

# **Web-Based Employee Management System for TCU faculty at Taguig City University**

## **Team Members:**

Balmoja, Dorothy Zoe L.  
Gonzales, Hannah Vine D.  
Jimenez, Althea Jasmine J.  
Orcales, Marianne V.  
Tanabe, Denise Koyasha H.

## **Software Requirements Specification Document**

**Version: (1)**

**Date: (mm/dd/yyyy)**

## Table of Contents

<b>1. Introduction</b>	<b>4</b>
1.1 Purpose	4
1.2 Scope	4-5
1.3 Definitions, Acronyms, and Abbreviations	5
1.4 References	5-6
1.5 Overview	6
<b>2. The Overall Description</b>	<b>6</b>
2.1 Product Perspective	7
2.1.1 System Interfaces	7
2.1.2 Interfaces	7
2.1.3 Hardware Interfaces	8
2.1.4 Software Interfaces	8
2.1.5 Communications Interfaces	8
2.1.6 Memory Constraints	9
2.1.7 Operations	9
2.1.8 Site Adaptation Requirements	10
2.2 Product Functions	11
2.3 User Characteristics	12
2.4 Constraints	12-13
2.5 Assumptions and Dependencies	13-14
<b>3. Specific Requirements</b>	<b>14</b>
3.1 External interfaces	14-18
3.2 Functions	19-20
3.3 Performance Requirements	21
3.4 Logical Database Requirements	21-23
3.5 Design Constraints	23
3.5.1 Standards Compliance	24
3.6 Software System Attributes	24
3.6.1 Reliability	24-25
3.6.2 Availability	25
3.6.3 Security	25
3.6.4 Maintainability	25-26
3.6.5 Portability	26-27
3.7 External Actor Descriptions	27
3.7.1 Human actors	27
3.7.2 Hardware actors	28
3.7.3 Software system actors	28
3.7.4 Use-Case Descriptions	29
3.7.4.1 Use case 1	29

3.7.5 Sequence Diagram	30
3.7.6 Activity Diagram	31
3.7.7 Dataflow Diagram	32
<b>4. Supporting Information</b>	<b>33</b>

## 1. Introduction

The Web-Based Employee Management System for the Faculty at Taguig City University (TCU) is a digital platform designed to streamline and automate the management of faculty records and activities. This system aims to address the inefficiencies of manual processes by providing a centralized, secure, and accessible solution for managing leave applications, schedules, and other critical faculty-related data. The system will serve as a tool for both faculty members and administrative staff to ensure smooth and efficient operations within the college.

### 1.1 Purpose

The purpose of the Web-Based Employee Management System is to provide an efficient, user-friendly, and secure platform to manage faculty information and activities at TCU. By automating tasks such as schedule management, and leave applications, the system seeks to reduce manual paperwork, minimize errors, and improve the overall organization of faculty records. Additionally, it aims to enhance communication between faculty members and administrative staff, ensuring that faculty schedules align properly and any gaps or conflicts are promptly resolved. This Software Requirements Specification (SRS) is intended for the system's stakeholders, including project developers, administrative staff, and faculty members, to clearly define the system's features, goals, and functionality.

### 1.2 Scope

The Web-Based Employee Management System, referred to as "TCU Faculty EMS," is designed to:

1. **Identify and Address Key Administrative Needs:** The software will automate leave applications, and schedule management for faculty at Taguig City University. This will reduce reliance on manual processes.
2. **Facilitate Faculty Operations:** Faculty members will be able to clock in and clock out for attendance, submit leave requests, and view their personal and professional profiles.

3. **Provide Administrative Support:** Administrators will have access to comprehensive tools for managing schedules, resolving conflicts, and ensuring those substitute faculties are identified promptly when needed.
4. **Enhance Data Accessibility and Security:** The system will store faculty-related data securely, ensuring privacy while allowing authorized users to access relevant information from any device with internet access.

The Web-Based Employee Management System will provide an organized and secure platform, reducing operational inefficiencies, enhancing faculty productivity, and improving overall administrative performance.

