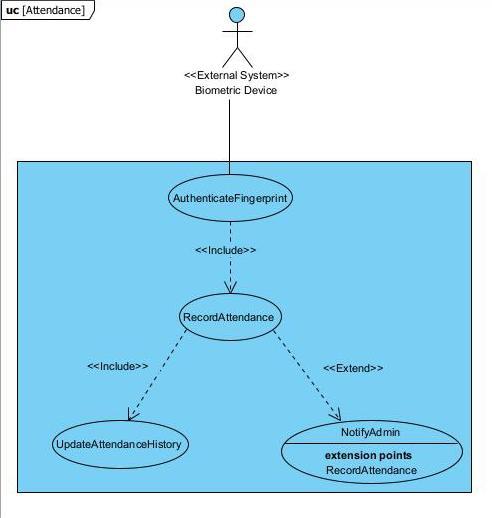


****

**Fig.1 Use Case Diagrams For Employee Management System**

**Table1. Descriptions of Use Case Diagrams For Employee Management System**

**Create Account**

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-1 | |
| **Use Case Name** | Create Account | |
| **Actor** | Admin | |
| **Summary** | This use case describes how the admin creates a new user account in the Employee Management System (EMS) by entering required details and assigning roles or permissions. | |
| **Precondition** | **1**.The user must have access to a device with an internet connection  **2**.The admin must be logged in with valid credentials and have permission to manage accounts. | |
| **Basic Scenario** | **Actor Action** | **System Response** |
| **Step1:**Admin selects “Create Account” option  **Step3:**Admin enters employee details  **Step5:**Admin assigns user  role/permissions  **Step7:**Admin submits the form | **Step2:**System displays the account creation form  **Step4:**System validates the entered information  **Step6:**System saves role settings  **Step8:**System creates the account and confirms success |
| **Alternative Scenario** | **Step9:**If validation fails, system displays an error message indicating the issue.  **Step10:**Admin corrects the data and resubmits the form. | |
| **Post\_Conditon** | The new employee account is successfully created, stored in the system, and ready for use. | |

**Login**

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-2 | |
| **Use Case Name** | Login | |
| **Actor** | Admin, Employee, Manager | |
| **Summary** | This use case describes how a system user (admin, employee, or manager) securely logs in to the EMS by providing valid credentials to access authorized features.  The Login use case enables authorized employees to securely access the EMS application. The system verifies user credentials (username/employee ID and password) against a stored database. Successful login grants access to the employee's assigned modules and data based on their role-based permissions. Unsuccessful login triggers appropriate error messages and security protocols. This use case is fundamental for system security, data integrity, and access control. | |
| **Precondition** | 1. The EMS application must be installed and running. 2. The employee's account must be created and active within the EMS database, with valid credentials (employee ID and password).   3. The employee's computer or device must have network connectivity to access the EMS server.  A. The user has a valid account in the system.  B.The user knows their login credentials (username and password).  C.The system is online and accessible. | |
| **Basic Scenario** | **Actor Action** | **System Response** |
| **Step1:**User navigates to the login page.  **Step3:**User enters username and password.  **Step5:**User clicks the “Login” button.  **Step7:** User is authenticated. | **Step2:**System displays the login form (username, password fields).  **Step4:**System validates the entered credentials.  **Step6:**System checks account status and permissions.  **Step8:**System grants access to the appropriate dashboard and displays a welcome message. |
| **Alternative Scenario** | **Step8:**If credentials are invalid, system displays an error message (e.g., “Invalid username or password”).  **Step9:**User re-enters correct login credentials.  **Step10:**System re-validates and, if correct, grants access. | |
| **Post\_Conditon** | 1.User is logged in, session started, redirected to role-specific dashboard.(The employee is authenticated and authorized to access the EMS application according to their assigned roles and permissions. An audit log entry is created, recording the successful login event, including the employee ID, timestamp, and IP address. A session is created and stored either client side or on the server to maintain the login state.)  2. Failed login attempts are recorded for security.(Unsuccessful Login: The employee is not granted access to the EMS application. The system may implement account lockout policies and security monitoring measures to prevent unauthorized access. Failed login attempts are logged for security analysis. | |

