INDUSTRY / RESEARCH INTERNSHIP

An

Internship Report Submitted To

VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Belagavi, Karnataka

For The Award of Degree

Bachelor of Engineering

By

Mr. Samarth Deshmukh
USN: 2JI22CS087

Guide
Prof. Keerti Neeralgimath
Assistant Professor
Department of Computer Science and Engineering
Jain College of Engineering, Belagavi

Department of Computer Science and Engineering
JAIN COLLEGE OF ENGINEERING (JCE)
Belagavi, Karnataka – 590014
2025-2026

JGI’s

JAIN COLLEGE OF ENGINEERING (JCE)

BELAGAVI

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Certificate

Certified that **Mr. Samarth Deshmukh,** an undergraduate student bearing **USN 2JI22CS087** has satisfactorily completed the **Internship** on **“FullStack Development”, at JCE Innovation and Entrepreneurship Foundation** submitted to **Visvesvaraya Technological University, Belagavi** in partial fulfillment for the award of **Bachelors** in **Computer Science and Engineering.**

Guide	**Internship Co-ordinator**	**Principal & Director**

Visvesvaraya Technological University, Belagavi

CERTIFICATE

Certified that the Industry Internship

"FullStack Development"

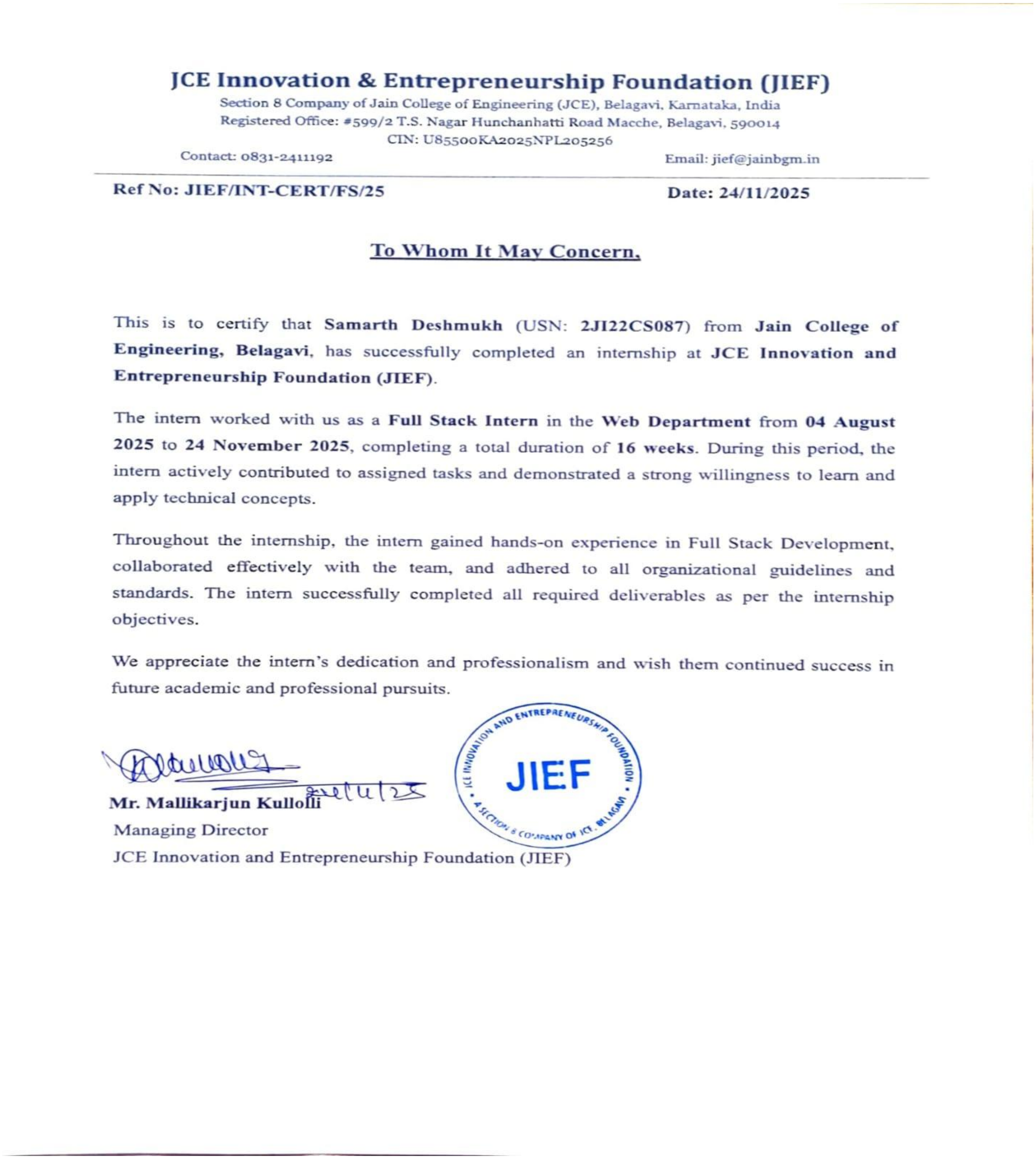
is a bonafide work carried out by

Mr. Samarth Deshmukh
USN: 2JI22CS087

In partial fulfillment for the award of **BACHELORS IN COMPUTER SCIENCE AND ENGINEERING** *of the* ***Visvesvaraya Technological University, Belagavi.*** *The report has been approved as it satisfies the academic requirements in respect of Internship Work prescribed for the said degree*

Name of the Examiners **Signature with date**

Internship Certificate



JCE Innovation & Entrepreneurship Foundation (JIEF)

Section 8 Company of Jain College of Engineering (JCE), Belagavi, Karnataka, India
Registered Office: #599/2 T.S. Nagar Hunchanhatti Road Macche, Belagavi, 590014
CIN: U85500KA2025NPL205256

Contact: 0831-2411192 Email: jief@jainbgm.in

Ref No: JIEF/INT-CERT/FS/25 **Date: 24/11/2025**

To Whom It May Concern,

This is to certify that **Samarth Deshmukh** (USN: **2JI22CS087**) from **Jain College of Engineering, Belagavi**, has successfully completed an internship at **JCE Innovation and Entrepreneurship Foundation (JIEF)**.

The intern worked with us as a **Full Stack Intern** in the **Web Department** from **04 August 2025** to **24 November 2025**, completing a total duration of **16 weeks**. During this period, the intern actively contributed to assigned tasks and demonstrated a strong willingness to learn and apply technical concepts.

Throughout the internship, the intern gained hands-on experience in Full Stack Development, collaborated effectively with the team, and adhered to all organizational guidelines and standards. The intern successfully completed all required deliverables as per the internship objectives.

We appreciate the intern's dedication and professionalism and wish them continued success in future academic and professional pursuits.

24/11/25

JCE INNOVATION AND ENTREPRENEURSHIP FOUNDATION
JIEF
A SECTION 8 COMPANY OF JCE, BELAGAVI

Mr. Mallikarjun Kullolli
Managing Director
JCE Innovation and Entrepreneurship Foundation (JIEF)

Acknowledgements

I am grateful and would like to express my sincere gratitude to my industry guide Mr. Mallikarjun Kullolli for his untiring guidance, continuous encouragement and constant support which made this Internship study possible. Sir has impressed me with his outstanding professional conduct and his strong conviction towards the Internship study. I am truly obliged for his progressive vision throughout the internship.

I am grateful to our beloved and respected Principal & Director **Dr. J Shivakumar** and my beloved Head of The Department of Computer Science & Engineering **Dr. Uttam Patil** for providing all there required resources for the successful completion of my internship.

I would like to express my sincere gratitude to my respected guide **Prof. Keerti Neeralgimath** Assistant Professor, Department of CSE**,** for her/his valuable suggestions and guidance in the preparation of the internship.

Finally I would like to thank my parents, Friends and all teaching and non-teaching staff members of CSE for all the help and co-ordination extended in bringing out this internship successfully in time.

Samarth Deshmukh
[2JI22CS087]

Sri Bhagawan Mahaveer Jain Educational & Cultural Trusts'

Jain College of Engineering Belagavi

approved by AICTE, New-Delhi, affiliated to VTU Belagavi, recognized by Govt. of Karnataka
accredited by NBA(Dept. of CVE, CSE, ECE & EEE)

Estd.2010

DECLARATION

I, Samarth Deshmukh student of Seventh semester B.E at the Department of Computer Science and Engineering, **Jain College of Engineering, Belagavi** declare that the **Internship (BINT883A)** entitled " FullStack Development " has been carried out by me and submitted in partial fulfilment of the course requirements for the award of degree in Bachelor of Engineering in Computer Science & Engineering discipline of **Visvesvaraya Technological University,** Belagavi during the academic year **2025-26.**

Place: Belagavi

Samarth Deshmukh

Date: **[2JI22CS087]**

ABSTRACT

The Industry Internship provided valuable practical exposure to real-world software development processes and industry practices. It offered an opportunity to apply academic knowledge in a professional environment and gain hands-on experience in full-stack web application development. During the internship, I worked on designing, developing, and deploying a complete project using modern technologies such as React.js, Node.js, Spring Boot, and MySQL.

The internship enhanced my understanding of end-to-end software development, including frontend design, backend API creation, database management, and real-time data handling. I also learned about project planning, teamwork, version control (Git/GitHub), and deployment techniques. The experience helped strengthen both technical and soft skills, such as problem-solving, collaboration, and communication.

Overall, the industry internship served as a bridge between theoretical learning and practical implementation, providing a strong foundation for future professional growth in the field of full-stack development and modern software engineering.

Table of Contents

CHAPTER 1

INTRODUCTION

1.1 BRIEF HISTORY OF THE ORGANIZATION

The **JCE Innovation & Entrepreneurship Foundation (JIEF)** was established as a strategic initiative under the Jain College of Engineering (JCE**)** ecosystem, which operates under the prestigious JGI Group. The foundation was conceptualized with the vision of creating a dedicated platform that nurtures innovation, entrepreneurship, and skill development among students and aspiring entrepreneurs. Formed as a nonprofit, Section-8 style organization, JIEF aims to create impactful opportunities through structured programs, incubation support, and industry collaboration.

JIEF was founded to bridge the existing gap between academia and industry, recognizing the need for hands-on learning, startup exposure, and real-world problem-solving abilities in students. From its early days, the foundation focused on developing a campus-based startup ecosystem that encourages idea generation, innovation challenges, entrepreneurial thinking, and technology-driven solutions. It began by conducting awareness sessions, workshops, and entrepreneurship development programs that helped students understand the fundamentals of business creation and startup culture.

In its initial phase, JIEF primarily engaged with students and early-stage entrepreneurs by providing mentorship, technical guidance, and skill-building programs. It collaborated with experienced mentors, industry professionals, and academicians to deliver expert-led training on innovation, design thinking, business planning, product development, and leadership. The foundation also introduced several incubation-related activities such as startup ideation events, prototype development support, and innovation competitions to help students transform their ideas into viable projects.

Over time, JIEF expanded its offerings to include pre-incubation and incubation support, helping young innovators move from concept to execution. The foundation actively supports student-led startups through resources like workspace, networking opportunities, guidance on funding avenues, and connections with industry partners. By adopting a structured approach, JIEF has steadily built a culture of entrepreneurship within the campus and positioned itself as a catalyst for student innovation in the region.