### **1.3 Definitions, Acronym, and Abbreviation**

**EMS:** Employee Management System - A software solution used to manage employee-related data and operations.

**TCU:** Taguig City University - The institution where the system will be implemented.

**Admin:** Administrator - Authorized personnel responsible for managing and overseeing system operations.

**Faculty:** Teaching staff or professors employed under the CICT department.

**Leave Request:** A formal application submitted by faculty members to request leave.

**Schedule Management:** A process that includes assigning and organizing faculty teaching schedules.

### **1.4 References**

[1] IEEE Standard 830-1993: Recommended Practice for Software Requirements Specifications, IEEE, 1993.

[2] N. . Dumacil, C. . Servo, A. J. . Mahilum, J. D. . Bernardino, M. D. . Pecayo, and A. . Adovas, “Teachers’ Scheduling System for Novaliches High School”, AASgBCPJMRA, vol. 2, no. 1, Apr. 2020.

[3] G. Kukreti, “Employee management system SRS,” Scribd, <https://www.scribd.com/document/648091287/Employee-Management-System-SRS> (accessed Dec. 2024).

## 1.5 Overview

This Software Requirements Specification (SRS) document is organized as follows:

- **Section 1:** Provides an overview of the system, including the purpose, scope, and intended audience.
- **Section 2:** Describes the overall functionality and the specific requirements of the Web-Based Employee Management System, focusing on what the software will provide to its users.
- **Section 3:** Contains the detailed technical requirements, including functional, non-functional, and interface requirements. This section is aimed at the developers and technical personnel responsible for implementing the system.
- **Section 4:** Provides any additional supporting information, including appendices, definitions, and references.

By following this structure, readers will be able to quickly locate information relevant to their needs. Potential users and stakeholders should focus on Section 2 for an understanding of system capabilities, while developers and technical staff should refer to Section 3 for implementation details.

## 2. The Overall Description

This section provides a high-level overview of the Web-Based Employee Management System for the Faculty of Taguig City University. The system is designed to improve the administrative workflow, reduce errors, and enhance operational efficiency by automating faculty-related processes.

## 2.1 Product Perspective

The Web-Based Employee Management System (EMS) is an independent, self-contained solution that centralizes and automates faculty management tasks, replacing manual systems. The system serves as a critical component of the university's administrative operations, enhancing faculty leave application processes, and schedule management.

While the EMS does not integrate with third-party tools like Google Calendar, it includes built-in functionality for managing schedules and resolving conflicts efficiently. Compared to existing manual or legacy systems in similar educational institutions, the EMS offers better data security, accessibility, and user experience tailored for TCU's specific needs.

### 2.1.1 System Interfaces

The Web-Based EMS is a standalone system and does not interact with any external systems. All functionalities, including data management, leave requests, and schedule coordination, are self-contained within the application.

### 2.1.2 Interfaces

The EMS will feature a user-friendly **Graphical User Interface (GUI)** accessible via a web browser. Key aspects include:

- **Logical Interface:** The system will have separate login portals for faculty and administrators.
- **Optimized User Experience:** The GUI will include intuitive menus, dashboards, and forms for ease of use, with accessibility features to accommodate all users.
- **Responsive Design:** The system will ensure compatibility with desktop and mobile devices.

### 2.1.3 Hardware Interfaces

The system has no specialized hardware interface requirements. The EMS will operate on standard web servers and personal devices (e.g., desktop computers, laptops, tablets, and smartphones) with internet access. Users will interact with the system through a compatible web browser.

If additional hardware requirements arise during system deployment, they will be specified in the design documentation.

### 2.1.4 Software Interfaces

The Web-Based EMS requires specific software components to operate effectively. These include:

- **Web Server Software:** The system will use XAMPP, which includes Apache, to serve as the web server environment.
- **Database Management System:** The system will utilize MySQL as its database management system to store and manage faculty data securely. Communication with the database will occur through SQL queries.
- **Web Browsers:** The system will support modern web browsers, including Google Chrome and Microsoft Edge, ensuring broad accessibility for end-users.

### 2.1.5 Communications Interfaces

The system does not require any specific communication protocols for its operations. Communication between users and the system will occur over the internet through standard HTTP/HTTPS protocols.



### 2.1.6 Memory Constraints

There are no significant memory constraints for the Web-Based EMS. The system is expected to run efficiently on devices with standard configurations, including at least 4GB RAM and a compatible browser. The server hosting the application should meet minimum performance requirements for web applications.

### 2.1.7 Operations

The system will support the following operational requirements:

#### Modes of Operation:

- **Interactive Operation:** The EMS will be operational during regular office hours when faculty and administrators actively use the system for leave requests, schedule management, and other tasks.
- **Unattended Operation:** The system will remain available 24/7 for users to log in, review schedules, and submit requests outside office hours.

