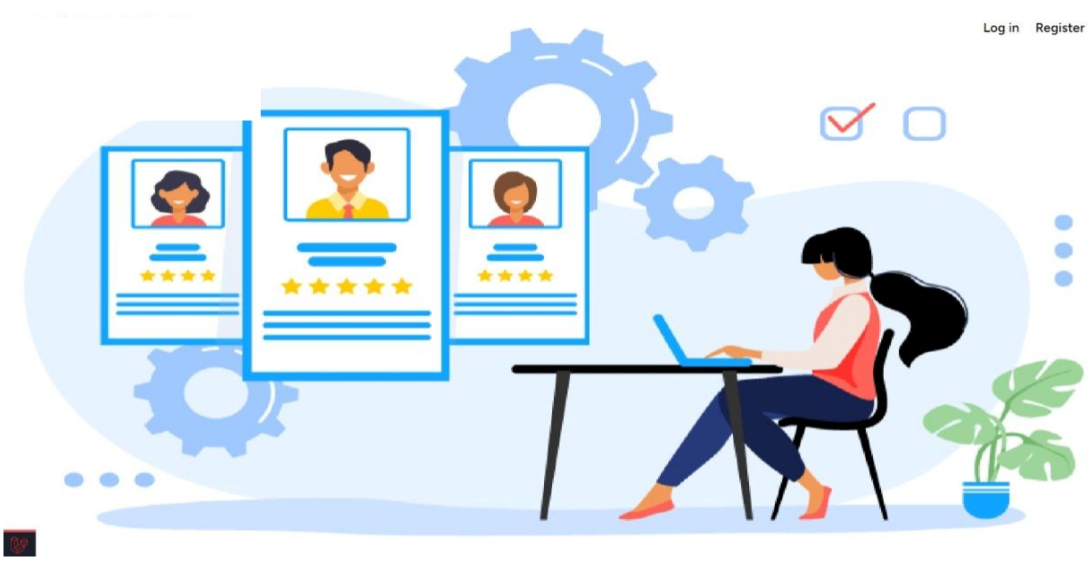
```
    errors={errors.dateOfBirth}
    type="date"
    placeholder=""
    label="Date Of Birth"
    id="dateOfBirth"
    required={true}
    pattern={/^\d{4}-\d{2}-\d{2}$/}
    message="please enter valid date"
  />
  <FormGroup
    register={register}
    errors={errors.superuser}
    type="checkbox"
    id="superuser"
    required={false}
  />
</div>
<div className="col-6">
  <FormGroup
    register={register}
    errors={errors.aadharNo}
    type="text"
    placeholder="XXXX XXXX XXXX"
    label="Aadhar Card No"
    id="aadharNo"
    required={true}
    pattern={/^[2-9][0-9]{3} [0-9]{4} [0-9]{4}$/}
    message="enter valid aadhar number"
  />
  <FormGroup
    register={register}
    errors={errors.panNo}
    type="text"
    placeholder="Enter your PAN number"
    label="PAN No"
    id="panNo"
    required={true}
    pattern={/[A-Z]{5}[0-9]{4}[A-Z]{1}/}
    message="enter valid PAN number"
  />
  <FormGroup
```