Today, JCEF continues to strengthen its mission by promoting innovation-driven learning, entrepreneurship development, and collaboration between academia, industry, and the startup ecosystem. The foundation plays a vital role in empowering the next generation of innovators by giving them the platform, support, and mentorship needed to build sustainable and impactful ventures.

1.2 COMPANY PROFILE

The JCE Innovation & Entrepreneurship Foundation (JIEF) was established under the esteemed Jain College of Engineering (JCE) and operates within the broader ecosystem of the JGI Group, Karnataka. Formed as a nonprofit, Section-8 style foundation, JIEF was created with the core objective of fostering a vibrant culture of innovation, technology adoption, and entrepreneurship among students, researchers, and early-stage startups. The organization strives to serve as a bridge between academia, industry, and society by promoting idea development, incubation, skill-building, and collaborative innovation.

JIEF functions as a regional entrepreneurship and innovation hub that empowers young innovators through structured mentorship programs, incubation services, technology training, and research support. With a strong foundation built on academic excellence and industry partnerships, the organization focuses on enabling students and aspiring entrepreneurs to transform their ideas into viable business ventures.

The foundation offers a wide range of programs, including pre-incubation and incubation support, startup acceleration, patent facilitation, prototype development assistance, and entrepreneurship development workshops. Through these initiatives, JIEF provides an environment where creativity, problem-solving, and innovation flourish.

JIEF operates with strong leadership under its management board:

- Dr. J. Shivakumar — Chairman, JIEF
- Mr. Mallikarjun Kullolli — Managing Director
- Dr. Sanjeev Sangami — Director

Guided by experienced academicians, entrepreneurs, and industry mentors, the organization ensures high-grade support, strategic direction, and impactful decision-making for fostering the entrepreneurial ecosystem on campus.

The foundation works toward a clear vision:

Vision
To position JCE Innovation & Entrepreneurship Foundation as a regional hub for innovation and entrepreneurship.

Mission

1. Incubate and accelerate startups, contributing to job creation and economic development.
2. Train students and faculty in entrepreneurship, business fundamentals, and emerging technologies.
3. Facilitate patent filings and collaborative research, inspiring innovation with tangible outcomes.
4. Develop community-focused and technology-driven solutions to address real societal challenges.

With a growing footprint, JIEF has become a catalyst for innovation and entrepreneurship in the region. It helps students gain hands-on exposure to startup culture through hackathons, accelerators, mentorship engagements, field visits, and expert-led technology bootcamps. The foundation actively supports idea-stage innovators by providing them with workspace, technical guidance, and access to domain experts.

1.3 Application Development

Complete range of application development services. We develop products as per your business requirements. To make your life simple we take care of entire development cycle under one roof from product architecture, development

Solve Your Complex Business Problems by the Simplicity of our Application Development Solutions

Advantage of Engaging JIEF Innovation & Development Services

- **Cost-Effective Support & Faster Time-to-Market for Startups and SMEs** JIEF provides affordable innovation and incubation support, helping early-stage ventures launch faster and more efficiently.
- **Flexibility & Customization in Technology and Business Solutions** We assist startups in tailoring open-source, digital, and operational solutions to meet their unique needs.
- **Modernizing Existing Business and Technical Systems** JIEF supports entrepreneurs in upgrading legacy ideas, prototypes, and systems into scalable and modern solutions.
- **Building Effective, Scalable, and Market-Ready Innovations** Our development support, mentorship, and infrastructure help startups solve pressing business and societal challenges.
- **Access to Expertise, Mentorship & High-Quality Development Resources** Entrepreneurs benefit from our ecosystem of industry mentors, academic experts, and young engineers passionate about innovation.

Our team of young innovators and engineers works across various development platforms, offering **cross-domain compatibility** and supporting multiple sectors.

Custom Innovation & Solution Development

At JIEF, we follow a flexible and proven development methodology to ensure that every prototype, MVP, and business solution aligns with user needs and stakeholder expectations. We have a track record of effectively supporting ventures of different scales, domains, and maturity levels.

Product Innovation & Startup Development

We understand the challenges in product lifecycle management—from ideation to market entry. JIEF adopts a structured Startup & Product Release Framework that reduces development costs, accelerates innovation cycles, and enhances quality control across product architecture and deployment.

Quality Assurance, Validation & Testing

Quality assurance is a core focus at JIEF. We believe that validation, testing, and user-centric refinement are essential to reduce risks, minimize costs, and ensure successful product deployment. JIEF offers comprehensive QA support—technical, functional, business, and market-fit—to help startups deliver high-quality, reliable products.

1.4 DIGITAL TRANSFORMATION & WEB SOLUTIONS

Empowering Startups and Enterprises to Go Digital

At JIEF, we help innovators, startups, and institutions establish a strong online presence. Our services include website development and digital solution building for any type of industry or domain.

With an ecosystem backed by experts and innovation-driven engineers, we support the creation of static, dynamic, and e-commerce websites across desktops and mobile platforms— incorporating custom features tailored to your specific needs.

JIEF serves as a trusted innovation partner, offering custom web application development**,** software prototyping**,** and digital transformation solutions to startups and enterprises. Our focus is to build scalable, modern, and cost-effective digital solutions that leverage our multi-domain expertise, strong technical capabilities, and structured development methodologies.

We stay aligned with emerging technologies and new digital platforms—enabling us to build cutting-edge websites, portals, and web applications**.**
Our team is proficient in modern open-source frameworks (WordPress, Drupal, Joomla, PHP), and we also support development of custom applications in advanced frameworks such as ASP.NET**,** Java, Python, Spring Boot, and modern JavaScript ecosystems.

Web 2.0 Design & Development

Web 2.0 has transformed how people interact with digital platforms. At JIEF, we design and develop **interactive, user-centric, and responsive Web 2.0 interfaces** that enhance user engagement, simplify navigation, and elevate the digital experience for both creators and end-users.

E-Commerce Solutions

JIEF provides end-to-end support for B2B and B2C e-commerce platforms, enabling startups and businesses to launch secure, scalable, and feature-rich online stores.
Our structured approach, modern methodologies, and expert use of technology help us build:

- Efficient shopping cart systems
- Custom e-commerce platforms
- Secure payment integrations
- Marketplace-style applications

These solutions are designed for performance, user experience, and business growth.

Content Management Systems (CMS)

Managing large volumes of content can be challenging without an efficient system. JIEF helps startups and organizations build and maintain powerful Content Management Systems that allow easy management of text, media, documents, and digital assets.

We work with popular CMS platforms and also build custom CMS solutions tailored to client-specific workflows and requirements.

Enterprise Web Application Development

JIEF supports the digital transformation of institutions, startups, and growing businesses by developing enterprise-grade web applications that improve productivity, collaboration, and organizational agility.
Our enterprise solutions help transform traditional environments into next-generation, connected, and scalable digital workplaces**.**

We design and deliver:

- Enterprise portals
- Internal management systems
- Community and collaboration platforms
- Data-driven operational applications
- Custom business process automation tools

1.5 Support & Maintenance

At **JIEF**, we believe in building long-term relationships where your success becomes our success. Our expert team helps you choose the right technology, delivers high-quality development, and supports you at every stage of your digital journey. With extensive experience across industries—including travel, retail, services, and enterprise applications—we ensure reliable and future-ready solutions.

1.5.1 Support & Maintenance Package (SMP)

JIEF offers a comprehensive Support & Maintenance Package (SMP) with all its software products. The SMP provides:

- Priority technical support
- All major product upgrades
- Minor releases at no additional cost
- Continuous monitoring and issue resolution

This ensures your systems remain secure, up to date, and aligned with evolving business needs.

1.5.2 Application Maintenance

At JIEF, our Application Maintenance Services (AMS) go far beyond simply managing legacy or overgrown applications. We understand that a significant portion of IT budgets is spent on maintenance—often more than anticipated. Through proactive monitoring, optimization, and enhancement, we help you:

- Reduce operational costs
- Improve application performance
- Enhance stability and scalability
- Extend the life and value of existing systems

Our aim is to keep your applications running smoothly so your teams can focus on innovation.

1.5.3 Website Maintenance

Launching your new website is just the beginning. To keep it relevant and performing well, regular updates and content maintenance are essential**.** Our website maintenance services help you:

- Keep information fresh and accurate
- Improve search engine rankings
- Increase user engagement and site traffic
- Maintain a strong and credible online presence

From content updates to security patches and performance improvements—we ensure your website always delivers its best.

1.5.4 System Administration

The JIEF technical team brings deep expertise in configuring, maintaining, and migrating both UNIX-based systems and NT/Windows-based servers. Effective system administration is critical to business continuity and operational efficiency. Our services include:

- Server setup and configuration

- Security hardening
- Backup and recovery management
- System performance monitoring
- Migration and virtualization support

With JIEF managing your systems, you can be confident that your IT infrastructure is secure, stable, and optimized for performance.

CHAPTER 2

TECHNOLOGY LEARNT

During my Full Stack Development internship, I gained hands-on experience in a wide range of modern technologies used in both frontend and backend development, as well as tools essential for deployment, version control, and database management. These technologies helped me understand the complete workflow of building, integrating, and deploying full-stack applications.

2.1FRONTEND TECHNOLOGIES

2.1.1 HTML

- I learned how to create structured and semantic web pages using modern HTML5 elements such as <header>, <nav>, <section>, <article>, and <footer>, which help in organizing content meaningfully.
- Understood the importance of semantic tags for improving accessibility, browser readability, and SEO performance.
- Worked extensively with HTML forms, including various input types, labels, placeholders, validation attributes, and form submission methods.
- Learned how to embed multimedia elements like images, audio, and video using tags such as <img>, <audio>, and <video>.
- Practiced creating tables, lists, hyperlinks, buttons, and other essential UI elements used in web pages.
- Gained knowledge of how HTML defines the basic document structure through elements like <!DOCTYPE html>, <html>, <head>, and <body>.
- Understood how HTML interacts with CSS and JavaScript to form the core foundation of the frontend development process.
- Learned to follow best practices like writing clean, readable, and well-formatted code, using comments, structuring elements properly, and ensuring accessibility compliance.
- Worked on projects where HTML served as the base layout for dynamic content rendering using JavaScript and backend integration.
- Improved understanding of how browsers interpret HTML and render content in the Document Object Model (DOM).

2.1.1 CSS

- I learned how to use CSS to control the design, layout, and visual appearance of web pages, giving structure and content a polished look.
- Understood how to apply different types of CSS such as inline, internal, and external styles, and recognized why external style sheets are preferred for maintainability.
- Worked with essential CSS concepts like selectors, properties, values, colors, borders, backgrounds, margins, padding, and typography.
- Gained experience using Flexbox to create flexible, aligned, and responsive layouts without relying on float-based structures.
- Learned to work with CSS Grid, which helps in building complex, multi-row and multi-column layouts efficiently.
- Practiced styling forms, buttons, navigation bars, images, tables, and other UI components to improve user experience.
- Worked with responsive web design (RWD) techniques, including media queries, relative units (%, em, rem), and viewport settings to ensure pages work on all screen sizes.
- Learned how to use pseudo-classes and pseudo-elements such as :hover, :active, :focus, ::before, and ::after for interactive UI effects.
- Explored CSS animations and transitions to add smooth visual effects and improve user engagement.
- Understood how to organize CSS code using class naming conventions, comments, and reusable styles to maintain clean and scalable stylesheets.
- Learned how CSS integrates with HTML and JavaScript to create complete, responsive, and interactive user interfaces.
- Introduced to CSS frameworks like Bootstrap/Tailwind (optional), which help in speeding up UI development with pre-defined classes.
- Improved understanding of browser rendering, the box model, and how CSS affects layout flow and element positioning.