#### Data Backup and Recovery:

- Regular backups of faculty-related data will be scheduled automatically to prevent data loss.
- In case of system failure, recovery processes will ensure data integrity and minimal downtime.

#### Processing Support:

- **Manual Validation of Data Inputs:** All data inputs, such as leave requests, will be reviewed and validated manually by administrators or HR personnel to ensure accuracy and compliance.
- Real-time updates of schedules and user records to ensure accuracy.\

### 2.1.8 Site Adaptation Requirements

The system requires the following adaptations prior to deployment:

**Database Initialization:** The MySQL database must be installed and configured on the server hosting the EMS. Required data tables must be created and populated with initial data (e.g., faculty records, administrator accounts).

**Web Server Setup:** A web server capable of hosting the EMS, such as Apache (included in XAMPP), must be installed and properly configured to support the application's requirements. Ensure the server has adequate storage capacity and meets the minimum specifications for stable performance under expected user load.

**Network Configuration:** A stable and secure internet connection must be available to facilitate uninterrupted access to the EMS for authorized users. Local Area Network (LAN) configuration may be required to ensure secure communication between the server and authorized users within TCU's network infrastructure. Firewalls and security protocols must be updated to allow the EMS to operate while protecting against unauthorized access.

**User Environment:** Faculty and administrative personnel must have access to devices capable of running a compatible web browser (e.g., Google Chrome, Microsoft Edge) with a minimum of 4GB RAM. Proper training sessions and documentation must be provided to ensure users are familiar with the system's functionalities and can navigate it efficiently.

## **2.2 Product Functions**

The Web-Based Employee Management System (EMS) for the Faculty at Taguig City University (TCU) is designed to make administrative tasks easier and faster. The system will provide these main features:

### **1. Secure Login:**

- Faculty and administrators will have their own accounts.
- Access to features will depend on the user's role (e.g., only administrators can approve leaves).

### **2. Leave Applications:**

- Faculty members can apply for leave online.
- Administrators can approve or reject leave requests.
- Notifications will keep users updated about their leave status.

### **3. Schedule Management:**

- Faculty can check their schedules.
- Administrators can create and adjust schedules to avoid conflicts.
- Substitute faculty can be assigned when needed.

### **4. Profile Updates:**

- Faculty can update their personal and professional details.
- Administrators can manage and keep faculty profiles accurate.

### **5. Reports and Analysis:**

- Generate reports on leave applications and faculty schedules.

## 2.3 User Characteristics

The people who will use the EMS are faculty members and administrative staff at TCU. Their general traits are:

### 1. Faculty Members:

- **Education:** College degree or higher.
- **Technical Skills:** Familiar with basic computer use and web applications.
- **Experience:** Mixed levels, from new teachers to experienced ones.

### 2. Administrative Staff:

- **Educational Level:** Bachelor's degree in management, information systems, or related fields.
- **Technical Expertise:** Moderate to advanced proficiency in using administrative tools, databases, and reporting systems.
- **Experience:** Experienced in administrative processes such as leave approvals, schedule management, and faculty record-keeping.

The design of the EMS will accommodate varying levels of technical expertise through:

- Intuitive user interfaces with clear navigation.
- Help documentation and support for new users.

## 2.4 Constraints

The development of the EMS must consider the following limitations:

### 1. Rules and Regulations:

- The system must follow data privacy laws like the Philippine Data Privacy Act (RA 10173).

2. **Hardware Needs:**

- The system should work smoothly on TCU's standard hardware, such as computers with at least 4GB of RAM.

3. **Limited Integration:**

- The system will not connect to other tools like Google Calendar.

4. **Multiple Users:**

- The system must support many users at the same time without slowing down.

5. **Tracking Changes:**

- Important actions, like approving leaves or updating schedules, must be logged.

6. **Security:**

- Only authorized users can perform sensitive tasks.
- Data must be protected during storage and transfer.

7. **Reliability:**

- The system should be available 99.9% of the time to ensure accessibility.

**2.5 Assumptions and Dependencies**

The following factors affect the system's design and requirements:

**Operating System:** The system assumes a modern server environment, like XAMPP with MySQL, will be used.

**User Skills:** Users are expected to know how to use web browsers and fill out online forms.

**Internet Access:** Reliable internet connections are necessary for both faculty and administrators.

**Initial Data:** TCU will provide the necessary data, like faculty records and schedules, before the system is launched.

