

# Software Engineering

---

## Deliverable 3

### Employee Payroll and Leave Management System (ASTRA-HR)

## FINAL REPORT

**Team Name:**

**Triovate**

**Team Members:**

Abdullah Aslam

22i 0784

Abdur-Raheem Shaikh

22i 0777

Afrah Syed

22i 1008

## Table of Contents

<b>1. Introduction .....</b>	<b>4</b>
<b>2. General Description .....</b>	<b>5</b>
<b>3. Functional Requirements .....</b>	<b>6</b>
<b>4. Non-Functional Requirements .....</b>	<b>7</b>
<b>5. User Stories.....</b>	<b>7</b>
<b>6. Product Backlog.....</b>	<b>13</b>
<b>7. Sprint Backlog.....</b>	<b>14</b>
<b>8. Project Plan .....</b>	<b>15</b>
<b>9. Architecture Diagram.....</b>	<b>16</b>
<b>10. Design (all sprint 3 items).....</b>	<b>19</b>
<b>11. Actual implementation screenshots.....</b>	<b>19</b>
<b>12. Product Burn down chart for the project.....</b>	<b>21</b>
<b>13. Trello board screen shots .....</b>	<b>23</b>
<b>14. Testcases -Black box .....</b>	<b>23</b>
<b>15. Testcases -White box .....</b>	<b>25</b>
<b>16. Work Division between group members .....</b>	<b>26</b>
<b>17. Lesson learnt by group .....</b>	<b>26</b>

# 1. Introduction

## 1.1 Purpose

The purpose of the Employee Payroll and Leave Management System is to automate and streamline payroll processing, leave management, and employee attendance tracking. It ensures accuracy in salary calculations, tax deductions, and leave policies, reducing manual effort and administrative workload.

## 1.2 Product Scope

The system is designed for organizations to manage payroll, leave applications, and employee records efficiently. It provides role-based access for employees, managers, and HR personnel, enabling self-service leave requests, payroll summaries, and policy configurations. The system aims to enhance transparency, improve decision-making through reporting, and ensure compliance with company policies.

## 1.3 Glossary

- **Payroll** – The process of calculating and distributing salaries, including tax deductions and bonuses.
- **Leave Balance** – The number of remaining leave days available for an employee within a specified period.
- **Tax Deduction** – The process of deducting applicable taxes from an employee's salary.
- **Payroll Summary** – A detailed report of an employee's salary, including earnings, deductions, and net payable amount.
- **Leave Policy** – Organizational rules that define the leave entitlement, approval process, and restrictions for employees.

## 1.4 Overview

The **Employee Payroll and Leave Management System** is a comprehensive platform designed to streamline payroll processing and leave management within an organization. It provides an efficient and automated approach for employees, managers, and HR personnel to manage payroll, leave requests, and employee records.

The system ensures:

- Seamless payroll calculations, including salary distribution and tax deductions.
- Efficient leave management, allowing employees to apply for leave and track their leave balance.
- Role-based access for employees, managers, ensuring secure and structured workflows.

This system improves accuracy, reduces administrative effort, and ensures compliance with company policies.

---

## 2. General Description

### 2.1 Product Perspective

The **Employee Payroll and Leave Management System** is designed as an integrated platform that automates payroll processing, leave management, and employee attendance tracking. It operates within an organization's existing HR framework, ensuring seamless interaction with payroll systems, tax authorities, and company policies.

### 2.2 System Interfaces

The system interacts with:

- Internal HR databases for employee records, payroll, and leave data.
- Payroll processing modules for salary computation, tax deductions, and benefits allocation.
- Email and notification systems for leave approval alerts and payroll updates.

### 2.3 User Interfaces

The system provides:

- A web-based dashboard for HR personnel to manage payroll and leave requests.
- A self-service portal for employees to apply for leave and view salary details.
- A web-based interface for managers to approve/reject leave requests on the go.

### 2.4 Hardware Interfaces

The system requires:

- A standard computer or server for hosting the payroll and leave management application.
- Secure cloud storage or on-premises databases for storing employee records.

### 2.5 Software Interfaces

The system integrates with:

- Database Management Systems for data storage.
  - Payroll processing software for salary calculations and tax deductions.
  - Web frameworks (React.js) for an interactive front-end experience.
-

### 3. Functional Requirements

The functional requirements define the core functionalities of the system, ensuring smooth payroll processing and leave management.

REQ-1.1: The system shall allow employees and managers to register with their name, email, password, and role selection.

REQ-1.2: The system shall provide a secure login/logout functionality.

REQ-1.3: The system shall allow role-based access control, restricting features based on user roles (Employee or Manager).

REQ-1.4: Employees shall be able to view their payroll summary and salary details.

REQ-1.5: Employees shall be able to check their leave balance categorized by leave type (Annual, Sick, Casual).

REQ-1.6: Employees shall be able to apply for leave by selecting leave type, start date, and end date.

REQ-1.7: Employees shall receive notifications when their leave request is approved or rejected.

REQ-1.8: Managers shall be able to view and manage pending leave requests.

REQ-1.9: Managers shall be able to approve or reject leave applications.

REQ-1.10: Managers shall be able to configure leave policies and set leave limits for employees.

REQ-1.11: Managers shall be able to track employee leave history to analyse leave trends.

REQ-1.12: The system shall calculate, and process employee salaries based on attendance and leave records.

REQ-1.13: The system shall automatically compute tax deductions and other salary components.

REQ-1.14: The system shall provide real-time notifications for leave request status updates and payroll processing and ensure secure accounts.

---

## 4. Non-Functional Requirements

### 2.1 Product Requirements

NF-1.1: The system shall have a user-friendly and responsive UI for easy navigation.

NF-1.2: The system shall ensure data consistency and integrity to prevent payroll miscalculations.

NF-1.3: The system shall support multiple users concurrently without performance degradation.

NF-1.4: The system shall store payroll and leave records for a minimum of five years for auditing purposes.

### 2.2 Organizational Requirements

NF-2.1: The system shall comply with company HR policies and government tax regulations.

NF-2.2: The system shall be maintainable, allowing easy updates for tax policies and leave rules.

NF-2.3: The system shall provide role-based training for employees and managers to use the system effectively.

### 2.3 External Requirements

NF-3.1: The system shall comply with data protection laws to secure employee information.

NF-3.2: The system shall be accessible via web browsers and mobile devices.