2.1.1 JAVASCRIPT

- I learned that JavaScript is the core programming language of the web, responsible for adding interactivity, dynamic behavior, and logic to webpages.
- Understood how JavaScript interacts with the DOM (Document Object Model) to update HTML elements, change styles, and modify the structure of a webpage in real-time.
- Gained experience with variables, data types, operators, conditionals, loops, and other basic programming concepts in JavaScript.
- Practiced using functions, including function expressions, arrow functions, and callback functions, to write reusable and modular code.
- Learned how to handle events such as click, submit, keypress, and mouse movements, allowing webpages to respond to user actions.
- Worked with arrays and objects, including methods like map(), filter(), reduce(), and object manipulation techniques used in modern JavaScript.
- Understood the concept of asynchronous programming, including callbacks, promises, async/await, and how JavaScript handles non-blocking operations.
- Learned how to perform API calls using Fetch/AJAX, enabling real-time data loading and communication with backend services.
- Explored ES6+ features such as template literals, let/const, destructuring, spread/rest operators, and modules, which make JavaScript code cleaner and more efficient.
- Gained knowledge of error handling using try...catch and debugging techniques with browser developer tools.
- Practiced manipulating and validating HTML forms through JavaScript to improve user experience and prevent invalid data submission.
- Understood how JavaScript integrates with CSS animations, transitions, and DOM manipulation to create smooth UI interactions.
- Learned about localStorage and sessionStorage for temporary data storage in the browser.
- Applied JavaScript in real project tasks such as form validation, dynamic content loading, interactive UI components, and API integration.

2.1.1 TAILWIND CSS

- I learned that Tailwind CSS is a utility-first CSS framework that allows styling directly in HTML using predefined utility classes, making UI development faster and more efficient.
- Understood how Tailwind eliminates the need for writing long custom CSS files by providing ready-made utility classes for spacing, colors, typography, borders, shadows, and layout.
- Worked with responsive design utilities like sm:, md:, lg:, and xl: to quickly build layouts that adapt to different screen sizes.
- Learned how Tailwind encourages a mobile-first approach, helping create responsive pages starting from smaller screens.
- Used Flexbox and Grid utilities such as flex, grid, justify-between, items-center, and grid-cols-* to build modern and flexible layouts.
- Gained experience customizing colors, fonts, and spacing scales using the Tailwind configuration file (tailwind.config.js).
- Understood how Tailwind improves developer productivity by allowing design decisions directly within the markup, reducing context switching between HTML and CSS files.
- Learned how to use hover, focus, active, and transition utilities to create interactive UI behaviors without writing custom CSS.
- Worked with utility classes for spacing and sizing like p-*, m-*, w-*, h-*, and gap-* to maintain consistent design.
- Gained knowledge of dark mode support using utilities such as dark:bg-gray-800 and understanding Tailwind’s media-query-based dark mode system.
- Understood how Tailwind works well with component-based frameworks like React, Angular, and Vue by enabling compact, reusable UI components.
- Learned about Tailwind’s Just-In-Time (JIT) compiler, which generates styles on demand, resulting in smaller CSS output and faster builds.
- Applied Tailwind CSS in real project tasks such as designing dashboards, forms, navigation bars, buttons, grids, and responsive layouts.

2.1.1 REACT

- I learned that React is a JavaScript library for building fast, dynamic, and component-based user interfaces, widely used in modern frontend development.
- Understood the concept of components, which allow breaking the UI into reusable, independent building blocks that make development more efficient and organized.
- Gained experience with JSX (JavaScript XML), which allows writing HTML-like code inside JavaScript to make UI creation more intuitive.
- Worked with props for passing data between components and state for managing dynamic data within a component.
- Understood how React follows a unidirectional data flow, making the application logic predictable and easier to debug.
- Learned how to handle events, update state, and re-render the UI automatically based on changes.
- Gained hands-on experience with React Hooks such as useState, useEffect, useRef, and useContext to manage state, lifecycle events, and shared data.
- Understood the virtual DOM, which improves performance by updating only the parts of the UI that change instead of reloading the entire page.
- Learned how to create functional components, which are simpler and more commonly used than class components in modern React development.
- Worked with React Router for implementing navigation and routing between multiple pages or views in a single-page application.
- Learned how to integrate external APIs using fetch or axios, handle asynchronous calls, and display data dynamically.
- Explored conditional rendering, lists rendering, and dynamic UI updates using JavaScript expressions in JSX.
- Understood the importance of component reusability, folder structure, and code organization for large-scale React applications.
- Worked with CSS frameworks (like Tailwind) and React component libraries to speed up UI development.

- Learned the basics of state management solutions such as Context API and Redux for handling large and complex application states.
- Gained practical experience by building components like forms, tables, modals, navigation menus, dashboards, and dynamic data-driven interfaces.

2.1.1 Advantages of Frontend Technologies

1. Enhanced User Experience

- Frontend technologies help create interactive, responsive, and visually appealing interfaces that improve user satisfaction.

2. Faster Development

- With frameworks like React, and libraries like Tailwind/Bootstrap, developers can build UIs faster using pre-built components and utilities.

3. Improved Performance

- Modern frontend tools reduce unnecessary page reloads and allow dynamic content updates, leading to faster load times and better performance.

4. Reusability of Components

- Component-based frameworks allow developers to reuse UI components, saving development time and reducing code duplication.

5. Better Maintainability

- Clear structure, modular code, and separation of concerns make the frontend easier to maintain and update**.**

2.1.1 Disadvantages of Frontend Technologies

1. Fast-Changing Ecosystem

- Frontend technologies evolve quickly, requiring continuous learning to keep up with new updates, frameworks, and trends.

2. Browser Compatibility Issues

- Different browsers interpret CSS and JavaScript slightly differently, sometimes causing inconsistent behavior or UI problems.

3. Performance Overload

- Adding too many libraries, heavy scripts, or animations can slow down the page, reducing performance.

4. Higher Initial Learning Curve

- Frameworks like Angular or React require understanding concepts like state, components, hooks, routing, build tools, etc., which may be challenging for beginners.

5. Security Risks

- Frontend applications are more exposed to attacks such as XSS (Cross-Site Scripting) if proper validation and sanitization are not done.

2.2 BACKEND TECHNOLOGIES

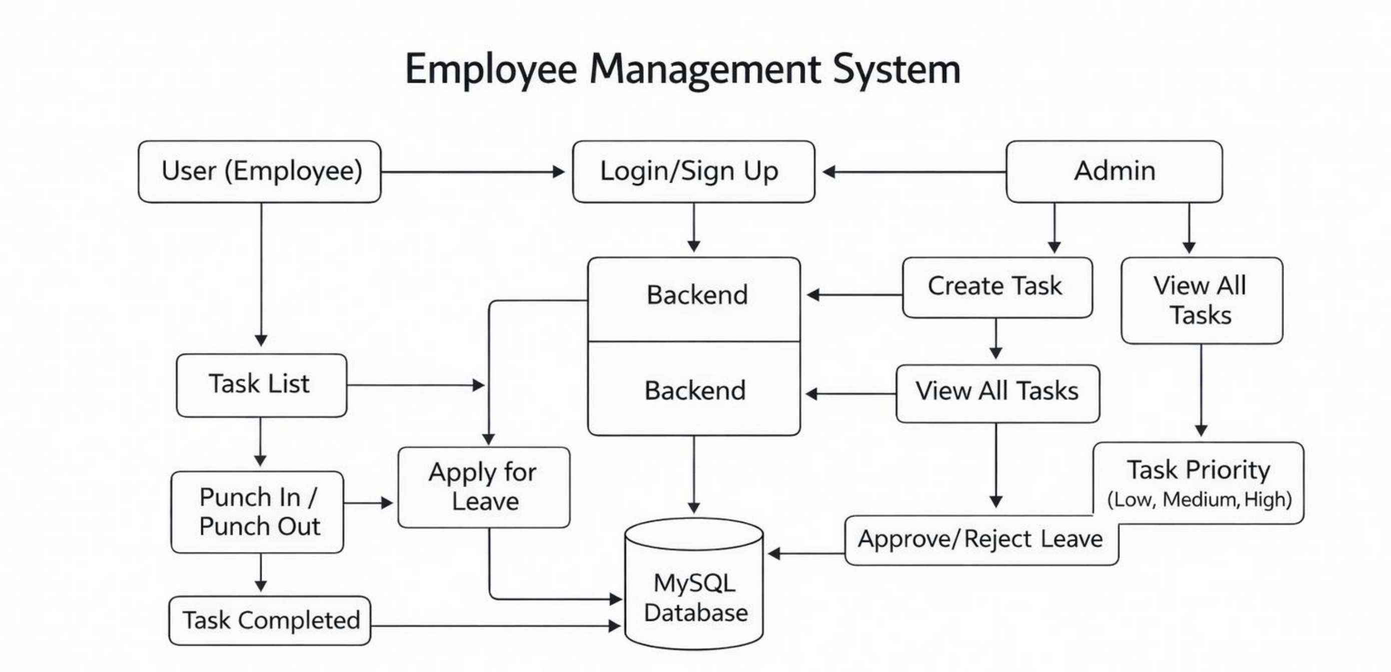


Fig 3.2 Application Flow Chart

2.2.1 NODE.JS :

- I learnt that Node.js is a JavaScript runtime that allows JavaScript to run outside the browser, mainly for backend development.
- I understood how Node.js uses an event-driven and non-blocking architecture, which helps in handling multiple requests efficiently.
- I learnt how to use NPM (Node Package Manager) to install and manage libraries required for backend development.

- I worked with Express.js, a lightweight framework used to build servers and REST APIs in Node.js.
- I gained knowledge of creating routes, handling different HTTP methods (GET, POST, PUT, DELETE), and sending responses to the frontend.
- I learnt how to use middleware in Express.js for processing requests, handling errors, and implementing authentication.
- I understood how to connect Node.js applications with databases like MySQL or MongoDB to perform backend CRUD operations.
- I used environment variables (.env files) to securely store sensitive backend configurations such as database credentials.
- I practised organizing the backend code into modules, such as controllers, routes, and services, for better structure.
- I learnt how to test backend APIs using tools like Postman or Thunder Client to ensure the correct flow of data.
- I also understood how Node.js makes backend development faster by allowing full-stack JavaScript development, using the same language for both frontend and backend.