**Logout**

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-3 | |
| **Use Case Name** | Logout | |
| **Actor** | Admin, Employee, Manager | |
| **Summary** | The **Logout** use case describes how a user securely exits the Employee Management System (EMS), terminating their session and ensuring no unauthorized access to their account once they leave the device or system. This process is essential for maintaining session security and data confidentiality. | |
| **Precondition** | 1.The user is currently logged into the EMS system.  2.The user has an active session.  3. The system is functioning properly and allows session control. | |
| **Basic Scenario** | **Actor Action** | **System Response** |
| **Step1:**User clicks the “Logout” button.  **Step3:**User confirms logout.  **Step5**User is redirected to login/home page.**:**  **Step7:**User sees confirmation message. | **Step2:**System prompts for logout confirmation (optional).  **Step4:**System terminates the user session.  **Step6:**System clears session data and cookies (if used).  **Step8:**System logs the logout event for audit purposes. |
| **Alternative Scenario** | **Step9:**If the user closes the browser or app without clicking "Logout  **a.**System auto-terminates session after a timeout period.  **b.**   |  | | --- | | User is required to log in again when accessing the system later. |   **Step10:**If logout request fails (e.g., due to server error):  **a**System displays an error message (e.g., "Logout failed").**.**  **b.**User can retry logout or close the session manually. | |
| **Post\_Conditon** | The user is securely logged out of the system.  All session data is cleared.  The system is ready for the next login attempt. | |

**Manage Attendance**

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-4 | |
| **Use Case Name** | Manage Attendance | |
| **Actor** | Admin, Manager, Employee (for viewing only) | |
| **Summary** | The **Manage Attendance** use case describes how the admin or manager records, updates, reviews, and monitors employee attendance in the EMS. This includes marking presence/absence, tracking working hours, approving time corrections, and generating attendance reports. Employees may have limited access to view their own attendance records. | |
| **Precondition** | 1. The admin or manager is logged into the system with proper permission 2. Attendance records are initialized in the system for the current period 3. Employee profiles exist and are active in the system. | |
| **Basic Scenario** | **Actor Action** | **System Response** |
| **Step1:**Admin/Manager navigates to the Attendance module.  **Step3:**Admin/Manager selects an employee or date range.  **Step5:**Admin/Manager marks attendance (present, absent, on leave, late, overtime) or edits incorrect records.  **Step7:**Admin/Manager submits or finalizes the attendance records. | **Step2:**System displays the attendance dashboard showing a list of employees and their daily/monthly attendance status  **Step4:**System displays detailed attendance records for the selected employee or period  **Step6:**System saves the updated attendance and recalculates related metrics (e.g., total hours worked, overtime, absences).  **Step8:**System confirms successful update and refreshes attendance summaries or reports. |
| **Alternative Scenario** | **Step8:**If the admin tries to edit a locked period (e.g., past payroll period):  System shows a warning message:   1. “Attendance records are locked for this period.” 2. Admin can request special access or unlock with higher permissions.   **Step9:**If there’s a system error (e.g., database issue):   1. System displays an error message: “Unable to save attendance changes. Please try again later.” 2. Admin can retry or contact support. | |
| **Post\_Conditon** | 1. Attendance records are saved, updated, and reflected in system reports. 2. Employee attendance summaries and payroll calculations are updated. 3. Employees (if permitted) can view their updated attendance status. | |

**View Attendance**

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-5 | |
| **Use Case Name** | View Attendance | |
| **Actor** | Employee, Manager, Admin | |
| **Summary** | The **View Attendance** use case allows employees, managers, and admins to access attendance records in the EMS. Employees can check their personal attendance details, while managers and admins can view attendance summaries for individuals, teams, or the entire organization. This improves transparency, helps employees monitor their performance, and supports managers in tracking team presence. | |
| **Precondition** | 1. The actor is logged into the EMS with the correct role and permissions. 2. Attendance data has been recorded in the system. 3. The system is online and operational. | |
| **Basic Scenario** | **Actor Action** | **System Response** |
| **Step1:**Actor logs into the EMS.  **Step3:**Actor navigates to the Attendance module.  **Step5:**Actor selects the date range or specific period to view.  **Step7:**Actor optionally filters or downloads the attendance report (if allowed). | **Step2:**System verifies login and shows the main dashboard.  **Step4:**System displays the attendance overview (calendar, table, or list view).  **Step6:**System retrieves and displays detailed attendance records (e.g., present, absent, late, leave).  **Step8:**System provides the filtered view or generates a downloadable report. |
| **Alternative Scenario** | **Step9:**If the system detects missing or incomplete attendance data:  a/.System shows a warning or message (e.g., “Attendance data is incomplete for the selected period.”).   1. Actor can notify the manager or admin to correct the records..   **Step10:**If the actor tries to access restricted records (e.g., an employee trying to view other employees’ data):  System shows an “Access Denied” message and restricts the view to personal records | |
| **Post\_Conditon** | 1. The actor successfully views the attendance details for the selected period. 2. Any data exports or reports (if available) are generated. 3. Sensitive or restricted data remains protected based on role permissions. | |