---

## 5. User Stories

### 1. View Payroll Summary

- **User Role:** Employee
- **Goal:** View payroll summary.
- **Reason:** Helps employees check their salary breakdown, deductions, and tax calculations.

*As an employee, I want to view my payroll summary so that I can check my salary breakdown, deductions, and tax calculations.*

#### **Pre-conditions:**

- The employee is logged into the system.

- Payroll data is already processed for the given period.

**Post-conditions:**

- The payroll summary is displayed to the employee.
- 

**2. Apply for Leave**

- **User Role:** Employee
- **Goal:** Apply for leave.
- **Reason:** Enables employees to request time off without paperwork.

*As an employee, I want to apply for leave through the system so that I can request time off without paperwork.*

**Pre-conditions:**

- The employee is logged into the system.
- The employee has available leave balance.

**Post-conditions:**

- The leave request is recorded in the system.
  - A notification is sent to the manager for approval.
- 

**3. Approve/Reject Leave**

- **User Role:** Manager
- **Goal:** Review and approve/reject leave requests.
- **Reason:** Helps managers manage their team's availability effectively.

*As a manager, I want to review and approve/reject leave requests so that I can manage my team's availability effectively.*

**Pre-conditions:**

- The manager is logged into the system.
- There are pending leave requests to review.

**Post-conditions:**

- The employee is notified of the approval/rejection status.
  - The leave balance is updated if approved.
- 

**4. Track Employee Leave History**

- **User Role:** Manager
- **Goal:** Track employees' leave history.
- **Reason:** Helps in managing attendance effectively.

*As a manager, I want to track employees' leave history so that I can manage attendance effectively.*

**Pre-conditions:**

- The manager is logged into the system.
- The system has records of employee leave history.

**Post-conditions:**

- The manager can view and analyse employees' leave trends.
- 

## **5. Configure Leave Policies**

- **User Role:** Manager
- **Goal:** Configure leave policies.
- **Reason:** Ensures company leave rules are updated and applied correctly.

*As a manager, I want to configure leave policies so that company leave rules are updated and applied correctly.*

**Pre-conditions:**

- The manager is logged into the system.
- The system has an option to update leave configurations.

**Post-conditions:**

- Updated leave policies are saved and applied to all employees.
- 

## **6. Process Monthly Payroll**

- **User Role:** Manager
- **Goal:** Process employee payroll.
- **Reason:** Ensures accurate and timely salary processing, including tax deductions and bonuses.

*As a manager, I want to process employee payroll so that salaries, tax deductions, and bonuses are calculated accurately and on time.*

**Pre-conditions:**

- The manager is logged into the system.
- Employee attendance and leave records are up-to-date.



**Post-conditions:**

- Payroll is processed and saved in the system.
  - Employees can view their updated payroll summary.
- 

**7. Deduct Tax from Salaries**

- **User Role:** Manager
- **Goal:** Ensure tax compliance in payroll processing.
- **Reason:** Helps automate tax calculations and deductions.

*As a manager, I want the system to automatically deduct taxes from salaries so that payroll complies with tax regulations.*

**Pre-conditions:**

- Employee salary and tax rules must be configured in the system.

**Post-conditions:**

- Tax deductions are applied to employee salaries.
  - Employees can view tax deductions in their payroll summary.
- 

**8. Manage Employee Salary Structure**

- **User Role:** Manager
- **Goal:** Manage the salary structure of employees.
- **Reason:** Allows updating of salaries, bonuses, and deductions.

*As a manager, I want to manage the salary structure of employees so that I can update salaries, and deductions when needed.*

**Pre-conditions:**

- The manager must have salary modification permissions.

**Post-conditions:**

- The updated salary structure is saved and reflected in payroll calculations.
- 

**9. View Leave Balance**

- **User Role:** Employee
- **Goal:** Check remaining leave balance.
- **Reason:** Helps employees plan their future leaves accordingly.

*As an employee, I want to check my remaining leave balance so that I can plan my future leaves accordingly.*

**Pre-conditions:**

- The employee is logged into the system.
- Leave records are updated in the system.

**Post-conditions:**

- The leave balance is displayed to the employee.
  - Employees can plan their leaves based on the available balance.
- 

## **10. Submit Leave Application with Dates**

- **User Role:** Employee
  - **Goal:** Apply for leave by selecting start and end dates.
  - **Reason:** Allows employees to request specific dates for leave.
  - **Pre-conditions:**
    - The employee is logged into the system.
    - Leave balance must be available.
  - **Post-conditions:**
    - Leave request is recorded with specified dates.
- 

## **11. View Leave Application Status**

- **User Role:** Employee
  - **Goal:** Check whether the leave request was approved or rejected.
  - **Reason:** Keeps employees updated about their leave request decisions.
  - **Pre-conditions:**
    - Employee has applied for leave.
  - **Post-conditions:**
    - Status (Approved/Rejected) is visible.
- 

## **12. View Monthly Salary Slip**

- **User Role:** Employee
  - **Goal:** View monthly salary breakdown.
  - **Reason:** Employees need to verify their earnings, deductions, and bonuses.
  - **Pre-conditions:**
    - Payroll must be processed for that month.
  - **Post-conditions:**
    - Salary slip with all details is displayed.
-

### 13. Manager Views Pending Leave Requests

- **User Role:** Manager
  - **Goal:** See a list of all pending leave requests.
  - **Reason:** Managers need a consolidated view to manage approvals.
  - **Pre-conditions:**
    - There are leave requests waiting for approval.
  - **Post-conditions:**
    - Pending requests list is shown to the manager.
- 

### 14. System Calculates Employee Net Salary

- **User Role:** System
  - **Goal:** Automatically calculate employee net salary after tax and deductions.
  - **Reason:** Ensures correct salary distribution.
  - **Pre-conditions:**
    - Employee attendance and leaves are recorded.
  - **Post-conditions:**
    - Net salary amount is calculated and saved.
- 

### 15. Employee Gets Notification after Leave Decision

- **User Role:** Employee
  - **Goal:** Receive notification when leave is approved or rejected.
  - **Reason:** Keeps employee informed.
  - **Pre-conditions:**
    - Manager has taken action on leave request.
  - **Post-conditions:**
    - Notification sent to employee.
-