2.2.2 EXPRESS.JS

- I learnt that Express.js is a minimal and flexible web framework built on top of Node.js, used for creating server-side applications and APIs.
- I understood how Express.js simplifies backend development through easy routing, middleware handling, and structured application flow.
- I gained experience in creating routes to manage different URL paths and handle requests using methods like GET, POST, PUT, and DELETE.
- I learnt how to use middleware functions for tasks such as logging, validating data, parsing JSON, and handling errors.
- I worked on building RESTful APIs using Express.js, allowing the frontend to communicate with the server efficiently.

- I understood how to send JSON responses, handle parameters, and manage request bodies using built-in and third-party middleware.
- I learnt how to structure backend applications using controllers, routers, and services for better readability and maintainability.
- I integrated Express.js with databases like MySQL or MongoDB to perform CRUD operations and return query results to the client.
- I used tools like Postman and Thunder Client to test API endpoints and ensure the backend logic works correctly.
- I also understood how Express.js supports environment variables, authentication middleware, and security practices to build safer and scalable backend applications.

2.2.3 ADVANTAGES OF BACKEND TECHNOLOGIES

1. Efficient Data Management

Backend technologies allow applications to store, retrieve, update, and process data efficiently using databases like MySQL, MongoDB, or PostgreSQL.

2. Strong Application Logic

They handle complex business logic on the server side, ensuring applications behave correctly and securely.

3. High Security

Backend frameworks include security features like authentication, authorization, input validation, and encryption.

4. Scalability

Backend systems can handle increased users, large data loads, and heavy processing using scalable frameworks like Node.js, Django, or Spring Boot.

5. API Development

Backend technologies support RESTful and GraphQL APIs, enabling smooth communication between client and server.

2.2.4 DISADVANTAGES OF BACKEND TECHNOLOGIES

1. Requires Strong Technical Knowledge

Backend development involves understanding servers, databases, APIs, and security concepts, making it complex for beginners.

2. Debugging Can Be Difficult

Errors in backend logic, database queries, or server configuration can be challenging to identify and fix.

3. Higher Setup Requirements

Setting up servers, databases, and environments requires more effort compared to front-end development.

4. Performance Bottlenecks

Improper backend code or inefficient queries can reduce system performance and slow down the entire application.

5. Security Vulnerabilities

Backend systems are prone to threats like SQL injection, XSS, CSRF, or unauthorized access if not properly secured.

2.3 DATABASE TECHNOLOGIES

Database technologies refer to the tools, systems, and software used to store, manage, organize, and retrieve data efficiently in web applications. In full stack development, database technologies play a crucial role in handling all the data-related operations of an application.

2.3.1 MySQL

- I learnt that MySQL is a relational database management system (RDBMS) that stores data in structured tables with rows and columns.
- I understood how MySQL uses SQL (Structured Query Language) to insert, update, delete, and retrieve data.
- I gained experience in creating databases, tables, relationships, and constraints, which ensure data accuracy and integrity.
- I learnt how to use primary keys, foreign keys, indexes, and data types to design efficient and normalized database structures.
- I worked with SQL queries including SELECT, JOIN, WHERE, GROUP BY, ORDER BY, and aggregate functions like COUNT, SUM, and AVG.
- I learnt how to perform CRUD operations (Create, Read, Update, Delete) in MySQL from backend applications.
- I understood how MySQL ensures strong consistency and supports structured data storage, making it suitable for transactional applications.
- I explored tools like phpMyAdmin, MySQL Workbench, and command-line MySQL for managing and visualizing databases.
- I also learnt how to connect MySQL with backend technologies such as Node.js, Express.js, Java, or PHP.
- I gained an understanding of query optimization, indexing, and improving database performance.

2.3.1.1 ADVANTAGES OF MYSQL

1. Open Source and Free

MySQL is open-source, which makes it easily accessible and cost-effective for students, developers, and organizations.

2. High Performance

MySQL delivers fast query processing, optimized indexing, and efficient data handling for large-scale applications.

3. Strong Data Integrity

It supports constraints like **primary keys, foreign keys, unique keys**, and ACID compliance, ensuring reliable and consistent data.

4. Easy to Use

MySQL is beginner-friendly with simple SQL commands and GUI tools like MySQL Workbench and phpMyAdmin.

5. Wide Compatibility

It works smoothly with many backend technologies like **Node.js, Java, PHP, Python**, and integrates well with web frameworks.

2.3.1.2 DISADVANTAGES OF MYSQL

2.3.1.2.1 Limited Support for NoSQL-Style Data

MySQL requires structured tables; it is not ideal for storing unstructured or highly flexible data.

2.3.1.2.2 Complex Scaling for Very Large Systems

Horizontal scaling and distributed systems require extra configuration and expertise.

2.3.1.2.3 Performance Drops in Very High Read/Write Loads

Although fast, it may struggle compared to specialized systems like PostgreSQL or NoSQL databases under extreme traffic.

2.3.1.2.4 Less Suitable for Real-Time Big Data

Applications handling real-time analytics or massive unstructured data may require NoSQL databases like MongoDB.

2.3.1.2.5 No True Full Text Search (Compared to Elasticsearch)

Its built-in full-text search capabilities are limited when compared to specialized search engines.

2.3.2 MONGODB

- I learnt that MongoDB is a NoSQL database that stores data in a flexible, document-based format (JSON-like documents).
- I understood how MongoDB allows schema-less design, meaning fields and structures can vary across documents.
- I gained experience in performing MongoDB CRUD operations using methods like insertOne, find, updateOne, deleteOne, etc.
- I learnt how MongoDB stores data inside collections instead of tables, making it ideal for unstructured or semi-structured data.
- I worked with MongoDB Query Language (MQL) for filtering, searching, and updating documents.
- I learnt how MongoDB supports high scalability and performance through sharding, replication, and distributed architecture.
- I understood how to use indexes in MongoDB to improve search speed and data retrieval.
- I worked with tools like MongoDB Compass, Mongo Shell, and Mongoose (ODM) for interacting with the database.
- I learnt how MongoDB integrates easily with Node.js and Express.js, commonly used in MERN stack development.
- I also understood use-cases where MongoDB is more suitable—such as applications needing flexibility, rapid development, or large amounts of unstructured data.

2.3.2.1 ADVANTAGES OF MONGODB

1. Flexible Schema Design

MongoDB is schema-less, which means documents in a collection can have different structures. This makes development faster and easier when data formats frequently change.

2. Stores Data in JSON-like Documents

It uses BSON (Binary JSON), making data more readable, natural, and easy to work with in JavaScript-based applications like Node.js and Express.js.

3. High Scalability

MongoDB supports horizontal scaling through **sharding**, allowing it to handle very large volumes of data and high-throughput applications.

4. Faster Read/Write Performance

Because of its NoSQL design, MongoDB provides high-speed data operations, especially useful for real-time applications.

5. Great for Unstructured and Semi-Structured Data

Perfect for applications where data doesn't fit into rigid table structures, such as user profiles, logs, IoT data, or social media content.

2.3.2.2 DISADVANTAGES OF MONGODB

2.3.2.2.1 No Joins Like SQL Databases

Although MongoDB supports lookup operations, relational joins are limited and not as efficient as traditional SQL joins.

2.3.2.2.2 Higher Memory Usage

Storing documents with varying structures and duplicated keys consumes more memory compared to structured SQL tables.

2.3.2.2.3 Less Suitable for Highly Structured Data

Applications requiring strict relationships (like banking or finance systems) are better served by relational databases.

2.3.2.2.4 Transactions Were Limited Earlier

Although MongoDB now supports multi-document ACID transactions, SQL databases are still considered more reliable for transactional systems.

2.3.2.2.5 Requires Database Design Expertise

Schema-less design can lead to unorganized or inconsistent data if not planned carefully.

2.4 TOOLS USED

2.4.1 POSTMAN

Postman is an API testing tool that I used to test backend routes and verify server responses. It allowed me to send GET, POST, PUT, and DELETE requests to the Express.js server and check the output in a structured format. Postman helped in debugging issues, validating request bodies, headers, and status codes, ensuring that the APIs worked correctly before connecting them to the frontend.

2.4.2 VISUAL STUDIO CODE (VS CODE)

VS Code is a lightweight yet powerful code editor used for writing and managing Node.js applications. It supports features like syntax highlighting, IntelliSense, auto-completion, debugging, extensions, and Git integration. I used VS Code to organize my project structure, write JavaScript files, manage Node packages, and run the development server efficiently.

2.4.3 GIT & GITHUB

Git is a version control system that helped me track changes in my project, manage versions, and collaborate efficiently. With GitHub, I stored my project repositories online, created commits, pushed updates, and maintained an organized code history. These tools ensured safe backup, teamwork flexibility, and easy deployment.

2.4.4 NODEMON

Nodemon is a development tool that automatically restarts the Node.js server whenever file changes are detected. I used Nodemon to improve productivity during development, as it removed the need to manually restart the server after every code update. This made backend development faster and more efficient.

2.4.5 THUNDER CLIENT

Thunder Client is a lightweight REST API client extension inside VS Code. It works similarly to Postman but directly within the editor. I used it for quick API testing without switching windows. It supports features like request history, response formatting, and headers/authorization testing.

CHAPTER 3

TASK PERFORMED

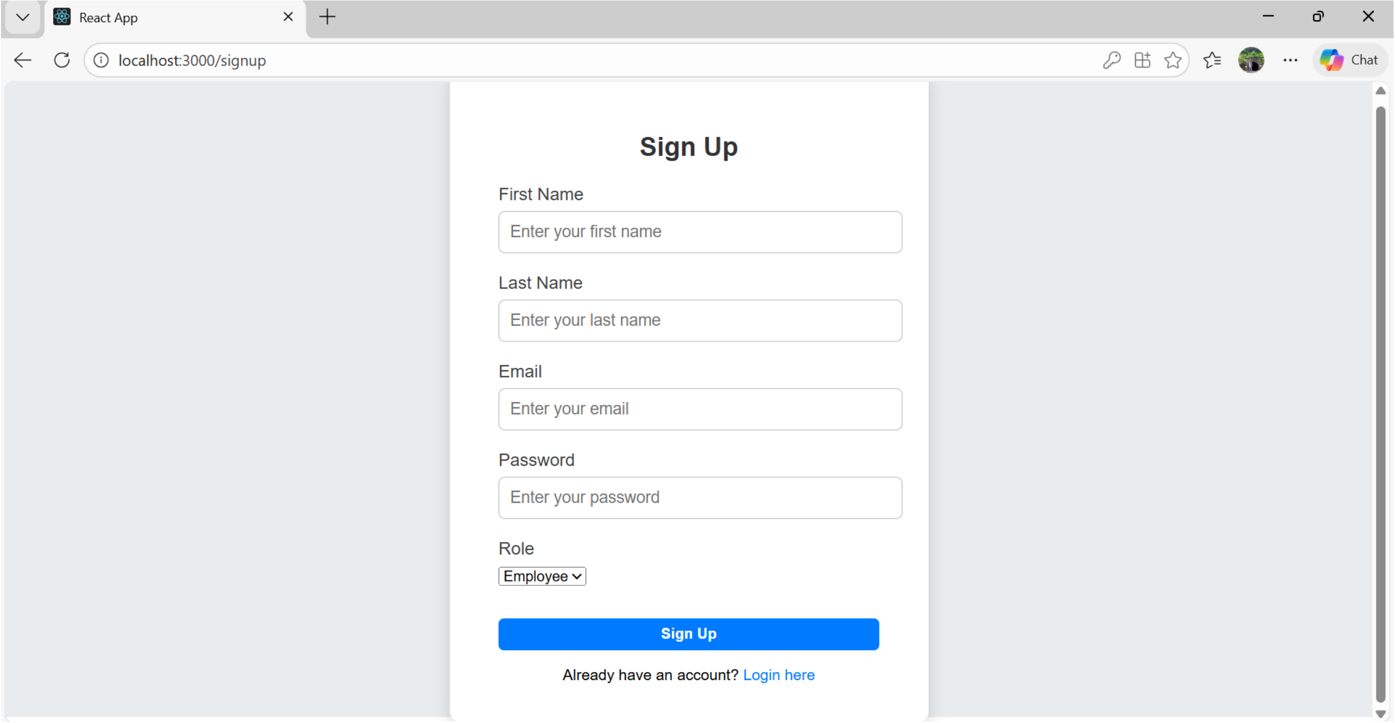
3.1 Employee Management System

Employee Management Systems are modern enterprise applications designed to manage employee-related operations such as task assignment, attendance tracking, and leave management through a centralized digital platform. With advancements in web technologies, traditional manual processes have been replaced by automated, role-based systems that improve efficiency and accuracy. The integration of frontend interfaces, backend servers, and databases enables real-time management of workforce activities.