**Manage Department**

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-6 | |
| **Use Case Name** | Manage Department | |
| **Actor** | Admin | |
| **Summary** | The Admin creates, updates, or deletes department records in the system to organize employees under the appropriate departments. | |
| **Precondition** | 1. The Admin is logged into the system. 2. The Admin has the necessary permissions to manage departments. | |
| **Basic Scenario** | **Actor Action** | **System Response** |
| **Step1:**Admin selects **Manage Department** option.  **Step3:**Admin clicks **Add**, **Edit**, or **Delete** on a department.  **Step5:**Admin fills in or updates department details and submits. | **Step2:**System displays a list of existing departments and management options.  **Step4:**System displays the appropriate form or confirmation dialog.  **Step6:**System validates the input and saves the changes. |
| **Alternative Scenario** | **Step8:**Admin submits incomplete or invalid data.  **Step9:**System displays error message and prompts Admin to correct the input. | |
| **Post\_Conditon** | 1. The department information is updated in the system (added, modified, or deleted). 2. Related employee records are linked to the updated department information. | |

**Manage Leave**

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-7 | |
| **Use Case Name** | Manage Leave | |
| **Actor** | Employee, Manager | |
| **Summary** | Employees can submit leave requests, and managers can review, approve, or reject these requests. | |
| **Precondition** | 1. Employee or Manager is logged into the system. 2. Employee has available leave balance. 3. Manager has access rights to review leave requests. | |
| **Basic Scenario** | **Actor Action** | **System Response** |
| **Step1:**Employee selects **Request Leave** option.  **Step3:**Employee fills in leave details and submits.  **Step5:**Manager selects **Manage Leave Requests** option.  **Step7:**Manager reviews and approves/rejects the request. | **Step2:**System displays leave request form.  **Step4:**System validates the request and forwards it to the Manager.  **Step6:**System displays list of pending leave requests.  **Step8:**ystem updates leave status and notifies Employee. |
| **Alternative Scenario** | **Step8:**Employee submits incomplete or invalid leave request. |  **Step9:** System displays error message and prompts Employee to correct the request. | |
| **Post\_Conditon** | 1. Leave request is stored in the system with the updated status (pending, approved, rejected). 2. Employee and Manager are notified of the leave decision. 3. Leave balance is adjusted if the leave is approved. | |

**Manage Employee**

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-8 | |
| **Use Case Name** | Manage Employee | |
| **Actor** | Admin, Manager | |
| **Summary** | The Admin or Manager can add, update, view, or remove employee records to maintain accurate and up-to-date employee information in the system. | |
| **Precondition** | 1. Admin or Manager is logged into the system. 2. Admin or Manager has appropriate permissions to manage employee records. | |
| **Basic Scenario** | **Actor Action** | **System Response** |
| **Step1:**Admin/Manager selects **Manage Employee** option.  **Step3:**Admin/Manager clicks **Add**, **Edit**, or **Delete** on an employee record.  **Step5:**Admin/Manager enters or updates employee details and submits.  **Step7:**Admin/Manager receives success message. | **Step2:**System displays list of employees and management options.  **Step4:**System displays the employee form or confirmation dialog.  **Step6:**System validates the input and saves the changes. |
| **Alternative Scenario** | **Step8: 1** Admin/Manager submits incomplete or invalid employee data.  **2**.Admin/Manager attempts to delete an employee linked to active tasks or projects.  **Step9:**a. System displays error message and prompts Admin/Manager to correct the input.  b.System warns about dependency and prevents deletion or suggests reassigning tasks. | |
| **Post\_Conditon** | 1. Employee information is updated, added, or removed in the system. 2. Related records (like department, leave, attendance) remain consistent and updated. | |