## 6. Product Backlog

ID	As a...	I want to be able to...	So that...	Priority	Sprint	Status
1	Employee	Register in the system	I can access my relevant features	Must	1	Done
2	User	Log in to the system	I can access the system	Must	1	Done
3	Employee	Access my dashboard	I can view leave balance and payroll summary	Must	1	Done
4	Manager	Access manager dashboard	I can manage leave requests and team details	Must	1	Done
5	Employee	Apply for leave	My request is submitted to my manager	Must	1	Done
6	Manager	Approve or reject leave requests	I can manage employee leave effectively	Must	1	Done
7	Employee	View leave balance	I can plan my future leaves accordingly	Must	1	Done
8	Manager	Track employee leave history	I can make informed leave approval decisions	Should	2	Done
9	Manager	Configure leave policies	Employee leave requests follow policies	Must	2	Done
10	Employee	View payroll summary	I can understand my salary breakdown	Must	3	Done
11	Manager	Process payroll	Employees receive salaries on time	Must	3	Done
12	Manager	Calculate taxes and deductions	Employee salaries are	Must	3	Done

## 7. Sprint Backlog

### Sprint 1: Employee Leave Management

**Sprint Goal:** Implement core functionalities for employees to apply for leave and track their leave balance.

ID	User Story	Tasks	Owner	Status	Estimated Effort	Day 1	Day 2	Day 3	Priority
US001	Employee and Manager Registration	Implement registration form	Abdullah	Completed	4	2	2	0	Must
US002	Login	Setup authentication system	Abdullah	Completed	2	0	1	1	Must
US003	Employee Dashboard	Design and <u>develop</u> dashboard UI	Abdur Raheem	Completed	2	0	2	0	Must
US004	Apply for Leave	Develop leave application functionality	Abdullah	Completed	4	0	2	2	Must
US005	View Leave Balance	Display leave balance on dashboard	Abdur Raheem	Completed	2	0	2	0	Must

### Sprint 2: Manager Leave Management

**Sprint Goal:** Enable managers to review, approve, and manage leave requests.

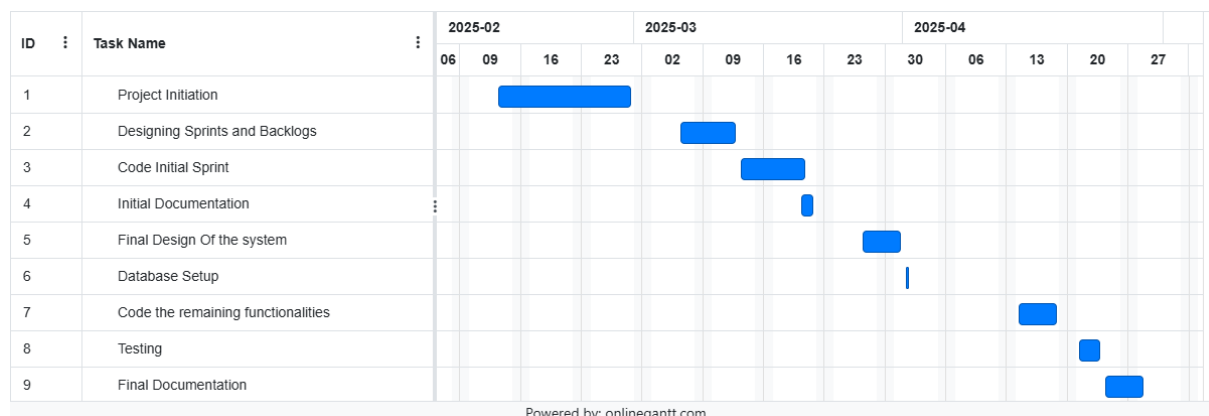
ID	User Story	Tasks	Owner	Status	Estimated Effort	Day 1	Day 2	Day 3	Priority
US006	Manager Dashboard	<u>Develop</u> manager dashboard UI	Abdur Raheem	To Do	10	5	3	2	Must
US007	Approve or Reject Leave Requests	Implement approval system	Afrah	To Do	12	6	4	2	Must
US008	Track Employee Leave History	Create leave history module	Abdullah	To Do	8	4	2	2	Should
US009	Configure Leave Policies	Develop policy configuration settings	Afrah	To Do	10	5	3	2	Must

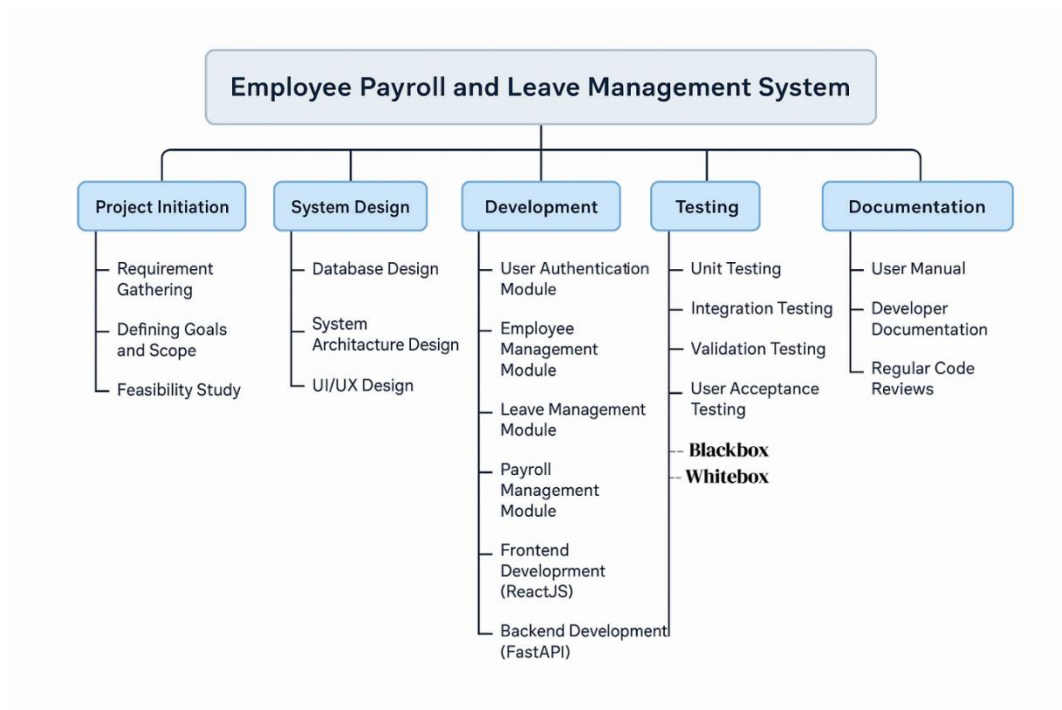
## Sprint 3: Payroll System

**Sprint Goal:** Enable managers and the system to process employee payroll, manage salaries, tax deductions, and generate payroll reports.

