**USE CASE DIAGRAM**



**ACTIVITY DIAGRAM**





**Use Case Specifications:**

|  |  |
| --- | --- |
| **Use Case ID** | **UC001** |
| **Use Case Name** | User Login |
| **Description** | Allows the user to log into the system. |
| **Actors (Primary)** | Primary: Employee, Admin |
| **Basic Flow** | 1. User navigates to login screen. |
|  | 2. Enters credentials. |
|  | 3. System verifies. |
|  | 4. User is granted access. |
| **Alternative Flow** | Invalid credentials: System prompts user to re-enter credentials. |
| **Exceptional Flow** | Server unavailable: Error message displayed to user. |
| **Pre-Condition** | User must have a valid username and password. |
| **Post-Condition** | User is successfully logged in. |
| **Assumptions** | System is connected to the server. |
| **Constraints** | Requires secure password storage. |
| **Dependencies** | Authentication system integration. |
| **Input** | Input: Username and password |
| **Output** | Output: Session Token. |
| **Business Rules** | Users are locked out after 3 failed login attempts. |
| **Miscellaneous** | Timeouts after 15 minutes of inactivity. |

|  |  |
| --- | --- |
| **Use Case ID** | **UC002** |
| **Use Case Name** | Add New Employee |
| **Description** | Enables admin to add employee details into the HRMS. |
| **Actors (Primary)** | Primary: Admin Secondary: Employee |
| **Basic Flow** | 1. Admin navigates to the "Add Employee" form. |
|  | 2. Fills in employee details. |
|  | 3. System saves data in the database. |
|  | 4. Confirmation is displayed. |
| **Alternative Flow** | Missing required fields: System displays error. |
| **Exceptional Flow** | Database error: Operation fails; error logged. |
| **Pre-Condition** | Admin must have proper permissions. |
| **Post-Condition** | New employee data is successfully saved. |
| **Assumptions** | Admin has correct employee details available. |
| **Constraints** | Data entry must conform to system validations. |
| **Dependencies** | Dependencies on employee ID generation service. |
| **Input** | Input: Employee details |
| **Output** | Output: Confirmation message. |
| **Business Rules** | Employee ID must be unique within the system. |
| **Miscellaneous** | Requires periodic audits for data accuracy. |

|  |  |
| --- | --- |
| **Use Case ID** | **UC003** |
| **Use Case Name** | Generate Payroll |
| **Description** | Automates payroll generation for employees. |
| **Actors (Primary)** | Primary: HR Admin Secondary: Finance Department |
| **Basic Flow** | 1. HR admin selects the payroll period. |
|  | 2. System calculates salaries. |
|  | 3. Payroll report generated. |
|  | 4. Payroll is distributed. |
| **Alternative Flow** | Missing salary components: System prompts for manual override. |
| **Exceptional Flow** | Incorrect calculation formula: Admin manually reviews and adjusts. |
| **Pre-Condition** | Salary components must be predefined in the system. |
| **Post-Condition** | Salaries are calculated and reports generated. |
| **Assumptions** | Salary components are pre-approved. |
| **Constraints** | Must adhere to tax laws and company policies. |
| **Dependencies** | Dependency on tax calculation APIs. |
| **Input** | Input: Employee salary details |
| **Output** | Output: Payroll report. |
| **Business Rules** | Calculations must include applicable taxes and deductions. |
| **Miscellaneous** | Regular updates required for tax codes. |

|  |  |
| --- | --- |
| **Use Case ID** | **UC004** |
| **Use Case Name** | Leave |
| **Description** | Allows managers to review and approve/reject leave requests. |
| **Actors (Primary)** | Primary: Manager Secondary: Employee |
| **Basic Flow** | 1. Manager logs in. |
|  | 2. Navigates to "Leave Requests". |
|  | 3. Reviews details. |
|  | 4. Approves or rejects the request. |
| **Alternative Flow** | Leave overlaps with critical project dates: System prompts for justification. |
| **Exceptional Flow** | Manager absent: Request escalated to higher authority. |
| **Pre-Condition** | Leave request must be submitted by the employee. |
| **Post-Condition** | Leave status is updated in the system. |
| **Assumptions** | Manager has necessary access to leave records. |
| **Constraints** | Requests older than 30 days cannot be processed. |
| **Dependencies** | Requires employee attendance records integration. |
| **Input** | Input: Leave request |
| **Output** | Output: Approval/rejection notification. |
| **Business Rules** | Leave cannot exceed accrued balance without special approval. |
| **Miscellaneous** | Requires regular update of leave policies. |