**View Employee Record**

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-9 | |
| **Use Case Name** | View Employee Record | |
| **Actor** | Admin, Manager, Employee | |
| **Summary** | The Admin and Manager can view the records of any employee. Employees can view their own personal records, including profile details, leave history, and attendance. | |
| **Precondition** | 1. Actor is logged into the system. 2. Actor has permission to access the requested employee record. | |
| **Basic Scenario** | **Actor Action** | **System Response** |
| **Step1:**Actor selects **View Employee Record** option.  **Step3:**Actor searches or selects an employee from the list.  **Step5:**Actor views the employee details.  **Step7:**Actor may navigate through different sections (personal info, attendance, leave, performance). | **Step2:**System displays search/filter options (by name, ID, department, etc.).  **Step4:**System retrieves the employee record.  **Step6:**System displays personal info, job details, attendance, leave history, etc  **Step8:**System dynamically updates the displayed data. |
| **Alternative Scenario** | **Step8:**Actor searches for a non-existent or unauthorized employee record  **Step9:**System displays an error message or “No record found” / “Access denied” message. | |
| **Post\_Conditon** | 1. Employee record is displayed on the system interface 2. No data is modified during the view operation. | |

**Add Record**

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-10 | |
| **Use Case Name** | Add rRecord | |
| **Actor** | Admin | |
| **Summary** | This use case allows the Admin to add a new employee record to the system, including personal details, position, department, and biometric data. | |
| **Precondition** | Admin must be logged into the system with appropriate access rights | |
| **Basic Scenario** | **Actor Action** | **System Response** |
| **Step1:**Admin selects "Add New Record" from the dashboard.  **Step3:**Admin fills in personal and employment data.  **Step5:**Admin initiates biometric data capture.  **Step7:**Admin submits the complete form. | **Step2:**System displays a form to input employee details.  **Step4:**System validates the entered data.  **Step6:**System communicates with biometric device to collect data.  **Step8:**System saves the employee record and confirms success. |
| **Alternative Scenario** | **Step9:**If biometric device fails to connect or capture data, system notifies Admin with error message.  **Step10:**Admin may retry the biometric capture or proceed with a partial save. | |
| **Post\_Conditon** | 1. New employee record is stored in the system database. 2. Biometric data is associated with the employee profile if successfully captured. | |

**Update Record**

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-11 | |
| **Use Case Name** | Update Record | |
| **Actor** | Admin | |
| **Summary** | This use case allows the Admin to update the details of an existing employee record, such as name, position, department, or biometric informatio | |
| **Precondition** | Admin must be logged into the system and have access to modify employee records. The employee record to be updated must already exist. | |
| **Basic Scenario** | **Actor Action** | **System Response** |
| **Step1:**Admin navigates to the "Employee Records" section.  **Step3:**Admin selects the employee record to update.  **Step5:**Admin edits the necessary information.  **Step7:** Admin confirms the update | **Step2:**System displays a list or search option for employee records.  **Step4:**System retrieves and displays the current details of the employee.  **Step6:**System validates the updated data.  **Step8:**System saves changes and confirms update success. |
| **Alternative Scenario** | **Step8:**If validation fails (e.g., invalid format or missing data), system displays appropriate error message.  **Step9:**Admin corrects the data and resubmits the update. | |
| **Post\_Conditon** | 1. The selected employee’s record is updated in the system database. 2. If biometric data was changed, it is re-linked to the updated profile. | |

**Delete Record**

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-12 | |
| **Use Case Name** | Delete Record | |
| **Actor** | Admin | |
| **Summary** | This use case allows the Admin to remove an existing employee’s record from the system when the employee has resigned, retired, or the record is no longer needed. | |
| **Precondition** | Admin must be authenticated and authorized. The employee record must already exist in the system. | |
| **Basic Scenario** | **Actor Action** | **System Response** |
| **Step1:**Admin navigates to the "Employee Records" section.  **Step3:**Admin searches for and selects the employee record to delete.  **Step5:**Admin clicks the “Delete” button.  **Step7:**Admin confirms the deletion. | **Step2:**System displays a list or search interface for employee records.  **Step4:**System retrieves and displays the selected employee’s details.  **Step6:**System prompts a confirmation dialog to prevent accidental deletion.  **Step8:**System permanently deletes the record and displays a success message. |
| **Alternative Scenario** | **Step8:**Admin cancels the confirmation, the system aborts the deletion.  **Step9:**System returns to the employee record detail page without making any changes. | |
| **Post\_Conditon** | The employee’s record is permanently deleted from the system database and is no longer accessible. | |