ID	User Story	Tasks	Owner	Status	Estimated Effort	Day 1	Day 2	Day 3	Priority
US010	Process Monthly Payroll	Develop payroll processing module	Afrah	To Do	12	6	4	2	Must
US011	Deduct Tax from Salaries	Implement tax deduction calculations	Abdur Raheem	To Do	10	5	3	2	Must
US012	View Monthly Salary Slip	Create salary slip generation functionality	Afrah	To Do	8	4	2	2	Must
US013	View Payroll Summary	Develop payroll summary view	Abdullah	To Do	8	4	2	2	Should
US014	View Tax Deduction Summary	Develop tax deduction details module	Afrah	To Do	8	4	2	2	Should
US015	System Calculates Net Salary	Integrate salary, leave deductions, and bonuses	Abdur Raheem	To Do	10	5	3	2	Must

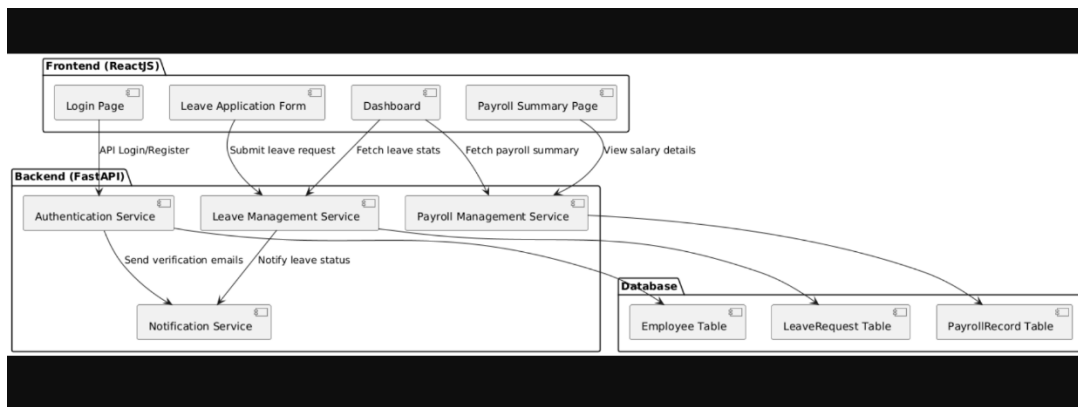
## 8. Project Plan



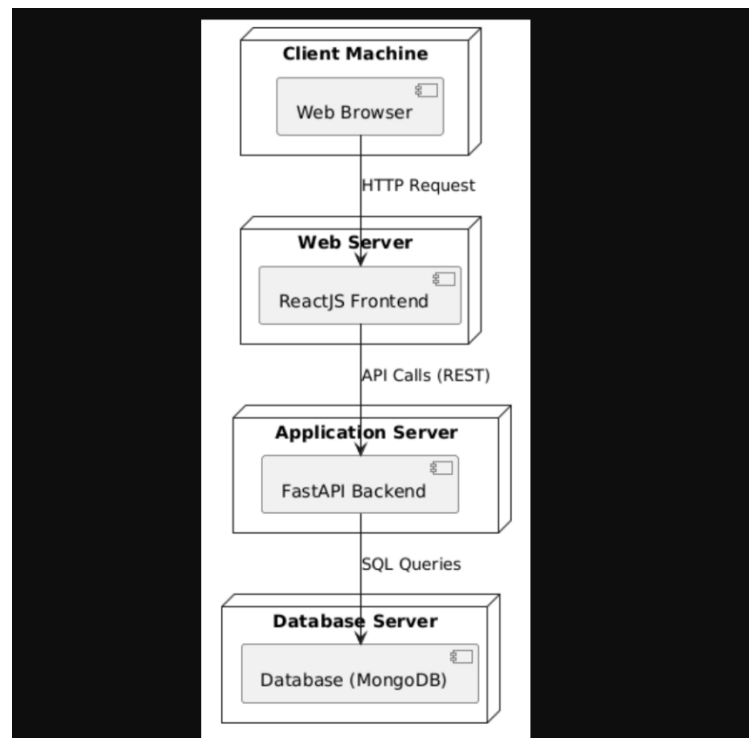


## 9. Architecture Diagram

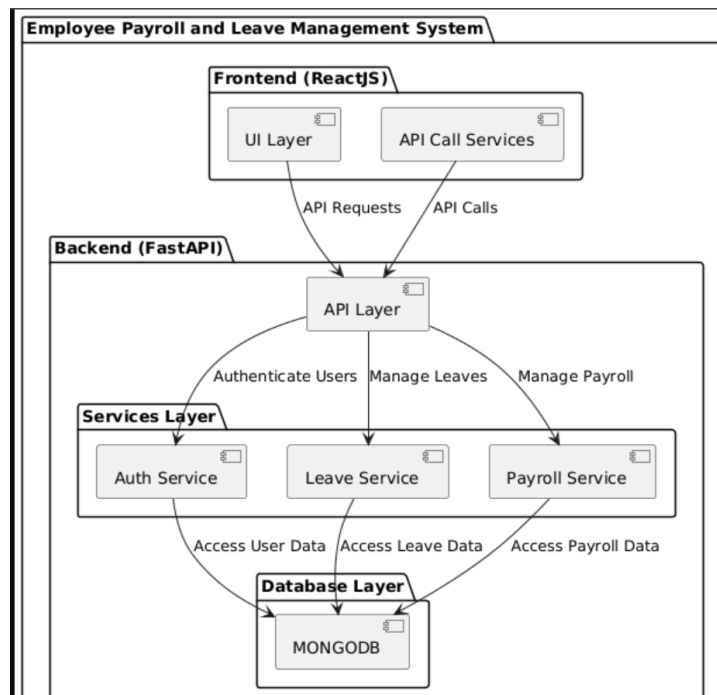
### Component Diagram:



## Deployment Diagram:



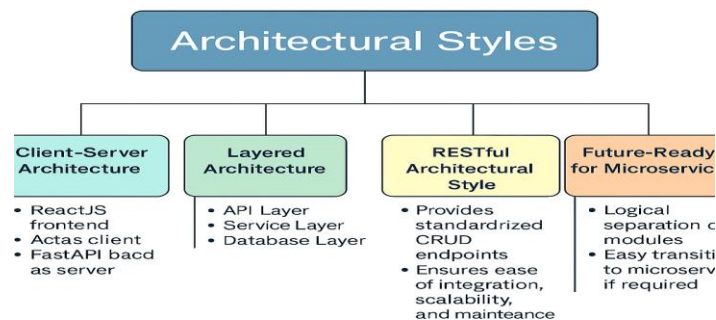
## Package Diagram:





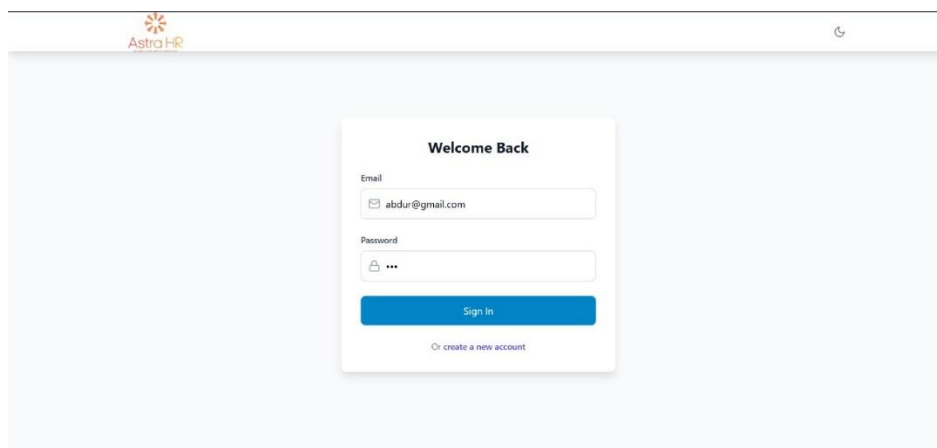
## Architectural Styles

- The system follows a Client-Server Architecture:
    - o ReactJS frontend acts as the client, managing user interactions.
    - o FastAPI backend acts as the server, handling business logic and database operations.
  - Layered Architecture is used within the backend:
    - o API Layer handles incoming HTTP requests.
    - o Service Layer processes business logic (Authentication, Leave Management, Payroll Management).
    - o Database Layer interacts with the database.
  - The backend APIs are designed following the RESTful Architectural Style:
    - o Provides standardized CRUD endpoints.
    - o Ensures ease of integration, scalability, and maintenance.
  - The system structure is designed to be future-ready for Microservices Architecture:
    - o Logical separation of modules like Authentication, Leave Management, and Payroll Management allows easy transition to microservices if required.
- 

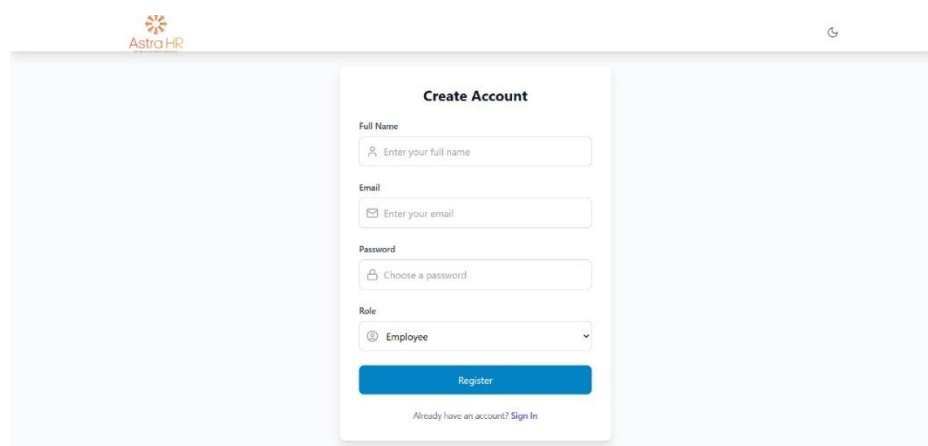


## 10. Design (all sprint 3 items)

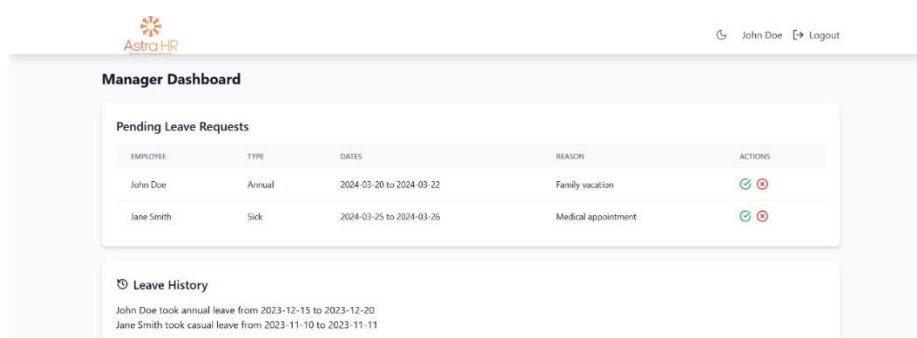
## 11. Actual implementation screenshots



The screenshot shows the 'Welcome Back' login page of the AstraHR system. The page features the AstraHR logo in the top left corner and a user profile icon in the top right corner. The main content is a white card with the title 'Welcome Back'. It contains two input fields: 'Email' with the value 'abdur@gmail.com' and 'Password' with masked characters '\*\*\*'. Below these fields is a blue 'Sign In' button. At the bottom of the card, there is a link that says 'Or create a new account'.



The screenshot shows the 'Create Account' registration page of the AstraHR system. The page features the AstraHR logo in the top left corner and a user profile icon in the top right corner. The main content is a white card with the title 'Create Account'. It contains four input fields: 'Full Name' with placeholder text 'Enter your full name', 'Email' with placeholder text 'Enter your email', and 'Password' with placeholder text 'Choose a password'. Below these fields is a dropdown menu for 'Role' with 'Employee' selected. At the bottom of the card is a blue 'Register' button. Below the card, there is a link that says 'Already have an account? Sign In'.




The screenshot shows the 'Manager Dashboard' of the AstraHR system. The page features the AstraHR logo in the top left corner and a user profile icon in the top right corner. The main content is a white card with the title 'Manager Dashboard'. It contains two sections: 'Pending Leave Requests' and 'Leave History'.