|  |  |
| --- | --- |
| **Use Case ID** | **UC005** |
| **Use Case Name** | Submit Expense Report |
| **Description** | Enables employees to submit expenses for reimbursement. |
| **Actors (Primary)** | Primary: Employee Secondary: Finance Department |
| **Basic Flow** | 1. Employee navigates to "Submit Expense". |
|  | 2. Uploads receipts and fills details. |
|  | 3. System validates entries. |
|  | 4. Expense report submitted for approval. |
| **Alternative Flow** | Invalid format for receipts: System prompts for corrections. |
| **Exceptional Flow** | Missing mandatory fields: System rejects submission. |
| **Pre-Condition** | Employee must have valid login credentials. |
| **Post-Condition** | Expense report is successfully submitted. |
| **Assumptions** | All receipts are available for upload. |
| **Constraints** | File upload size limits apply. |
| **Dependencies** | Dependency on file storage services. |
| **Input** | Input: Expense details, receipts |
| **Output** | Output: Confirmation email. |
| **Business Rules** | Expenses exceeding limits require additional approvals. |
| **Miscellaneous** | Requires adherence to corporate expense policies. |

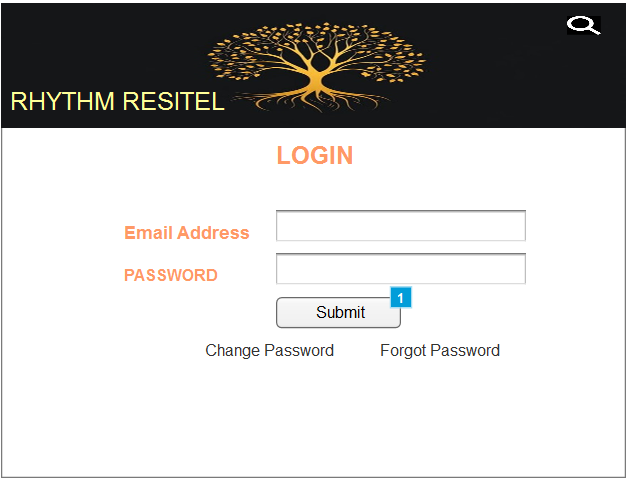
|  |  |
| --- | --- |
| **Use Case ID** | **UC006** |
| **Use Case Name** | View Attendance |
| **Description** | Enables employees to submit expenses for reimbursement. |
| **Actors (Primary)** | Primary: Employee Secondary: HR Admin |
| **Basic Flow** | 1. Employee logs in. |
|  | 2. Navigates to "Attendance". |
|  | 3. Views monthly or yearly records. |
|  | 4. Filters and downloads report if needed. |
| **Alternative Flow** | Incorrect data: User requests admin for corrections. |
| **Exceptional Flow** | Data retrieval error: System displays error message and suggests retry. |
| **Pre-Condition** | Attendance records must be updated daily. |
| **Post-Condition** | Attendance is displayed to the user. |
| **Assumptions** | Attendance data is accurately logged in the system. |
| **Constraints** | Data must comply with labour laws and company policies. |
| **Dependencies** | Dependency on attendance tracking hardware/software. |
| **Input** | Input: Date range for attendance |
| **Output** | Output: Attendance records. |
| **Business Rules** | User cannot edit attendance; updates require HR admin intervention. |
| **Miscellaneous** | Attendance system must sync with other HR modules. |