**Manage Leave**

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-13 | |
| **Use Case Name** | Manage Leave | |
| **Actor** | Employee, Manager | |
| **Summary** | This use case allows employees to apply for leave and enables managers to approve or reject leave requests. | |
| **Precondition** | 1. Employee must be logged into the system. 2. Leave balance must be available. 3. Manager must be authenticated. | |
| **Basic Scenario** | **Actor Action** | **System Response** |
| **Step1:**Employee navigates to the "Leave Management" section.  **Step3:**Employee fills out the leave request form and submits it.  **Step5:.** Manager logs into the system and views pending leave requests.  **Step7:**Manager selects a request and chooses to approve or reject. | **Step2:**System displays available leave types and balances.  **Step4:**System records the request and forwards it to the appropriate Manager.  **Step6:**System displays list of pending leave requests.  **Step8:**System updates the leave status and notifies the employee. |
| **Alternative Scenario** | **Step9:**If the employee attempts to apply without available leave balance, the system blocks submission and displays an error.  **Step10:**If the manager does not act within a predefined period, the request is marked as "Pending Review" with a reminder sent. | |
| **Post\_Conditon** | The leave request is either approved, rejected, or marked pending. Employee leave records are updated accordingly. | |

**Calculate Payment**

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-14 | |
| **Use Case Name** | **Calculate Payment** | |
| **Actor** | Admin, External System (Biometric Device) | |
| **Summary** | This use case describes how the system calculates an employee’s salary based on attendance, allowances, deductions, and other payroll rules. | |
| **Precondition** | 1. Attendance records must be available from the biometric device. 2. Employee salary structure and deduction rules must be predefined. | |
| **Basic Scenario** | **Actor Action** | **System Response** |
| **Step1:**Admin initiates payroll processing.  **Step3:**Admin confirms the time period and selects employees.  **Step5:**Admin reviews any custom inputs (bonuses, deductions).  **Step7:**Admin reviews the final salary report. | **Step2:**System fetches attendance records from biometric device.  **Step4:**System retrieves employee salary structure and related data.  **Step6:**System applies all payment rules and computes gross and net salary.  **Step8:**System generates a salary statement and stores it in payroll history. |
| **Alternative Scenario** | **Step9:**If attendance data is incomplete or missing, the system alerts the Admin to correct it before proceeding.  **Step10:**If an employee is inactive or terminated, system excludes them from payroll and logs a note | |
| **Post\_Conditon** | Employee payments are calculated and recorded. Payroll summary and salary slips are generated and stored. | |

View Report

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-14 | |
| **Use Case Name** | View Report | |
| **Actor** | Admin, Manager | |
| **Summary** | This use case describes how an admin or manager views various reports such as attendance, leave, performance, and payroll. | |
| **Precondition** | 1. The user (Admin or Manager) must be logged into the system. 2. The required data must be available in the system database. | |
| **Basic Scenario** | **Actor Action** | **System Response** |
| **Step1:**Actor navigates to the reports section.  **Step3:**Actor selects the type of report to view (e.g., attendance, payroll).  **Step5:**Actor enters filter criteria and submits the request.  **Step7:**Actor views the generated report. | **Step2:**System displays available report categories.  **Step4:**System prompts for any filters (e.g., date range, department).  **Step6:**System retrieves and processes the requested data.  **Step8:**System displays the report in a readable format and provides options to export (PDF, Excel). |
| **Alternative Scenario** | **Step8:**If no data is found for the selected filters, the system displays a "No data available" message.  **Step9:**If the report generation fails, the system displays an error and logs the issue for admin review. | |
| **Post\_Conditon** | Report is displayed or exported successfully, or an appropriate message is shown if no data is found. | |