**Pending Leave Requests**

EMPLOYEE	TYPE	DATES	REASON	ACTIONS
John Doe	Annual	2024-03-20 to 2024-03-22	Family vacation	<span>✓</span> <span>✗</span>
Jane Smith	Sick	2024-03-25 to 2024-03-26	Medical appointment	<span>✓</span> <span>✗</span>

**Leave History**

John Doe took annual leave from 2023-12-15 to 2023-12-20  
Jane Smith took casual leave from 2023-11-10 to 2023-11-11

 John Doe Logout

### Apply for Leave


Leave Type  
Annual Leave

Start Date  
mm/dd/yyyy

End Date  
mm/dd/yyyy

Reason

Submit Application

 John Doe Logout

### Employee Dashboard

#### Annual Leave

Total: 20 days  
Used: 5 days  
Remaining: 15 days

#### Sick Leave

Total: 10 days  
Used: 2 days  
Remaining: 8 days

#### Casual Leave

Total: 5 days  
Used: 1 days  
Remaining: 4 days

#### Payroll Summary

Base Salary: \$5000  
Bonuses: +\$500  
Deductions: -\$300  
**Net Salary: \$5200**  
Last Updated: March 2025

Apply for Leave

### Configure Leave Policies

Annual Leave Limit: 20

Sick Leave Limit: 10

Casual Leave Limit: 5

### Process Monthly Payroll

Run Payroll

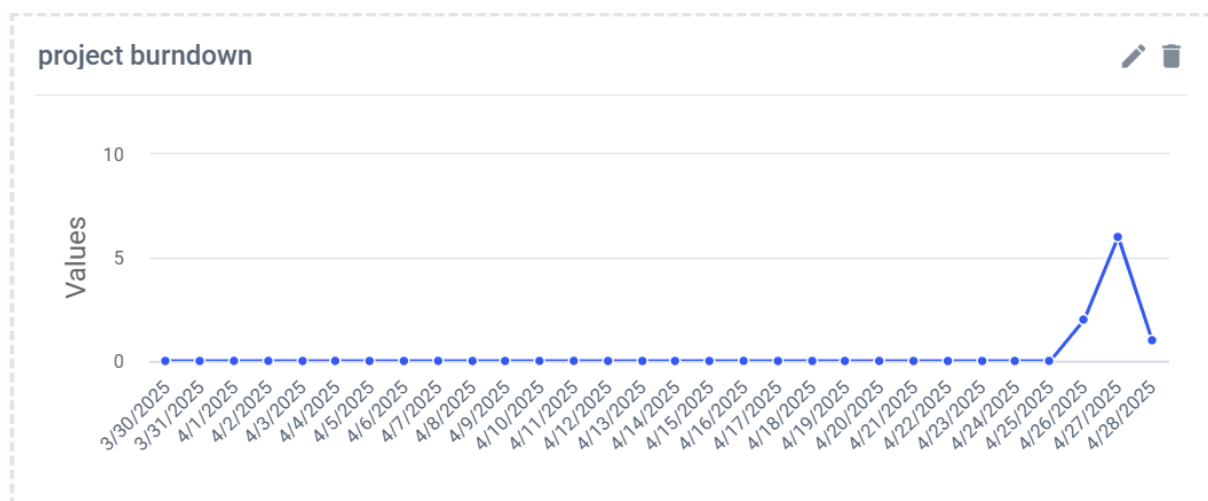
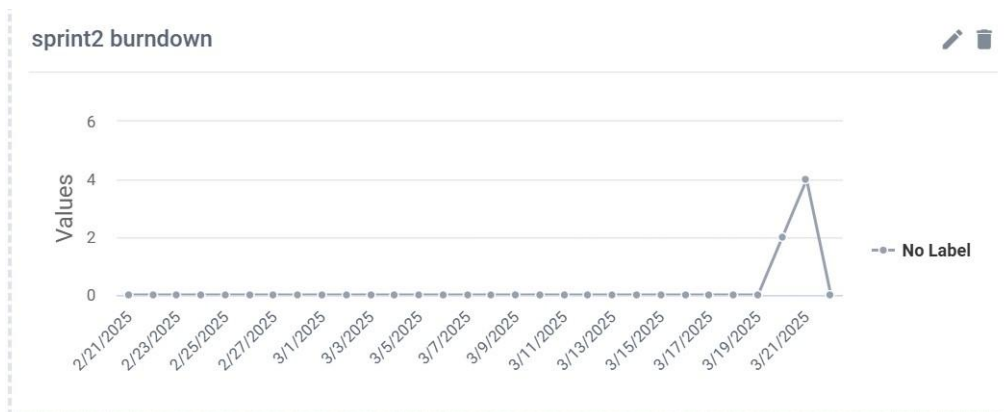
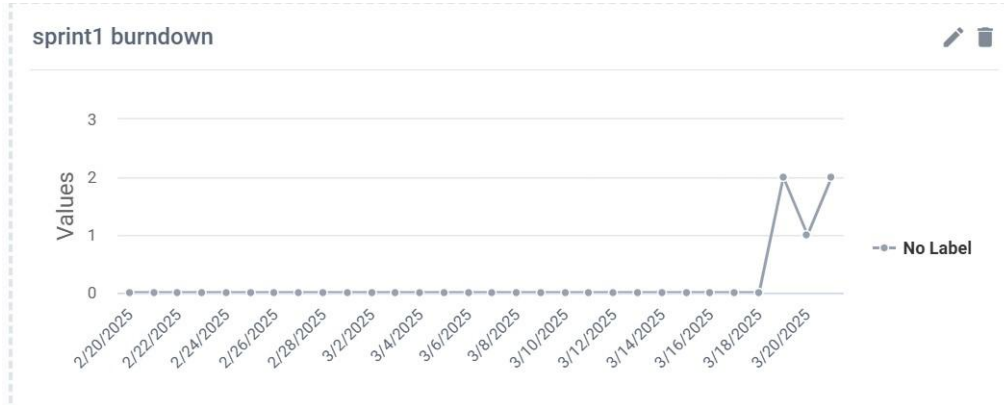
### Manage Salary Structures