In this system, software components handle key operations including user authentication, role-based access control, task creation and assignment, punch-in and punch-out attendance tracking, task status updates, and leave application processing. Employees can log in to view assigned tasks, mark attendance, update task completion status, and apply for leave, while administrators can create and prioritize tasks, monitor employee activities, and approve or reject leave requests. All system operations are processed through structured backend logic to ensure data consistency and security.

The application uses a React-based frontend for an interactive user interface and a Spring Boot backend to manage business logic and API communication. RESTful APIs using HTTP and JSON enable seamless interaction between the frontend and backend, while a MySQL database stores employee details, task records, attendance logs, and leave information. This interconnected architecture allows users to access the system from any device and provides a scalable and reliable solution for effective employee management.

3.2 Project Screen Shots:



Sign Up

First Name

Enter your first name

Last Name

Enter your last name

Email

Enter your email

Password

Enter your password

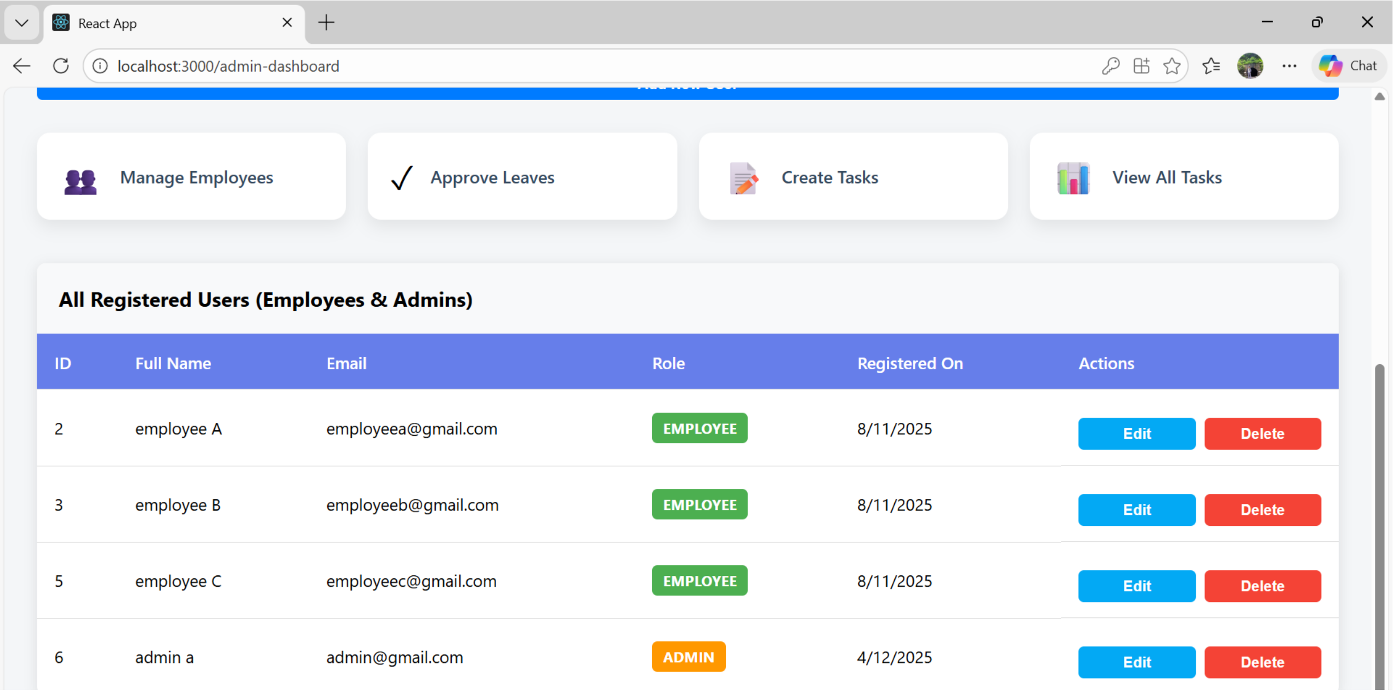
Role

Employee

Already have an account? Login here

Figure 3.2(a) Login / Register form

The login and signup section of the Employee Management System provides a clean and user-friendly interface that allows employees and administrators to securely access the application. The design includes a centered authentication card with clearly labeled input fields for email and password, ensuring simplicity and ease of use. Role-based authentication ensures that users are redirected to their respective dashboards after successful login. The visually appealing layout enhances user experience, while the prominent login and signup buttons enable quick and secure access. Overall, this section ensures safe entry, smooth navigation, and a professional first interaction with the system.

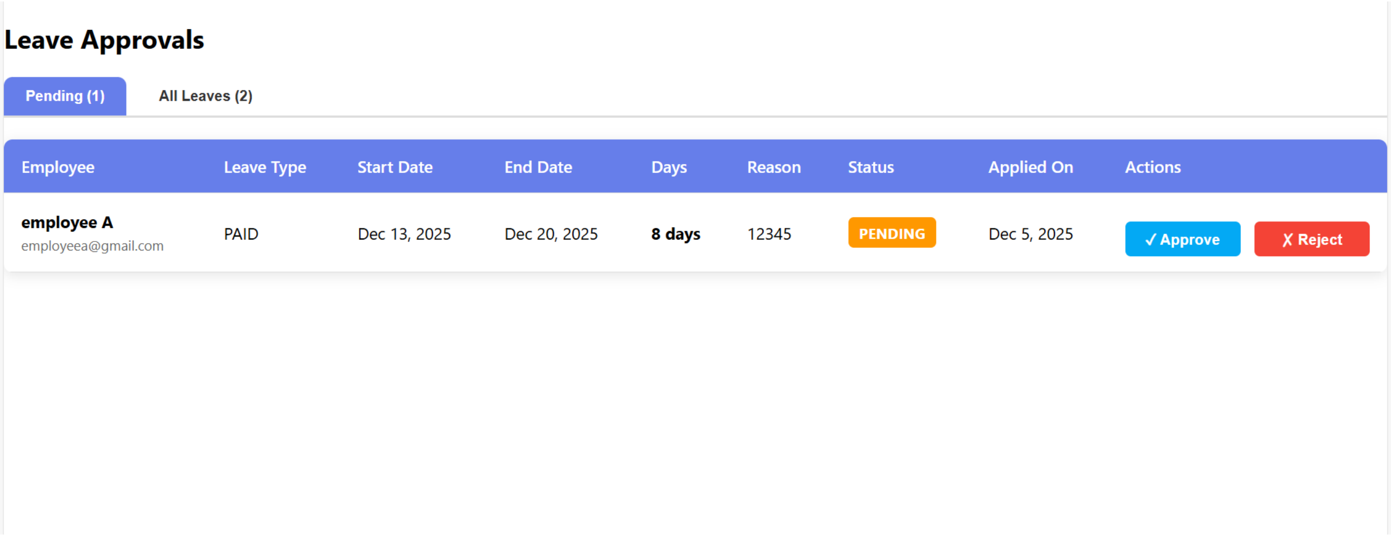


All Registered Users (Employees & Admins)

ID	Full Name	Email	Role	Registered On	Actions
2	employee A	employeea@gmail.com	EMPLOYEE	8/11/2025	Edit Delete
3	employee B	employeeb@gmail.com	EMPLOYEE	8/11/2025	Edit Delete
5	employee C	employeec@gmail.com	EMPLOYEE	8/11/2025	Edit Delete
6	admin a	admin@gmail.com	ADMIN	4/12/2025	Edit Delete

Figure 3.2(b) Admin Dashboard

The admin dashboard provides a clean and organized management interface that enables administrators to efficiently control all employee-related operations from a single screen. It includes quick-access options such as **Add New User**, **Approve Leave Requests**, **Create Task**, and **View All Tasks**, allowing smooth handling of daily administrative activities. The dashboard also displays a centralized list of all registered employees along with **edit and delete options**, making it easy to update employee details or remove users when required. Overall, the interface is designed to ensure simplicity, fast navigation, and effective system management within the Employee Management System.

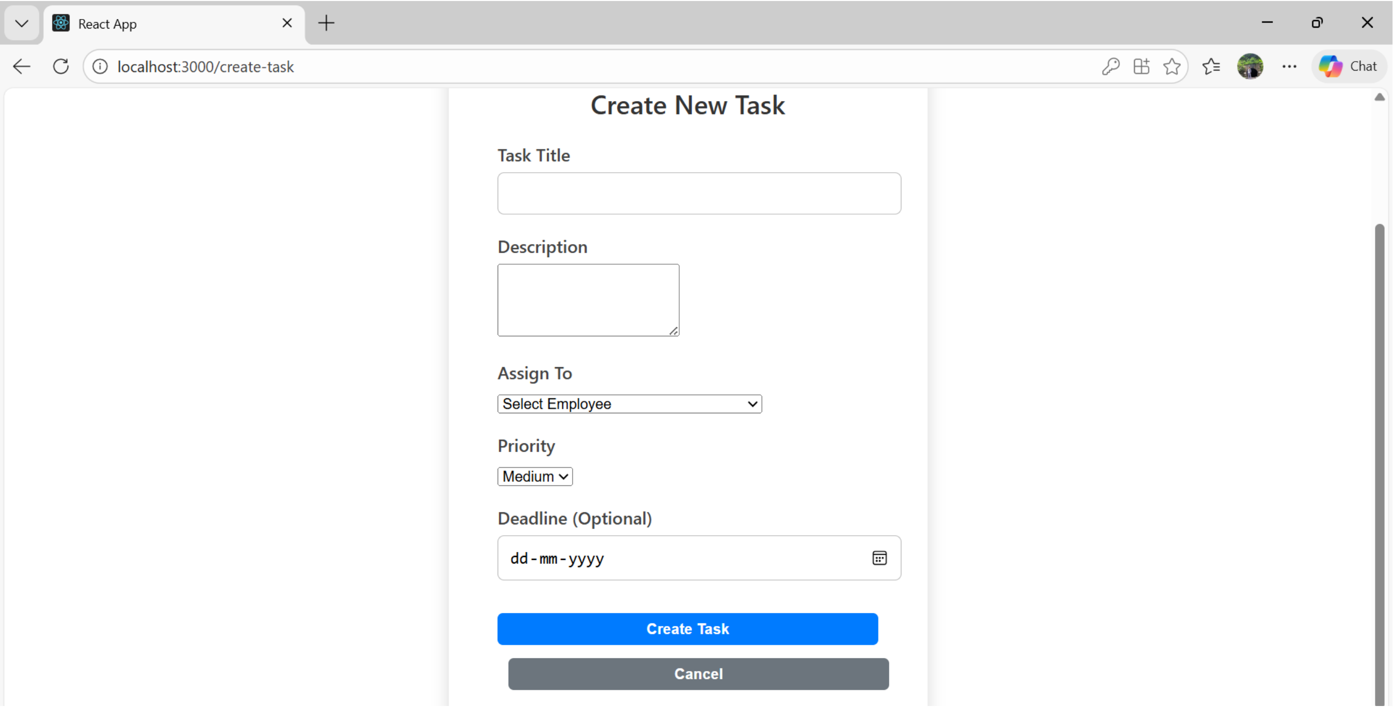


Leave Approvals

Employee	Leave Type	Start Date	End Date	Days	Reason	Status	Applied On	Actions
employee A employeea@gmail.com	PAID	Dec 13, 2025	Dec 20, 2025	**8 days**	12345	PENDING	Dec 5, 2025	✓ Approve ✗ Reject