**View Profile**

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-15 | |
| **Use Case Name** | View Profile | |
| **Actor** | Employee | |
| **Summary** | This use case allows an employee to view their personal profile information including contact details, job title, department, and employment status. | |
| **Precondition** | 1. Employee must be logged into the system. 2. Profile data must already exist in the system database. | |
| **Basic Scenario** | **Actor Action** | **System Response** |
| **Step1:**Employee logs into the system.  **Step3:**Employee navigates to the "My Profile" section.  **Step5:**Employee selects to view full profile. | **Step2:**System verifies login credentials.  **Step4:**System retrieves profile data from the database.  **Step6:**System displays the complete profile information. |
| **Alternative Scenario** | **Step7:**If the profile data is not found, the system displays an error message such as “Profile not available.”  **Step9:**If system connection fails, it prompts the user to try again later. | |
| **Post\_Conditon** | Profile information is successfully displayed to the employee or appropriate error is shown. | |

**Submit Attendance**

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-16 | |
| **Use Case Name** | Submit Attendance | |
| **Actor** | Employee, External System (Biometric Device) | |
| **Summary** | This use case describes how an employee’s attendance is submitted either manually by the employee or automatically via a biometric device. | |
| **Precondition** | 1. Employee is registered in the system. 2. The system or biometric device is operational | |
| **Basic Scenario** | **Actor Action** | **System Response** |
| **Step1:**Employee places finger on biometric device OR logs in manually to submit attendance.  **Step3:**Employee selects the attendance submission option (if manual).  **Step5:-** Employee confirms attendance submission.  **Step7:**System updates the attendance record in the database. | **Step2:-** System reads biometric data or employee credentials.  **Step4:**System verifies identity from stored records.  **Step6:**System logs the current date and time.  **Step8:**System confirms attendance has been successfully recorded. |
| **Alternative Scenario** | **Step8:**If the biometric scan fails or credentials are incorrect, the system prompts to try again.  **Step9:**If the system is offline or cannot connect to the database, it stores attendance locally and syncs later. | |
| **Post\_Conditon** | Attendance is successfully submitted and stored in the system database, or queued for later if offline. | |

**Request Leave**

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-17 | |
| **Use Case Name** | Request Leave | |
| **Actor** | Employee | |
| **Summary** | This use case allows an employee to submit a request for leave through the system, specifying leave type, dates, and reason. | |
| **Precondition** | 1. Employee is logged into the system. 2. Employee has remaining leave balance (based on leave type). | |
| **Basic Scenario** | **Actor Action** | **System Response** |
| **Step1:**Employee logs into the system.  **Step3:**Employee navigates to "Request Leave" section.  **Step5:**Employee fills in leave type, start and end date, and reason.  **Step7:**Employee submits the leave request. | **Step2:**System verifies employee credentials.  **Step4:**System displays leave request form.  **Step6:**System validates the form and checks leave balance.  **Step8:**System stores request and notifies the manager for approval. |
| **Alternative Scenario** | **Step8:**If the leave balance is insufficient, the system displays an error message and blocks submission.  **Step9:**If the form is incomplete, the system prompts the employee to complete all required fields. | |
| **Post\_Conditon** | Leave request is stored and routed for managerial approval. | |

**View Leave Status**

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-18 | |
| **Use Case Name** | View Leave Status | |
| **Actor** | Employee | |
| **Summary** | This use case allows an employee to view the status of their submitted leave requests, including whether the request has been approved, rejected, or is pending. | |
| **Precondition** | The employee must be logged into the system and must have at least one leave request submitted. | |
| **Basic Scenario** | **Actor Action** | **System Response** |
| **Step1:**Employee navigates to the leave section.  **Step3:**Employee selects "View Leave Status."  **Step5:**: Employee views the list of submitted leave requests. | **Step2:**System displays the leave management options.  **Step4:**System fetches leave request records from the database.  **Step6:**System displays the status (approved/rejected/pending) for each request. |
| **Alternative Scenario** | **Step8:**If there are no leave requests found,  **Step9:**System displays a message: "No leave requests found." | |
| **Post\_Conditon** | The employee is informed of the status of their leave requests. | |

**Deny Leave**

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-19 | |
| **Use Case Name** | Deny Leave | |
| **Actor** | Manager | |
| **Summary** | The manager denies an employee's leave request. | |
| **Precondition** | The employee has submitted a leave request pending approval. | |
| **Basic Scenario** | **Actor Action** | **System Response** |
| **Step1:**Manager logs in  **Step3:**Manager views pending leave requests  **Step5:**Manager selects a specific request and chooses "Deny"  **Step7:**Manager confirms denial | **Step2:**System verifies credentials  **Step4:**System displays all pending leave requests  **Step6:**System updates the status to "Denied"  **Step8:**System notifies the employee |
| **Alternative Scenario** | **Step9:**If request is already approved/denied, system displays an error  **Step10:**Manager returns to dashboard | |
| **Post\_Conditon** | Leave status is updated to "Denied" and employee is notified. | |