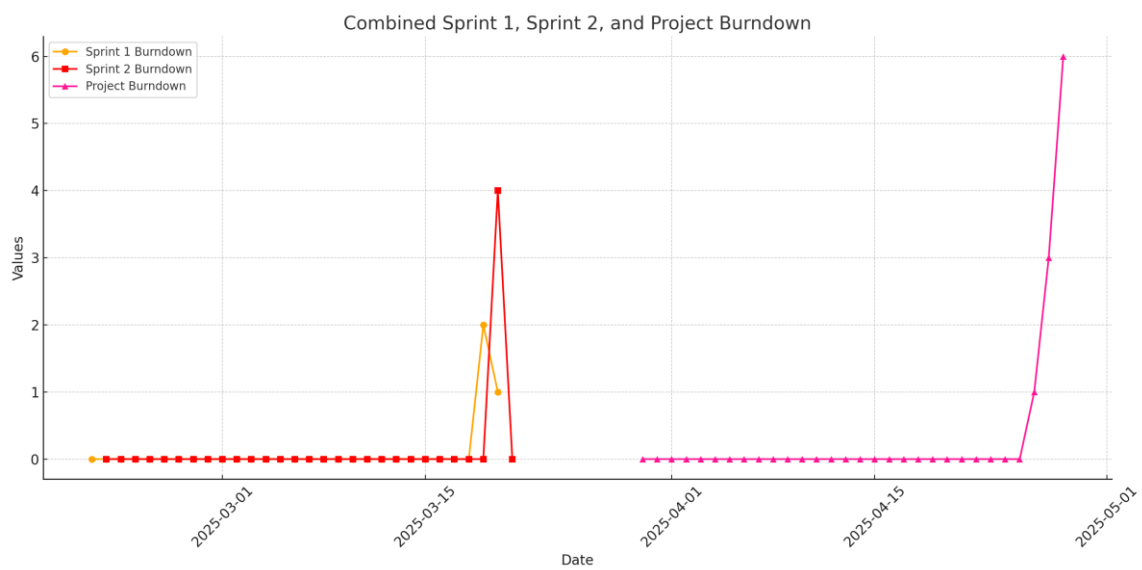
Role	Base Salary	Actions
Software Engineer	\$5000	5000
Project Manager	\$5500	5500
HR Manager	\$4800	4800

### Payroll Report

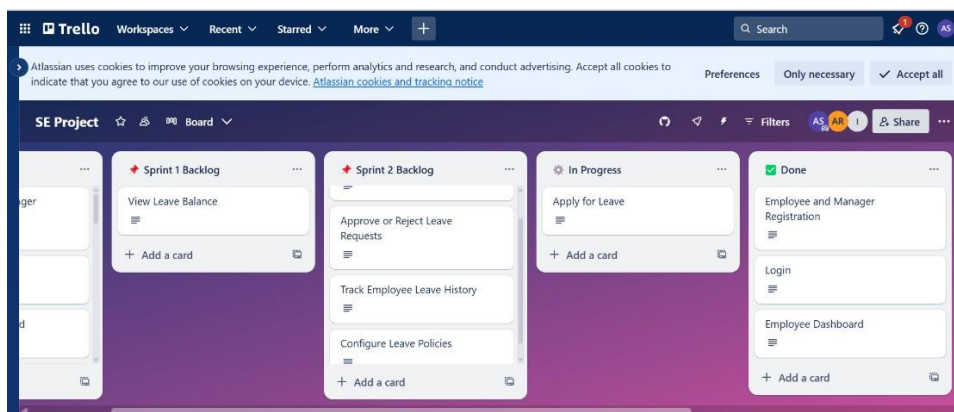
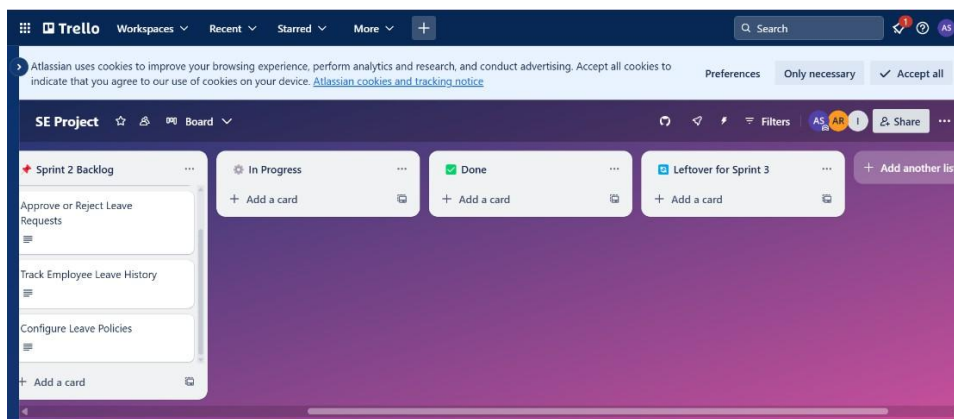
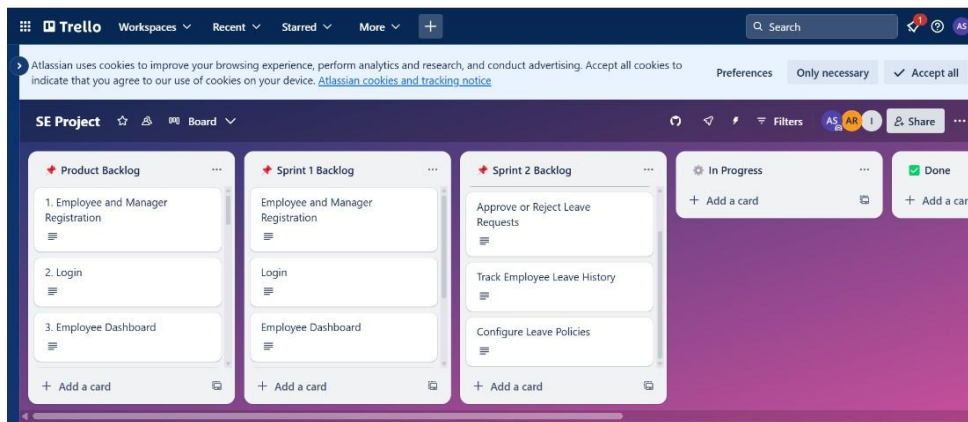
Employee	Base Salary	Overtime	Deductions	Net Salary
John Doe	\$5000	\$200	-\$300	\$4900
Jane Smith	\$5500	\$150	-\$350	\$5300

