



## streamline the workflow, your way.

Empower your workforce, simplify employee management and transform your business, with ease.



**IT314 - GROUP 1**

## ABOUT THE TEAM

Group Leader

**Sanket Doshi**

202001008

Front-end Engg.

**Sanket Doshi**

202001008

**Arnish Satasiya**

202001031

**Darshan Popat**

202001034

**Aditya Patel**

202001034

Documentation

**Saurav Kapadiya**

202001033

**Malav Rohit**

202001056

**Aditya Joshi**

202001059

Back-end Engg.

**Shrey Pobaru**

202001021

**Krupal Shah**

202001057

Testing

**Tarang Soni**

202001044

**Malav Rohit**  
(GUI testing)

202001056

UI Design, Prototyping, PPT

**Malav Rohit**

202001056

### TIMELINE 🌟

3 months, Group project.

### TEAM'S ROLE 💬

Ideation, Product Research, Frontend and Backend Engg., Visual Design, Documentation.

## THE PROCESS

# our objectives



1.

## /Improve Efficiency\

To increase overall operational efficiency by eliminating manual paperwork, and automating repetitive operations by improving HR procedures including onboarding, offboarding, and employee data management.

2.

## / Enhance HR Analytics \

Deep insights into employee performance, attendance patterns, and other crucial HR analytics features can be provided by the system to utilize these data to design plans for developing talent, assess employee potential, and more.

3.

## / Foster Employee Self-Service \

By managing leave requests, updating personal information, access salary status, and keep track of their performance, this option decreases reliance on HR staff, boosts employee autonomy, and enhances overall satisfaction.

# our milestones



1

2

3

4

5

## Database and Backend Setup

- Establish the database structure & design for efficient storage and retrieval of employee information, attendance and leave records, and other organizational data.
- Create backend models and APIs to handle CRUD operations for employee management, department management, and user authentication.

## Admin Dashboard and Departments

- Develop an admin dashboard that provides a comprehensive view of all departments and their respective employees.
- Implement functionality for the admin to manage departments, including adding, editing, and deleting department records.

## Employee Dashboard

- Create features that allow the admin to view and manage employee records, including personal information, contact details, and employment history.
- Implement user roles and access control, ensuring that admins have full access to all departments and employees, while managers have access to their respective department's information.

## Manager Dashboard and Leave Application System

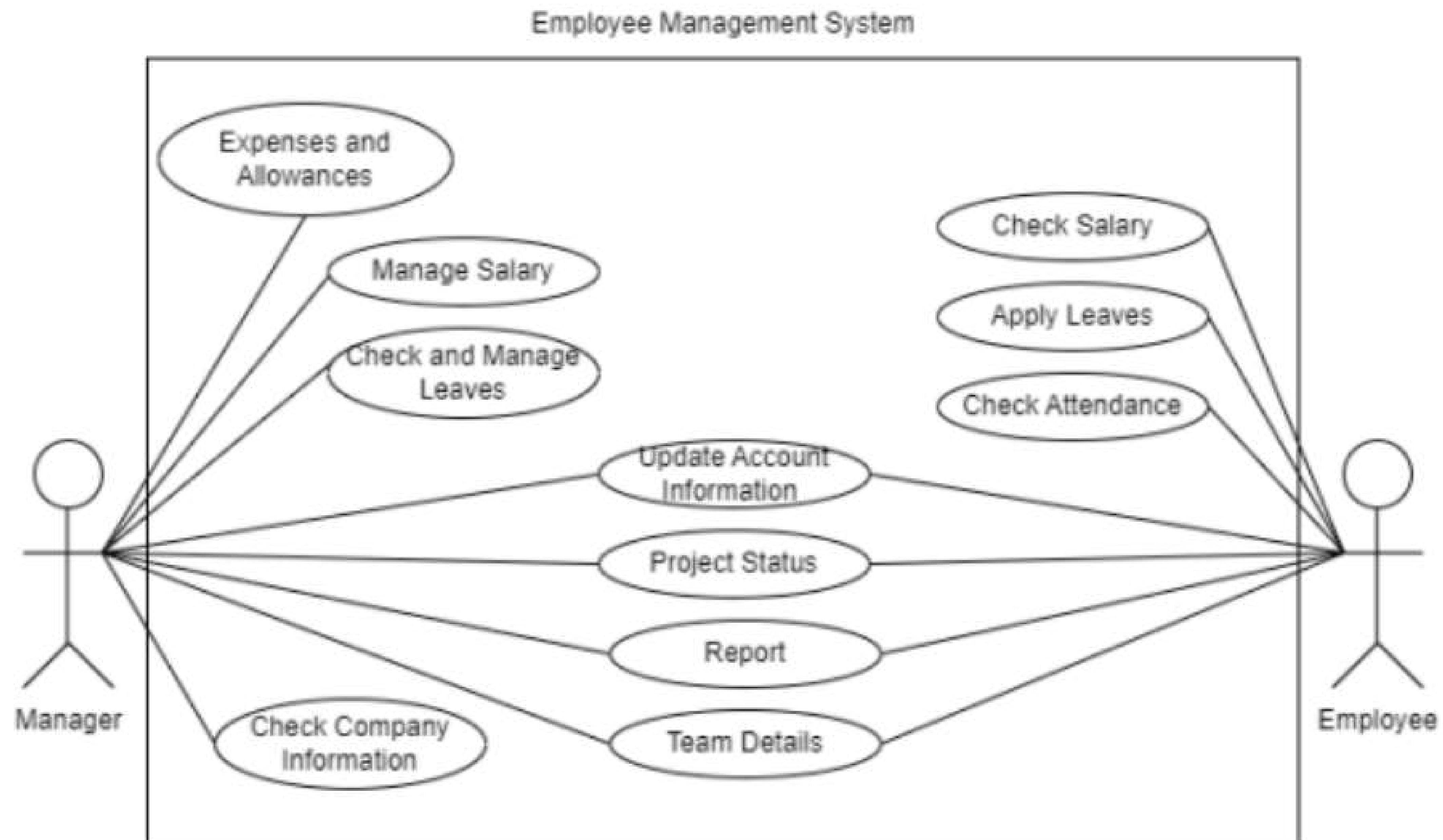
- Design a dedicated manager dashboard that provides managers with a sophisticated view of their department's employees and their leave statuses.
- Develop a leave application system that enables employees to submit leave requests and allows managers to review and approve/reject those requests.