Figure 3.2(c) Leave-Approvals Dashboard

The leave approval dashboard provides a clear and well-structured interface for managing employee leave requests efficiently. The panel displays a list of leave applications along with complete employee details such as name, role, leave type, and duration, while clearly indicating the current leave status as **Pending** or **Approved**. Action buttons like **Approve** and **Reject** are provided for each request, allowing the admin to take immediate decisions. When no leave requests are available, the dashboard displays an appropriate message to inform the administrator. This layout ensures a smooth and intuitive leave management experience, enabling quick review, decision-making, and real-time status updates within the system.



Create New Task

Task Title

Description

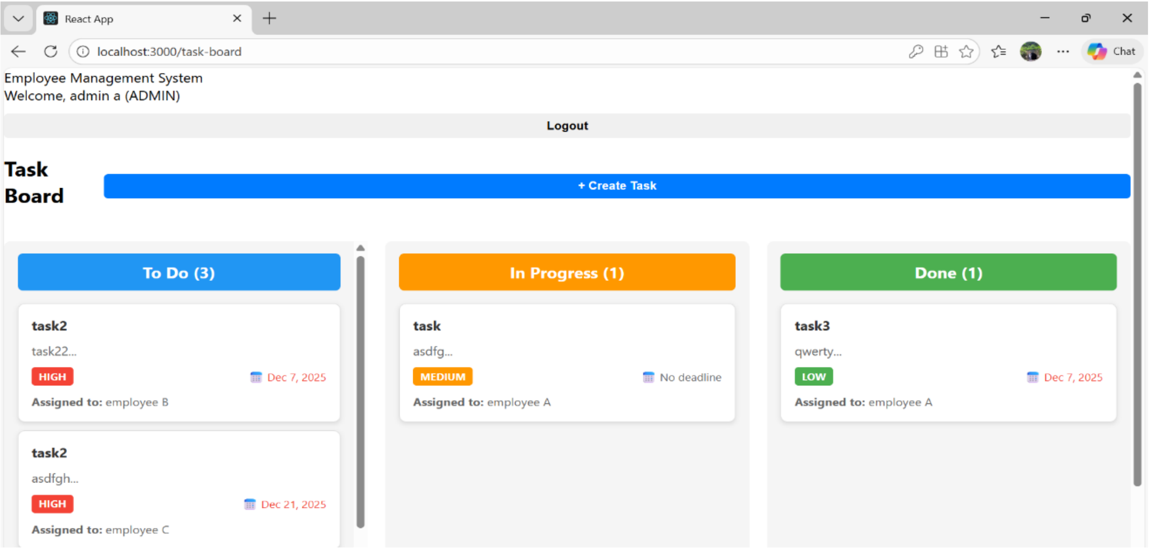
Assign To
Select Employee

Priority
Medium

Deadline (Optional)
dd-mm-yyyy

Figure 3.2(d) Create-Task Dashboard

The create task dashboard provides a clean and user-friendly interface that allows administrators to efficiently assign tasks to employees. It includes clearly defined input fields such as **task title**, **task description**, **assign to**, **priority level**, and **deadline selection using a calendar view**, ensuring accurate task creation. Action buttons like **Create Task** and **Cancel** enable the admin to either submit the task or discard changes with ease. This structured layout simplifies task assignment, improves clarity, and ensures smooth task management within the Employee Management System.



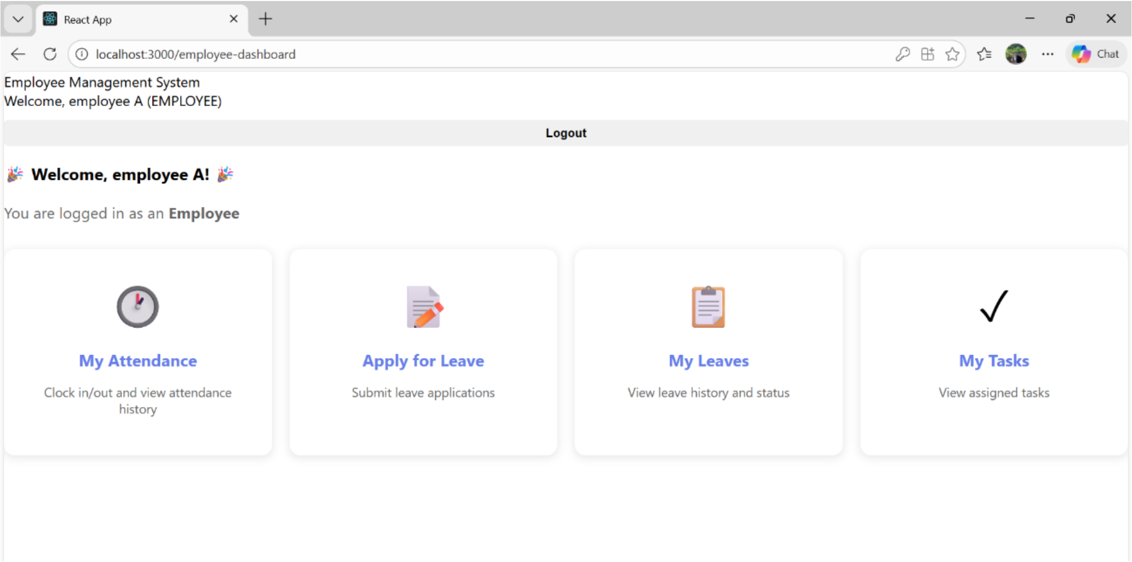
Employee Management System
Welcome, admin a (ADMIN)

Task Board

To Do (3)	In Progress (1)	Done (1)
task2 task22... HIGH Dec 7, 2025 Assigned to: employee B	task asdfg... MEDIUM No deadline Assigned to: employee A	task3 qwerty... LOW Dec 7, 2025 Assigned to: employee A
task2 asdfgh... HIGH Dec 21, 2025 Assigned to: employee C		

Figure 3.2(e) All-Task-View Dashboard

The view all tasks dashboard provides a clear and organized overview of all tasks within the system, allowing administrators to easily monitor task progress and assignments. Tasks are categorized into separate sections such as **To-Do**, **In Progress**, and **Done**, enabling quick status tracking at a glance. Each task entry displays relevant details, including the task title, priority, and the employee to whom it is assigned, ensuring full visibility of responsibilities. This dashboard offers an intuitive way to manage workloads, track progress, and maintain effective task coordination across the organization.



Employee Management System
Welcome, employee A (EMPLOYEE)

Logout

🎉 Welcome, employee A! 🎉

You are logged in as an **Employee**

My Attendance
Clock in/out and view attendance history

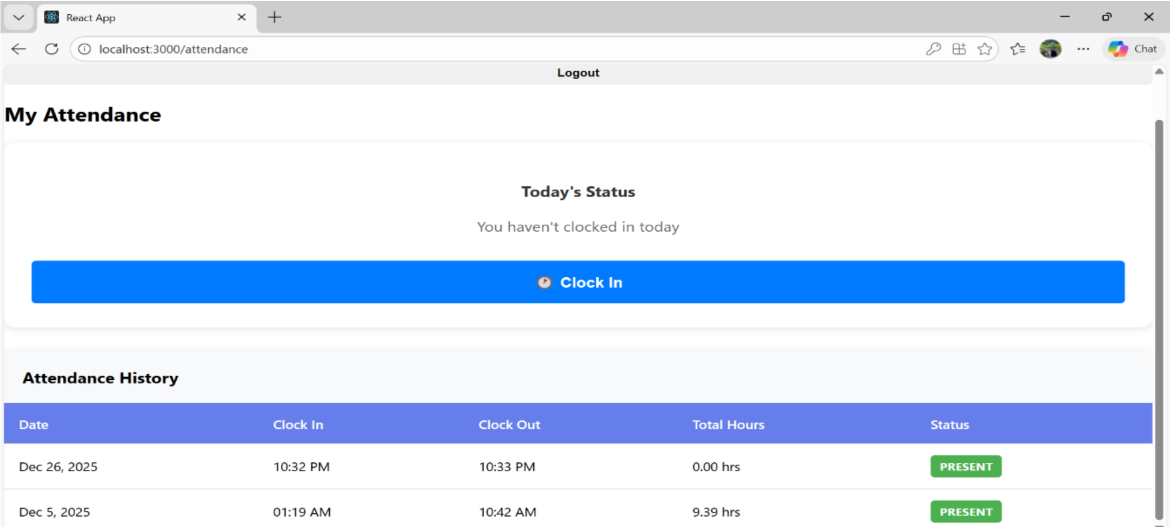
Apply for Leave
Submit leave applications

My Leaves
View leave history and status

✓ **My Tasks**
View assigned tasks

Figure 3.2(f) Employee-Dashboard

The employee dashboard provides a personalized and user-friendly interface that welcomes the employee by name and allows secure access to system features. It includes a visible **logout** option along with dedicated sections for **My Attendance**, **Apply for Leave**, **My Leaves**, and **My Tasks**, enabling employees to manage their daily activities from a single screen. This organized layout ensures easy navigation, real-time access to personal information, and a smooth user experience within the Employee Management System.