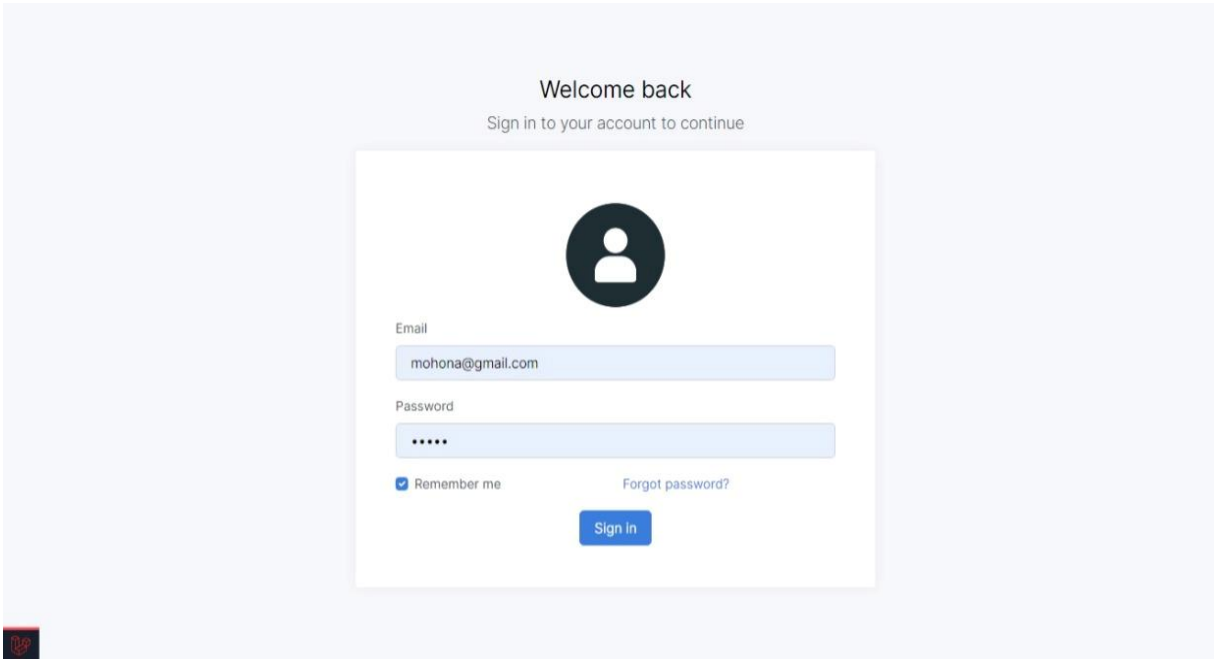
```
      register={register}
      errors={errors.address}
      type="text"
      placeholder="City, State"
      label="Address"
      id="address"
      required={true}
    />
    <FormGroup
      register={register}
      errors={errors.linkedIn}
      type="text"
      placeholder="enter your linkedInID"
      label="LinkedIn ID"
      id="linkedIn"
      required={true}
    />
    <FormGroup
      register={register}
      errors={errors.githubId}
      type="text"
      placeholder="enter your github id"
      label="Github ID"
      id="githubId"
      required={true}
    />
    <FormGroup
      register={register}
      errors={errors.tel}
      type="tel"
      placeholder=""
      label="Phone Number"
      id="tel"
      required={true}
      pattern={/^[+]{1}(?:[0-9\-\\(\\)\\/.]\s?){6,15}[0-9]{1}$/}
      message="please enter valid phone number"
    />
  </div>
</div>
<Button variant="" type="submit" className="custom-button w-100 p-2">
  Submit
```

```
          </Button>
        </Form>
      </div>
    </div>
  );
};

export default NewUser;
}
```

Output:





Welcome back

Sign in to your account to continue

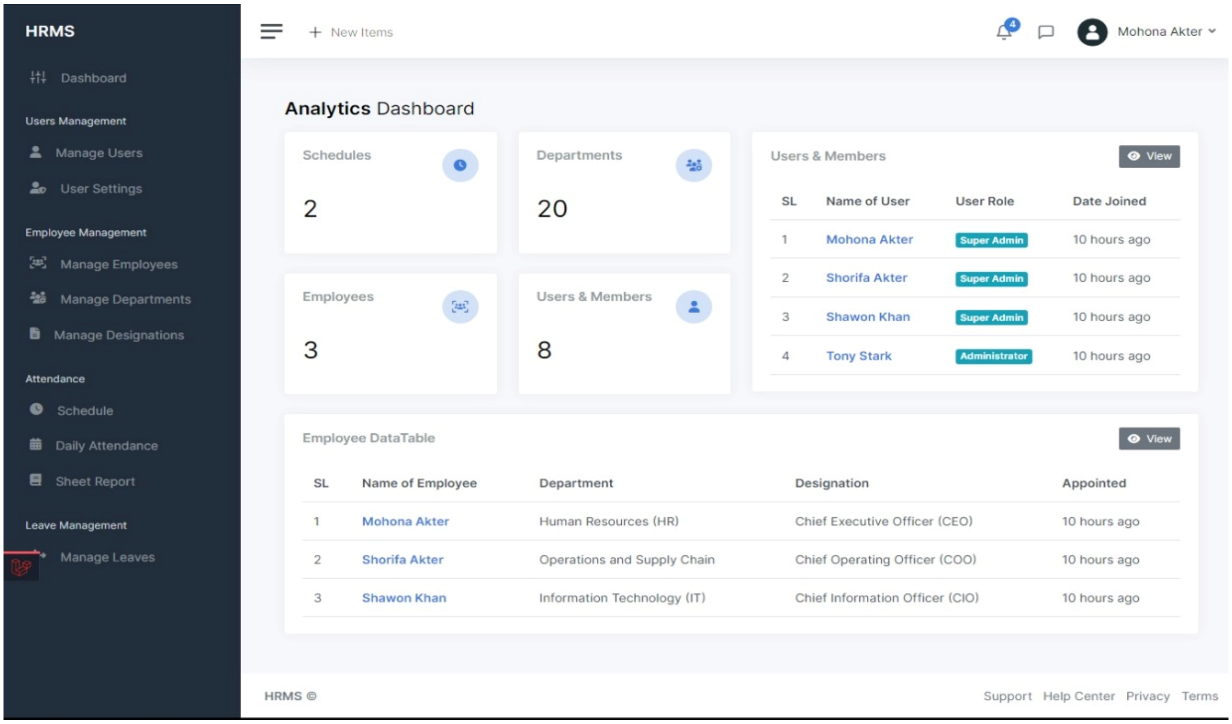
Email

mohona@gmail.com

Password

•••••

☑ Remember me



Analytics Dashboard

Schedules 2

Departments 20

Employees 3

Users & Members 8

Users & Members

SL	Name of User	User Role	Date Joined
1	Mohona Akter	Super Admin	10 hours ago
2	Shorifa Akter	Super Admin	10 hours ago
3	Shawon Khan	Super Admin	10 hours ago
4	Tony Stark	Administrator	10 hours ago

Employee DataTable

SL	Name of Employee	Department	Designation	Appointed
1	Mohona Akter	Human Resources (HR)	Chief Executive Officer (CEO)	10 hours ago
2	Shorifa Akter	Operations and Supply Chain	Chief Operating Officer (COO)	10 hours ago
3	Shawon Khan	Information Technology (IT)	Chief Information Officer (CIO)	10 hours ago

Conclusion

Employee Management Systems (EMS) represent a pivotal advancement in modern HR practices, offering organizations a comprehensive solution to optimize workforce management. By automating routine tasks, streamlining processes, and providing valuable insights through analytics, EMS contribute significantly to operational efficiency, compliance, and employee satisfaction. The diverse range of features, including payroll processing, performance management, and self-service portals, underscores the versatility and effectiveness of EMS in addressing the multifaceted needs of today's businesses. Despite challenges such as resistance to change and data security concerns, proactive strategies and robust implementation approaches can mitigate these obstacles. Looking ahead, the future of EMS is poised to be shaped by emerging trends such as artificial intelligence and mobile accessibility, further enhancing their functionality and impact. As organizations continue to navigate evolving workforce dynamics, EMS will remain indispensable tools for driving organizational success and fostering a positive work environment.

REFERENCES

1. S. Kit Yeng, A. Osman, S. N. Salahuddin, S. Abdullah, Y. J. Lim and C. L. Sim, "Relative Advantage and Competitive Pressure towards Implementation of E-commerce: Overview of Small and Medium Enterprises (SMEs)," Procedia Economics and Finance, vol. 35, pp. 434-443, 2016.

2. M. K. Alrousan and E. Jones, "A conceptual model of factors affecting e-commerce adoption by SME owner/managers in Jordan," International Journal of Business Information Systems, vol. 21, no. 3, 2016.

3. Kemenkopukm, "KemenKopUKM Fasilitasi Inkubasi Wirausaha Tingkatkan Kualitas Dan Kuantitas Wirausahawan," 8 December 2021. [Online]. Available: https://kemenkopukm.go.id/read/kemenkopukm-fasilitasi-inkubasiwirausaha-tingkatkan-kualitas-dan-kuantitas-wirausahawan. [Accessed 02 May 2022].

4. R. Rahayu and J. Day, "Determinant Factors of E-commerce Adoption by SMEs in Developing Country: Evidence from Indonesia," Procedia - Social and Behavioral Sciences, vol. 195, pp. 142-150, 2015.

5. S. Kurnia, J. Choudrie, R. M. Mahbubu and B. Alzougool, "Ecommerce technology adoption: A Malaysian grocery SME retail sector study," Journal of Business Research, vol. 68, no. 9, pp. 19061918, 2015.

6. S. Dahbi and C. Benmoussa, "What Hinder SMEs from Adopting Ecommerce? A Multiple Case Analysis," Procedia Computer Science, vol. 158, pp. 811-818, 2019.

7. S. Kabanda and I. Brown, "A structuration analysis of Small and Medium Enterprise (SME) adoption of E-Commerce: The case of Tanzania," Telematics and Informatics, vol. 34, no. 4, pp. 118-132, 2017.

8. Sommerville, Software Engineering, 9th ed., Boston, Mass: Addison-Wesley, 2009.

9. Fernando, Anharudin and Fadli, "Rancang Bangun Aplikasi EPortofolio Hasil Karya Mahasiswa UNSERA Menggunakan Metode Scrum," JSiI (Jurnal Sistem Informasi), vol. 5, no. 1, pp. 7-12, 2018. R. Febrianto, A. Wulansari and Latipah, "Pengembangan Sistem Pengelolaan dan Pemantauan Proyek dengan Metode Agile Pola Scrum," Jurnal Teknik Informatika Dan Sistem Informasi, vol. 6, no. 2, pp. 206-221, 2020.