**Approve Leave**

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-20 | |
| **Use Case Name** | Approve Leave | |
| **Actor** | Manager | |
| **Summary** | The manager approves an employee's leave request. | |
| **Precondition** | Leave request exists in pending state. | |
| **Basic Scenario** | **Actor Action** | **System Response** |
| **Step1:**Manager logs in  **Step3:**Manager opens leave requests page  **Step5:**Manager selects request to approve  **Step7:**Manager confirms | **Step2:**System verifies login  **Step4:**System displays all requests  **Step6:**System changes status to "Approved"  **Step8:**System sends notification to employee |
| **Alternative Scenario** | **Step8:**Leave request expired → System prevents approval  **Step9:**Manager gets a warning message | |
| **Post\_Conditon** | Leave request marked as "Approved"; employee informed. | |

**Generate Report**

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-21 | |
| **Use Case Name** | Generate Report | |
| **Actor** | Admin or Manager | |
| **Summary** | Admin or Manager generates a report on attendance, performance, or leave. | |
| **Precondition** | Actor is logged in with proper privileges. | |
| **Basic Scenario** | **Actor Action** | **System Response** |
| **Step1:**Actor navigates to reports section  **Step3:**Actor selects report type and filters  **Step5:**Actor clicks “Generate”  **Step7:**Actor downloads or prints report | **Step2:**System displays report options  **Step4:**System processes request  **Step6:**System generates and displays report  **Step8:**System logs report generation event |
| **Alternative Scenario** | **Step9:**Invalid filters → system prompts correction  **Step10:**No data found → system informs user | |
| **Post\_Conditon** | Report is generated and optionally downloaded/printed | |

**Authenticate Fingerprint**

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-22 | |
| **Use Case Name** | Authenticate Fingerprint | |
| **Actor** | External System (Biometric Device) | |
| **Summary** | Biometric device scans and verifies an employee's fingerprint. | |
| **Precondition** | Employee has enrolled fingerprint data in the system. | |
| **Basic Scenario** | **Actor Action** | **System Response** |
| **Step1:**Employee places finger on scanner  **Step3:**System matches image with stored data  **Step5:**If match, system logs user | **Step2:**System captures fingerprint image  **Step4:**System verifies identity  **Step6:**Access or attendance action is initiated |
| **Alternative Scenario** | **Step7:**No match found → system denies access  **Step8:**Scanner error → request retry | |
| **Post\_Conditon** | Fingerprint is authenticated or denied. | |

**Record Attendance**

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-23 | |
| **Use Case Name** | Record Attendance | |
| **Actor** | External System (Biometric Device) | |
| **Summary** | Employee attendance is recorded through biometric authentication. | |
| **Precondition** | Employee is successfully authenticated. | |
| **Basic Scenario** | **Actor Action** | **System Response** |
| **Step1:**Biometric device authenticates employee  **Step3:**Device sends timestamp  **Step5:**System updates attendance log. | **Step2:**System confirms identity  **Step4:**System records check-in or check-out time  **Step6:**System confirms with device |
| **Alternative Scenario** | **Step8:**Network failure → data queued locally  **Step9:**Duplicate entry → system prevents re-entry | |
| **Post\_Conditon** | Attendance log is updated. | |

Update Attendance History

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-24 | |
| **Use Case Name** | Update Attendance History | |
| **Actor** | Admin | |
| **Summary** | Admin manually corrects or updates an employee’s attendance records. | |
| **Precondition** | Admin is logged in and has appropriate permissions. | |
| **Basic Scenario** | **Actor Action** | **System Response** |
| **Step1:**Admin opens attendance module  **Step3:**Admin selects employee  **Step5:**Admin edits specific entry  **Step7:**Admin confirms changes | **Step2:**System displays list of employees  **Step4:**System shows attendance history  **Step6:**System validates and saves updates  **Step8:**System logs update and displays confirmation |
| **Alternative Scenario** | **Step8:**Admin cancels before saving → no change made  **Step9:**Invalid input → system shows error | |
| **Post\_Conditon** | Attendance record is updated and logged. | |