## Front-end + User Interface and Connection Setup

- Design and implement user-friendly interfaces for all screens, ensuring a smooth and intuitive user experience.
- Establish secure connections between the front-end and back-end systems to facilitate seamless data transfer and real-time updates.

THE PROCESS

# Use Case Diagram



THE PROCESS

# Lessons Learnt

Lesson Learned #1



## Emphasize Teamwork

Collaborating effectively and fostering good communication within the team is crucial for the success of a collaborative and large-scale project like this. Encouraging teamwork and establishing clear roles and responsibilities help ensure smooth progress and timely completion.

Lesson Learned #1



## Selecting Technologies

Choosing the right technologies plays a vital role in simplifying development and improving overall efficiency, such as deciding between Node.js or Django for the backend, and Flutter or React for the front-end, and Proper research and evaluation of technologies beforehand can save time and effort in the long run.

Lesson Learned #1



## Documentation is Crucial

It is crucial to read and comprehend the documentation of the technologies being utilised for development as well as the project itself. A development team's awareness of the technologies and their capabilities is ensured by proper documentation, which also facilitates knowledge transfer and troubleshooting.

Lesson Learned #1



## Flexibility in Requirements

We recognized the importance of flexibility in project requirements being key. Considering that as the project progresses, new ideas may emerge, and certain pre-defined features may prove infeasible or unnecessary to implement, so by being open to adapting and adjusting requirements we led the project to better outcomes.

Lesson Learned #1



## Learn All The Way

With the ever-evolving technology, it's essential to maintain a proper mindset of continuous learning and teaching ourselves new things by keeping up with the latest updates, features, and best practices in the chosen technologies which helped us in uncovering new possibilities & ensures optimal utilization.

# our achievements 🎾

1.

## Effective Team Collaboration

- We successfully learned and implemented effective teamwork practices, allowing the development team to collaborate seamlessly, share knowledge, and work towards a common goal.

2.

## Wholesome Development Experience

- Completed the development process with a well-rounded experience, gaining insights into various aspects of software development, including requirement analysis, design, implementation, testing, and deployment.

3.

## Maximum Feature Implemented

- Our capacity to comprehend, prioritise, and deliver on the required functionality was demonstrated by the fact that we were able to successfully incorporate a considerable number of the features listed in the Software Requirements Specification (SRS).

4.

## Learning New Technologies

- Successfully acquired advanced proficiency in new technologies such as Flutter and MongoDB, while expanding the team's skillset in understanding the Design phase and how important Visual Design ultimately helps the product with user engagement and enabling the development of robust and modern solution.

## THE PROCESS

# Mistakes we Made

### Mistake #1

#### Miscalculating Size of Project and Effort Analysis:

This error occurred due to underestimation of the time, resources, and effort needed to finish particular tasks or deliver particular features. The team consequently encountered unforeseen delays, an increase in work, and probable scope creep.

### Mistake #2

#### Naming Convention

We had our share of inconsistent/unclear naming conventions used in the codebase, leading to confusion and decreased code readability. So we implemented a standardized naming convention that followed best practices, making the codebase more organized and easier to understand for the development team.

### Mistake #3

#### Backend-Frontend Timezone Discrepancy

A discrepancy in timezones between the backend (located in Singapore) and the frontend caused issues with timestamp handling and data synchronization. So we tried to ensure consistent time zone management between the backend and frontend, either by aligning both to a specific timezone or implementing proper timezone conversion mechanisms to maintain data integrity across the system.

## FEEDBACK

# Rating Software Artifacts ✨

**Requirement Document (SRS)**  **3.5/5**

**Design Document**  **4/5**

**Coding Components**  **4/5**

**Testing**  **3/5**

**Project Planning and Scheduling**  **2.5/5**

**Overall Project Rating** **8/10**