## 12. Product Burn down chart for the project





## 13. Trello board screen shots



## 14. Testcases -Black box

### 1. Equivalence Class Partitioning (ECP)

TC ID	Test Scenario	Input	EC Type	Expected Output	Status
TC-ECP-01	Apply Leave – Valid duration	Type: Annual, Days: 5	Valid	Leave request submitted	Pass
TC-ECP-02	Apply Leave – Invalid type	Type: “Emergency”	Invalid	Error: Invalid leave type	Pass
TC-ECP-03	Login – Valid	Email + correct	Valid	Logged in to dashboard	Pass

	credentials	password			
TC-ECP-04	Login – Incorrect password	Correct email + wrong password	Invalid	Error: Invalid credentials	Pass
TC-ECP-05	Salary – Below threshold	Salary = 10,000 PKR	Invalid	Error: Invalid salary	Pass
TC-ECP-06	Salary – Valid range	Salary = 40,000 PKR	Valid	Payroll processed	Pass
TC-ECP-07	Register – Invalid role	Role: “Admin”	Invalid	Error: Invalid role	Pass
TC-ECP-08	Leave Request – Invalid date range	Start: 2025-06-05, End: 2025-06-01	Invalid	Error: Invalid dates	Pass
TC-ECP-09	View Payroll – Valid employee	EmployeeID: E101	Valid	Payroll summary shown	Pass
TC-ECP-10	Configure Leave Policy – Invalid leave limit	Annual leave = -5	Invalid	Error: Invalid value	Pass

## 2. Boundary Value Analysis (BVA)

TC ID	Test Scenario	Input	Boundary Type	Expected Output
TC-BVA-01	Leave Days – Lower boundary	Days = 1	Lower valid	Leave request accepted
TC-BVA-02	Leave Days – Upper boundary	Days = 30	Upper valid	Leave request accepted
TC-BVA-03	Leave Days – Just below lower	Days = 0	Invalid	Error: Minimum 1 day
TC-BVA-04	Leave Days – Just above upper	Days = 31	Invalid	Error: Max leave exceeded
TC-BVA-05	Salary – Minimum valid	Salary = 15,000 PKR	Lower valid	Payroll processed
TC-BVA-06	Salary – Maximum valid	Salary = 500,000 PKR	Upper valid	Payroll processed
TC-BVA-07	Salary – Below minimum	Salary = 14,999 PKR	Invalid	Error: Salary too low
TC-BVA-08	Salary – Just above maximum	Salary = 500,001 PKR	Invalid	Error or special approval
TC-BVA-09	Payroll Month – Lower boundary	Month = 1	Valid	Payroll for January processed
TC-BVA-10	Payroll Month – Above valid range	Month = 13	Invalid	Error: Invalid month

## 3. Functional Test Cases based on User Stories

TC ID	Functionality	Input	Expected Result
TC-FUNC-01	View Payroll Summary	Logged-in employee	Payroll summary visible
TC-FUNC-02	Apply for Leave	Valid leave form	Request sent, notification sent to

			manager
TC-FUNC-03	Approve Leave	Manager action	Leave approved, employee notified
TC-FUNC-04	Track Leave History	Manager	Historical leave shown
TC-FUNC-05	Configure Leave Policies	Set 15 sick leaves	Policies updated
TC-FUNC-06	Process Monthly Payroll	Run payroll for April	Salary processed
TC-FUNC-07	Deduct Tax from Salaries	Salary = 60,000	Tax applied in summary
TC-FUNC-08	Manage Employee Salary	Update bonus to 10%	New salary calculated
TC-FUNC-09	View Leave Balance	Employee logged in	Current balance shown
TC-FUNC-10	Invalid Login	Wrong password	Error: Invalid credentials

## 15. Testcases -White box

Test Type: Unit and Integration Testing (White-Box Testing)

Test Suites Run: 2

Tests Executed: 13

Tests Passed: 13 Test Framework Used: Jest

File	% Stmts	% Branch	% Funcs	% Lines	Uncovered Line #s
All files	95.57	85.71	93.75	96.39	
task-management-system	100	100	100	100	
app.js	100	100	100	100	
task-management-system/configs	89.65	86.36	66.66	92.85	
config.js	100	93.75	100	100	1
db.config.js	75	100	100	75	10
log.config.js	85.71	66.66	50	92.3	42
task-management-system/constants	100	100	100	100	
constant.js	100	100	100	100	
task-management-system/domain/task	100	100	100	100	
task.controller.js	100	100	100	100	
task.schema.js	100	100	100	100	
task.service.js	100	100	100	100	
task.validator.js	100	100	100	100	
task-management-system/middleware	80	100	100	80	
validator.js	80	100	100	80	7
task-management-system/util	88.88	50	100	88.88	
utility.js	88.88	50	100	88.88	16
Test Suites: 2 passed, 2 total Tests: 13 passed, 13 total Snapshots: 0 total Time: 5.673 s, estimated 48 s Ran all test suites.					

### Key Observations:

- High Coverage: Most of the critical modules like controllers, services, and schemas are at 100% coverage.
- Config Issues: The `log\_config.js` file has the lowest branch and statement coverage.



- Middleware Gaps: The `auth.js` middleware needs more test cases to handle edge cases.
- Validator Test: One function in `validator.js` remains uncovered.

**Recommendations:**

- Improve Branch Testing: Focus on edge cases in `log\_config.js` and `auth.js`.
- Maintain Coverage: Ensure future updates preserve 95%+ coverage.
- Use Mocks/Stubs: Simulate logging/db return paths to improve coverage.

**Tools & Environment:**

- Testing Library: Jest
- Language: JavaScript (Node.js Backend)

## 16. Work Division between group members

- **Abdullah**
  - Worked on the backend development.
  - Contributed to creating the presentation slides and some system diagrams.
- **Abdur Raheem**
  - Worked on the frontend development.
  - Assisted in creating the presentation slides and testing.
- **Afrah**
  - Worked on the frontend development.
  - Prepared the final project report and testing.

## 17. Lesson learnt by group

- Importance of clear communication and task distribution within the team.
- Early planning and requirement gathering are critical for project success.
- Hands-on experience improved our system design and problem-solving skills.
- Collaboration and regular updates helped in managing deadlines effectively.
- Emphasis on testing and debugging to ensure system reliability and performance.
- Adaptability and flexibility were key to overcoming unexpected challenges.
- Learned the importance of adaptability in handling changing requirements and challenges during development.
- Gained a deeper understanding of how timely completion of tasks is essential in meeting deadlines and ensuring project success.