My Attendance

Today's Status

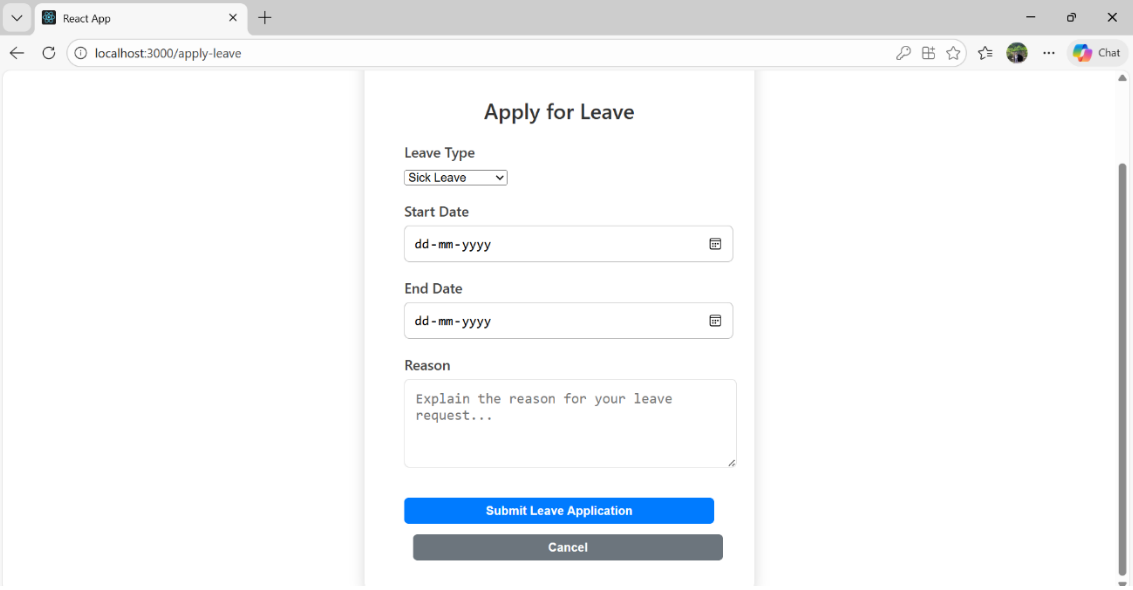
You haven't clocked in today

Attendance History

Date	Clock In	Clock Out	Total Hours	Status
Dec 26, 2025	10:32 PM	10:33 PM	0.00 hrs	PRESENT
Dec 5, 2025	01:19 AM	10:42 AM	9.39 hrs	PRESENT

Figure 3.2(g) My-Attendance Dashboard

The **My Attendance** dashboard provides a comprehensive and interactive interface that allows employees to manage and track their daily attendance efficiently. It includes clearly visible **Clock-In** and **Clock-Out** options, enabling employees to record their working hours accurately at the start and end of each work session. Along with real-time attendance actions, the dashboard displays a detailed history of the employee's attendance records, including **date-wise in-time and out-time entries**, total **hours worked**, and overall attendance summaries. This historical view helps employees review their work patterns and ensures transparency in time tracking. The structured layout and clear presentation of attendance data make it easy to monitor daily activity, reduce discrepancies, and maintain accurate attendance records within the Employee Management System.



Apply for Leave

Leave Type
Sick Leave

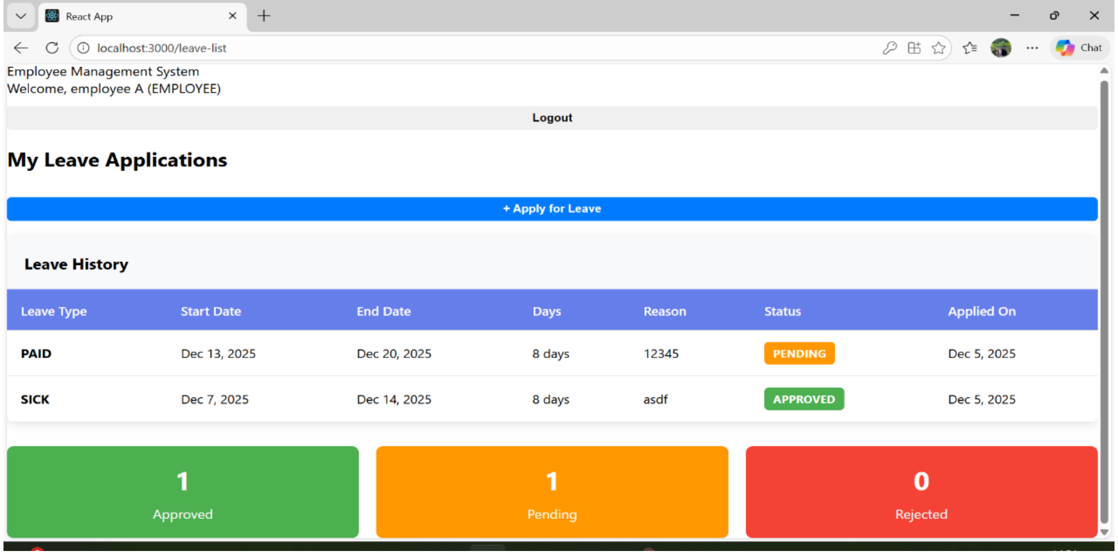
Start Date
dd-mm-yyyy

End Date
dd-mm-yyyy

Reason
Explain the reason for your leave request...

Figure 3.2(h) Apply-Leave Dashboard

The **Leave Apply** dashboard offers a simple and intuitive interface that allows employees to submit leave requests with ease. It includes calendar-based selection options for **start date** and **end date**, enabling accurate specification of the leave duration. The dashboard also provides a dropdown to choose the **leave type**, such as sick leave, maternity leave, paternity leave, or other applicable categories, along with a text area to enter the **reason for leave**. A clearly visible submit option allows employees to apply for leave, while the structured layout ensures clarity, reduces input errors, and supports smooth leave request processing within the Employee Management System.



Employee Management System
Welcome, employee A (EMPLOYEE)

Logout

My Leave Applications

+ Apply for Leave

Leave History

Leave Type	Start Date	End Date	Days	Reason	Status	Applied On
PAID	Dec 13, 2025	Dec 20, 2025	8 days	12345	PENDING	Dec 5, 2025
SICK	Dec 7, 2025	Dec 14, 2025	8 days	asdf	APPROVED	Dec 5, 2025

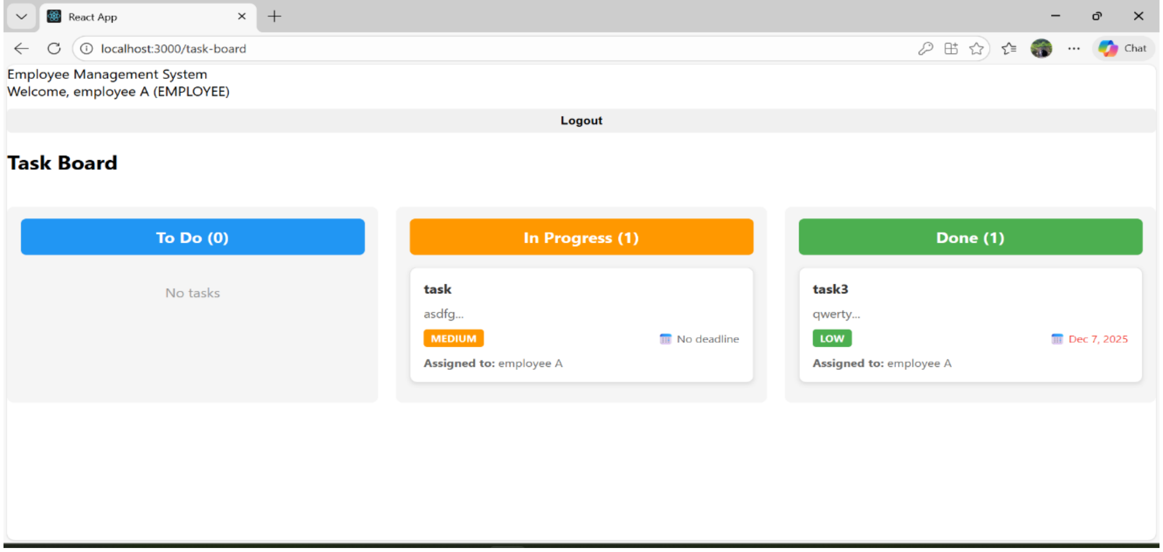
1 Approved

1 Pending

0 Rejected

Figure 3.2(i) Leave-List Dashboard.

The **Leave List** dashboard provides a detailed and well-organized view of an employee's complete leave history, enabling clear tracking of all leave-related activities. The interface displays leave requests categorized by **Pending**, **Approved**, and **Rejected** statuses, allowing employees to quickly understand the current state of their applications. Each leave entry includes comprehensive details such as leave type, start and end dates, total number of leave days, reason for leave, and approval status, ensuring full transparency. This dashboard helps employees monitor past and current leave requests, stay informed about approval decisions, and maintain accurate personal leave records, making leave management simple, clear, and efficient within the Employee Management System.



Employee Management System
Welcome, employee A (EMPLOYEE)

Logout

Task Board

To Do (0)

No tasks

In Progress (1)

task
asdfg...
MEDIUM
No deadline
Assigned to: employee A

Done (1)

task3
qwerty...
LOW
Dec 7, 2025
Assigned to: employee A

Figure 3.2(j) Task-List Dashboard

The **Task List** dashboard provides a clear and organized view of all tasks assigned to the employee, helping them track their work efficiently. Tasks are categorized into **To-Do**, **In Progress**, and **Done** sections, allowing employees to easily understand their current workload and progress. Each task entry displays key details such as task title, priority, and current status, enabling better task planning and timely completion. This structured dashboard ensures smooth task management, improves productivity, and gives employees a clear overview of their responsibilities within the Employee Management System.

CHAPTER 4

REFLECTION NOTES

4.1 What Is An Internship?

An internship is any carefully monitored work or service experience in which a student has intentional learning goals and reflects actively on what she or he is learning throughout the experience. Characteristics include:

- Duration of anywhere from a month to two years, but a typical experience usually lasts from three to six months.
- Generally a one-time experience.
- May be part-time or full-time.
- May be paid or non-paid.
- Internships may be part of an educational program and carefully monitored and evaluated for academic credit, or internships can be part of a learning plan that someone develops individually.
- An important element that distinguishes an internship from a short-term job or volunteer work is that an intentional “learning agenda” is structured into the experience.
- Learning activities common to most internships include learning objectives, observation, reflection, evaluation and assessment.
- An effort is made to establish a reasonable balance between the interns’ learning goals and the specific work an organization needs done.
- Internships promote academic, career and/or personal development.

4.2 How Do Internships Benefit Company?

4.2.1 Students bring new perspectives to old problems.

4.2.2 Visibility of your organization is increased on campus.

4.2.3 Quality candidates for temporary or seasonal positions and projects.

4.2.4 Freedom for professional staff to pursue more creative projects.

4.2.5 Flexible, cost-effective work force not requiring a long-term employer commitment.

4.2.6 Proven, cost-effective way to recruit and evaluate potential employees.

4.2.7 Our image in the community is enhanced as we contribute our expertise to the educational enterprise.

At the start of the internship it is recommended that the supervisor and intern create a list of learning goals the intern hopes to complete. The learning goals serve as the academic and professional roadmap for the intern's semester/time with your organization. This activity helps to clearly identify the intern's learning objectives and how the intern plans to accomplish them. Creating these goals is also an opportune time to discuss work place requirements, intern responsibilities, and hours expected to complete (to earn credit). We ask our supervisors to take the time to meet with the interns during the first week of the internship to discuss and negotiate learning goals.