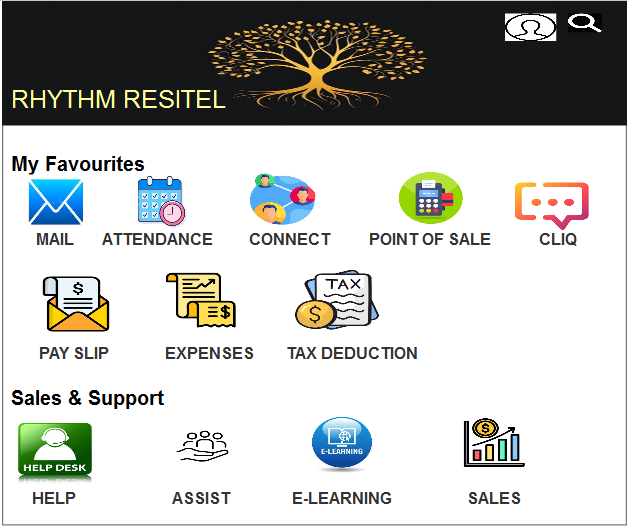
|  |  |
| --- | --- |
| **Use Case ID** | **UC007** |
| **Use Case Name** | Generate Reports |
| **Description** | Allows admins to generate various analytical and financial reports. |
| **Actors (Primary)** | Primary: Admin Secondary: Management |
| **Basic Flow** | 1. Admin selects report type. |
|  | 2. Specifies date range and parameters. |
|  | 3. System generates and downloads the report. |
|  |  |
| **Alternative Flow** | Missing data: Report excludes certain sections and provides warnings. |
| **Exceptional Flow** | Heavy system load: Report generation is delayed; admin notified. |
| **Pre-Condition** | Admin must have necessary permissions. |
| **Post-Condition** | Reports are successfully generated. |
| **Assumptions** | Data required for reports is complete and updated. |
| **Constraints** | Certain reports may take longer to generate under high load. |
| **Dependencies** | Dependency on reporting tools and database integration. |
| **Input** | Input: Report type, parameters |
| **Output** | Output: Downloadable report. |
| **Business Rules** | Reports must comply with internal and regulatory standards. |
| **Miscellaneous** | Regular maintenance of reporting system needed. |

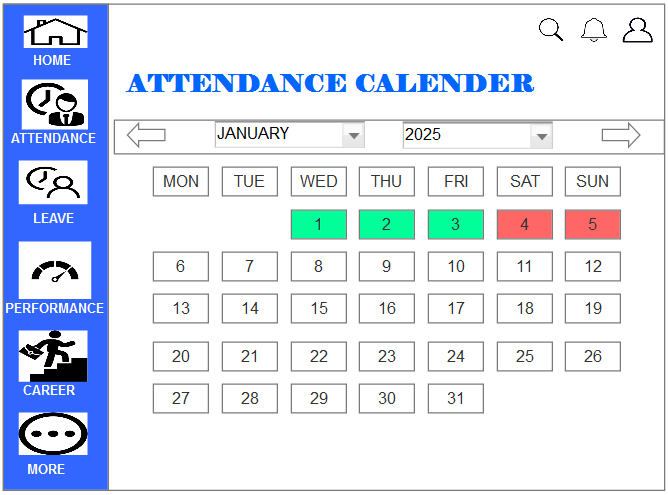
|  |  |
| --- | --- |
| **Use Case ID** | **UC008** |
| **Use Case Name** | Process Termination |
| **Description** | Manages employee exit processes, including final settlements and account deactivation. |
| **Actors (Primary)** | Primary: HR Admin Secondary: IT Admin |
| **Basic Flow** | 1. HR initiates termination process. |
|  | 2. Final settlements calculated. |
|  | 3. IT deactivates user accounts. |
|  | 4. Exit checklist completed. |
| **Alternative Flow** | Outstanding loans or dues: System prompts for clearance. |
| **Exceptional Flow** | Payroll integration error: Settlements delayed until resolved. |
| **Pre-Condition** | Termination request must be pre-approved by management. |
| **Post-Condition** | Termination process is successfully completed. |
| **Assumptions** | All required approvals are obtained in advance. |
| **Constraints** | Process cannot proceed without approval from all departments. |
| **Dependencies** | Dependency on payroll and IT systems. |
| **Input** | Input: Termination request |
| **Output** | Output: Final settlement, exit checklist. |
| **Business Rules** | Termination process must comply with legal and company policies. |
| **Miscellaneous** | Requires secure handling of sensitive employee data. |

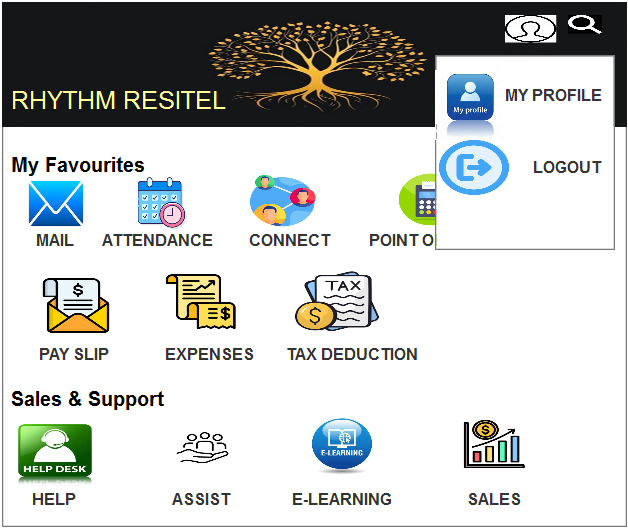
**Document 7- Screens and pages**

****

****

****

****

****

**Document 8- Tools-Visio and Axure**

In this project, Visio and Axure were essential tools that facilitated effective communication, design, and documentation throughout various stages of development.

**MS Visio** was primarily used for creating detailed process flows, such as sales and HR workflows, system architecture diagrams, and organizational charts. These visuals provided a clear and structured representation of both AS-IS and TO-BE processes, ensuring all stakeholders had a shared understanding of current operations and the proposed system improvements. The ability to use pre-built templates and customize diagrams in Visio allowed for quick iterations and updates as feedback was received.

**Axure**, on the other hand, played a pivotal role in designing the user interface and overall user experience. Its robust prototyping capabilities enabled us to create interactive wireframes and mock-ups that closely simulated the final application. This functionality proved invaluable in capturing stakeholder feedback early in the project lifecycle, reducing the risk of rework during later stages. Axure also allowed us to test user interactions and workflows, ensuring that the design was intuitive and aligned with end-user expectations.

Together, these tools not only enhanced collaboration among the project team and stakeholders but also provided a clear roadmap for developers and testers to follow. By leveraging Visio and Axure effectively, we were able to bridge the gap between business requirements and technical implementation, ensuring the project stayed on track and met its objectives.

**Document 9- BA experience**

#### 1. ****Requirement Gathering****

During this phase, I employed the **MoSCoW technique** to prioritize requirements effectively into Must-Have, Should-Have, Could-Have, and Won't-Have categories. While gathering requirements, there were instances when the client was unavailable for a period. To mitigate this, I identified alternate points of contact from the client’s side to ensure uninterrupted information flow. I validated the collected requirements using the **FURPS technique** (Functionality, Usability, Reliability, Performance, and Supportability), ensuring they aligned with the project goals. Duplicate or repetitive requirements were promptly identified and removed to maintain clarity. To capture more specific requirements, **prototyping** was extensively used, which provided visual feedback and ensured mutual understanding.

#### 2. ****Requirement Analysis****

In the analysis phase, I used **UML diagrams** to visually describe requirements, such as use case diagrams, sequence diagrams, and class diagrams. Additionally, **activity diagrams** were created to map the process flows, ensuring a thorough understanding of the system. Once completed, I communicated these diagrams to the team for feedback. Differences in interpretation often arose, requiring me to facilitate discussions, consider all inputs, and update the diagrams accordingly. I also prepared essential documentation, including the **Business Requirements Specification (BRS)** and **Software Requirements Specification (SRS)** documents.

#### 3. ****Design****

The design phase involved creating **test cases** directly from use case diagrams. I ensured both positive and negative test scenarios were covered, leaving no room for overlooked scenarios that might impact the project in later stages. Constant communication with the client was maintained to validate the design and the solution documents. Additionally, I prepared **test data** for testing purposes and consistently updated the **Requirements Traceability Matrix (RTM)** to ensure all requirements were mapped and accounted for in the design phase.

#### 4. ****Development****

I organized and facilitated **JAD (Joint Application Development) sessions** to ensure clear communication between stakeholders and the technical team. During development, I clarified queries from the development team to ensure they aligned with the documented requirements. Handling conflicts or resistance during JAD sessions was part of the process. I conducted one-on-one discussions with team members to explain how their inputs and cooperation impacted the project’s success, fostering a collaborative environment. Additionally, I referred to diagrams and documentation during coding to align development efforts. Regular meetings with the technical team and the client were organized, and for those who couldn’t attend, sessions were recorded, and follow-up discussions were arranged to ensure no one missed critical updates.

#### 5. ****Testing****

In the testing phase, I prepared and reviewed test cases derived from the use cases, performed **high-level testing**, and ensured the test data provided by the client was complete and accurate. I continuously updated the RTM to validate that every requirement was tested. Once testing was completed, I obtained client signoff and prepared them for **User Acceptance Testing (UAT)** by providing necessary guidance and support.

#### 6. ****Deployment****

During deployment, I shared the finalized RTM attached to the **project closure document** with the client. I coordinated the preparation of **end-user manuals** and organized **training sessions** to familiarize end-users with the new system. I ensured all relevant stakeholders attended the sessions and resolved any issues or queries they had to facilitate a smooth transition to the deployed system.

This structured approach in each phase helped ensure the project’s success, from gathering accurate requirements to delivering a system that met stakeholder expectations.