4.3 The Stages of an Internship

Some of the concerns and challenges interns face seem to occur in a predictable order. Each stage has its own obstacles and opportunities. There are concerns interns will have at each stage, and to some extent, those concerns must be resolved for them to move forward and continue learning and growing. However, the process of resolving the concerns is also a learning experience in and of itself. At each stage, there are important tasks that will help interns address the concerns. If these tasks go undone and the concerns are left unresolved, the intern can become stuck in a stage.

Rate of progress through the stages is affected by many factors, including the number of hours spent at the agency; previous internships or field experiences; our personality; the personal issues and levels of support we bring into the experience; the style of supervision and the nature of the work. The more we understand as a supervisor, the better the learning experience for the intern.

Anticipation

4.3.1 Beginning of internship.

4.3.2 Anxiety "What if" Interaction with many people

4.3.3 Level Task accomplishment low.

4.3.4 Definition of specific goals with consideration of skills needed to complete established goals.

4.3.5 Assumptions- need to examine and critique.

4.3.6 Development of good relationships with supervisor, co-workers, etc.

Disillusionment

Gap between expectations for the internship and what is actually occurring

4.3.7 Sometimes gaps small, others large.

4.3.8 “What if…” from previous stage has become “What’s wrong”

4.3.9 Possibility for tremendous for personal and professional growth by working through problems.

Confrontation

To get past Disillusionment stage intern must face and study what is happening

4.3.10 Failure to acknowledge and discuss problems can diminish learning experience, performance, etc.

4.3.11 May need to reevaluate goals- may be unrealistic or opportunities may have changed.

4.3.12 As raised in disillusionment are resolved, task and morale accomplishments will raise

4.3.13 Must continue to confront and not get “frozen in the moment”.

4.3.14 Empowerment from knowing can grapple with problems effectively.

Competence

4.3.15 Morale high- trust in yourself

4.3.16 Transition into “professional” from “apprentice”

4.3.17 Good platform to demand more from self and assignment- increase work load, responsibility

Culmination

4.3.18 May face a variety of feelings at this stage

4.3.19 Pride in accomplishments, sadness in leaving

4.3.20 Need to address emotions, find avenue to express them.

4.4 The Value of an Internships

A primary objective of internships is “to provide students with an opportunity to test abilities and attitudes toward particular material or career possibilities for the future”. While classroom learning is important, it does not provide students with real work experience to prepare them for a career after graduation. Exposure to the workplace environment is needed to bridge the gap between learning in the classroom and application in the work environment and found that internships as a form of experiential learning give students “valuable opportunities to discover the professional world first-hand and to apply classroom knowledge to practice”. Internships benefit both the student in the classroom and the young professional in the workplace post- graduation. An internship may even help students decide if their intended career is one they actually want to pursue. Furthermore, “virtually all discussions of internship programs mention clarification of career choice as a major function and justification”. Helping students clarify and assess their career goals is a primary goal of internship programs.

In addition to providing students with real workplace experience, internships enhance student development. According to the Conference on Undergraduate Internships, “vocational development, intellectual development, personal growth, and community service” are the four major functions of internships. Students are able to gain experience in public speaking and voicing opinions to fellow co-worker. They are given the opportunity to observe the work of others and learn from their own job responsibilities. By immersing themselves in their internship, students gain a sneak peek into what their future careers in a particular field look like. Working in these real-world businesses and organizations gives interns “the career-related, job acquisition, interpersonal, and communication skills” that classroom learning cannot.

While classroom learning and internship learning have their differences, they are also intertwined and relatable. Much of what is learned in the classroom setting can be applied to workplace situations, tasks, and activities. Once students are able to link theory from the classroom to practice in the workplace, they are more likely to be successful in both areas. Students with internship experience “appear to bring more to the classroom, link the importance of classroom discussions to practice, and demonstrate a significant maturation process lacking in non-internship students”. Thus, internships help to create well-rounded students and professionals who enrich both the work environment and the classroom.

In addition to linking theory to practice, another benefit of an internship “may be a change in the significance students place on traits that are important to obtain professional employment” Before starting an internship, students have preconceived notions of what the field will be like and what character traits will be important to employers. Gaining that experience and workplace exposure allows the student to determine what traits and characteristics are actually valuable when it comes to being successful in a job or internship. Internship experiences destroy the fantasy of preconceived notions and allow interns to learn more about “work and working with others, their own strengths and weaknesses as future employees, as well as the nature of their potential future careers”. Once these personal, academic, and professional discoveries are made, they give interns the experiences and knowledge they need to succeed in class, at work, and at play.

4.5 The Importance of Reflection

Although internships are supposed to help students apply what they have learned in the classroom to what they are doing in the workplace and vice versa, these connections are often not made without reflecting on the experience and asserts that internship reflection “enhances a learner’s experience through a linkage of education, work, and personal development”. Through reflection, students gain an appreciation for the experience and self-confidence in their abilities. Having a mentor or adviser to assist in internship reflection can help the student see his or her experiences from another perspective, learners must have “the opportunity to reflect on and observe experiences from many perspectives”. Encouraging students to examine issues that arose in the internship from a variety of different perspectives can help students better understand the actions, feelings, and reactions that occurred. In addition, this knowledge can help students reflect on better ways to handle future situations in the workplace. Allowing students to find their own answers and make their own discoveries is crucial, but advisers can play a pivotal role in sparking student reflection by asking reflective questions and providing encouragement.

The experiences of the internship can also cause students to become more reflective in other aspects of their lives, such as their duties as organizational leaders. Students are more likely to see reflection as a helpful tool once it assists them in their internship. If advisers are successful in their efforts to help students reflect, students will be encouraged by the results. It introduces them to a mind-set of continuous learning and teaches them how to ask reflective questions. These skills will add to their abilities as leaders and may even encourage others in their groups and organizations to use reflection in their daily lives.

4.6 Ways to Aid Students with Internship Reflection

This section will offer specific strategies that advisers can use to promote student reflection. Since students often get academic credit for participating in internships, students tend to express interest in participating in internships during meetings with their academic advisers. Advisers should note in students' files when they are participating in internships so that they prepare for discussion and reflection when they return to campus. Although a faculty member usually serves as the instructor for internship courses, advisers still have a role to play. For example, advisers can help students establish goals and expectations about the internship and maintained that the adviser and the student "should jointly develop the goals for the internship and state them clearly so that everyone understands individual responsibilities."

Once students have completed their internships, advisers should encourage them to make an appointment to discuss their experiences or be prepared to discuss the internship when the students comes in for their regular advising appointment. In planning for the meeting, the adviser can remember these three principles from the experiential learning literature:

1. Learning should be relearning where students' ideas are brought out, examined, and mixed with new refined ideas.
2. Learning involves the whole person, i.e., thinking, feeling, perceiving, and behaving.
3. Learning is a process of combining new experiences with old experiences and vice versa.

These principles can be implemented during reflection to encourage the student to examine certain situations and think outside the box. Advisers would benefit from using these ideas to help students realize the extent of "the experiences they had and the potential impact on their futures".

4.7 Create a List of Reflective Questions

Advisers can promote student reflection by creating a list of questions to ask them about their internship experience. These questions should make students take the time to think about and deeply consider their experience. Examples of effective questions include, what were our first thoughts and observations when starting the internship? How was our internship experience different from what you expected? What did we learn in your internship that you can now apply to the classroom setting or to your daily life? What connections have we made between theory and practice? How have our career plans changed due to your internship experience? What skills did we acquire during your internship that we will be able to highlight in your resume, cover letters, and/or interviews? Advisers should discuss with students how they can use what they learned from their internship “to help accomplish their future life and career goals”.

Listen for Inconsistencies

Advisers should point out overlooked interpretations to the student but avoid offering their own interpretations. Students will often brush off something as being insignificant when there may be more to the matter that needs to be discussed. For example, a student may view criticism from a supervisor as evidence that their work or project is a failure. In reality, the supervisor liked the overall idea of the project but can see that the student needs to work on time management to improve the overall outcome. Advisers should resist offering their interpretation of the event and instead ask questions that allow students to explore it further without bias. This allows the students to make discoveries themselves without limiting their scope or point of view.

Encourage Self-Awareness

Advisers should encourage students to be aware of feelings and how those feelings have affected the internship experience, as “emotional sharing is positively related to both learning and mentoring”. Students need to be open and honest in terms of their emotional experiences with others, so encouraging emotional awareness will benefit reflection and understanding. Advisers can encourage awareness by prompting students to recall their initial thoughts and feelings during certain times in the internship. An event may have occurred that sparked an emotional reaction beyond what the student realized. This type of reflection will help the student to provide the necessary information for the adviser to “respond appropriately and give positive and constructive feedback”.

Ask Students to Share Artefacts from the Internship

Advisers should ask students to bring in to the appointment photographs, journals, or any other artefact’s they would like to share from their internship. This will keep advisers abreast of what is happening in the field and position them to effectively advise future students about what they can expect in their internships. It will also provide students the opportunity to further reflect on the internship.

Host an Internship Panel

Advisers can both promote reflection by student interns when they return to campus and ignite Interest about internships among younger students by hosting a panel discussion event. It would allow students who have participated in internships to share their experiences with their fellow students. This kind of collaborative learning would lead to group reflection and hopefully encourage other students to participate in an internship.

4.8 Benefits & Advantages of Internship

1. Gain work experience and transferable skills

Students all have their educational experience in common. What stands out to employers is those students who ALSO have work experiences by the time they graduate. This automatically makes them more marketable; as they may require less training and are assumed to be able to handle more responsibilities. In addition to the specialized skills of your field, transferable skills are generally required at any job, e.g. communication/interpersonal skills, computer proficiency, and team work.

2. Possibly earn course credit

Some degree programs incorporate a work opportunity into a requirement for graduation by giving course credit for the work experience. Other programs have an internship as an elective course for credit. By contacting our academic advisor for more information because earning credit varies by departments.

3. Be able to experience a prospective carrier path

Most students enter college with an idea of a major or career path and most students end up changing their minds. Exploring is a very important part of the academic process, and gaining a work experience is a great way for students to acquaint themselves with a field they are looking to learn about. By the time of graduation, the students have confidence that the degrees they are receiving are the right ones for them.

4. Gain practical experience by applying methods and theories learned in classes

Many people learn best by being hands on. But everyone can benefit from seeing the things that they have been learning in class, put to action; whether it's in a chemistry research lab, a marketing development meeting, or a substance abuse counselling session.

5. Network professionals in our field, for references and future job opportunities

It's all about who we know. As a student intern, we are surrounded by professionals in the industry that we are seeking access to. It's more than just about getting a grade, earning credit, or making money. This is an opportunity to learn from everyone around you, ask questions, and impress them with your eagerness. These people can be your future colleagues or can be the connection to your first job.

6. Develop new skills and refine others

Learning our strengths and weaknesses by creating learning objectives and receiving feedback from your supervisor. This is a unique learning opportunity that we may never have again as a working adult. Embrace the mistakes that we'll make and the many things that we won't know. Ask questions, observe, and take risks.

7. Gain confidence in your abilities

Practice makes perfect. If we've learned about a specific technique in the classroom, we're able to test it out in the world of work. Then, we'll be much more equipped with the technique.