**Hardware Setup:** Adequate servers and computers must be ready to support the system.

### 3. Specific Requirements

This section outlines the software requirements in sufficient detail to allow developers to design the system and testers to verify its functionality. Each requirement is externally perceivable by users, operators, and is organized for clarity and traceability.

#### 3.1 External Interfaces

This section provides a detailed description of all inputs and outputs for the Web-Based Employee Management System for the Faculty at Taguig City University (TCU). While the system is standalone and does not interact with external systems, this section describes the internal interfaces developers can use when working within the system.

It contains both content and format as follows:

##### Name of Item

- **Web-Based Employee Management System for the Faculty at Taguig City University**

##### Description of Purpose

- The internal interfaces define how developers interact with the system's components. These interfaces handle operations such as employee record, schedule management, and leave management. Developers use these interfaces to ensure smooth data flow and maintain modularity in the system.

### Source of Input

- Data entered via system forms (e.g., add or update faculty details).
- Data files uploaded by authorized users.

### Destination of Output

- System-generated or managing reports.
- Downloadable reports in formats like PDF.

### Valid Range, Accuracy, and/or Tolerance

- **Username:**
  - Format: Alphanumeric (letters and numbers only, no special characters except underscores).
  - Length: Minimum 5 characters, maximum 20 characters.
- **Password:**
  - Format: Must include a combination of uppercase, lowercase, numbers, and special characters.
  - Length: Minimum 8 characters, maximum 20 characters.
  - Accuracy: The password must meet complex requirements (at least one uppercase letter, one number, and one special character).
- **Dates:**
  - Format: ISO 8601 (YYYY-MM-DD).
- **Time Fields:**
  - Format: HH:MM for schedules and attendance tracking.

### Units of Measure

- **Leave Balance:** Days (integer values).
- **Time:** Measured in hours and minutes (HH:MM).

## Timing

- **System Response Time:**
  - Operations, such as scheduling faculty tasks, should be completed within 1 to 5 seconds.
  - Adding or updating data should be completed within 1 to 5 seconds.
- **Report Generation:**
  - Generating small reports (e.g., attendance for one faculty member in a single day record) may take 1-5 seconds.
  - Managing reports like searching, saving, and filtering may take 1-5 seconds.

## Relationships to Other Inputs/Outputs

- **Faculty Records:** Feed into other modules including:
- **Leave Management:** To track leave balances and approve/reject leave requests.
- **Schedule Management:** For managing daily schedules of faculty members.
- **Outputs:** This includes generated reports and dashboards.

## Screen Formats/Organization

- For user interactions, the system has organized screens:

### **Employee Management Screen:**

- Input forms for adding or updating faculty data.
- A table that lists faculties with filtering options.

### **Leave Management Screen:**

- Approve/Reject leave applications.
- Track leave balances.

### **Schedule Management Screen:**

- Interface to manage and log faculty schedules.



## Window Formats/Organization

- **Date Input and Management:**
- **Pop-up Modals:** For adding or editing data without leaving the main dashboard.
- **Dedicated Sections:** Separate screens for reports.

## Data Formats

- **Input Format:** Data is submitted using PHP to handle server-side operations and MySQL to store the data.

### Example (Adding Employee in PHP/MySQL):

```
$employee = [
    "employee_id" => 123,
    "name" => "John Doe",
    "position" => "Professor",
    "salary" => 45000.00,
    "leave_balance" => 12
];

// Sample query to insert data into MySQL database
$sql = "INSERT INTO employees (employee_id, name, position, salary, leave_balance)
VALUES (?, ?, ?, ?, ?)";
$stmt = $mysqli->prepare($sql);
$stmt->bind_param("issdi", $employee['employee_id'], $employee['name'],
$employee['position'], $employee['salary'], $employee['leave_balance']);
$stmt->execute();
```

- **Output Format:** Responses are also formatted by PHP and XAMPP.

### Example (Success Response in PHP):

```
$response = [
    "status" => "success",
    "message" => "Employee added successfully."
];
```

- **Report Outputs:** Generated reports are downloadable in PDF.

## Command Formats

- Standard commands are supported by the system to manage data modification internally:

**Add Employee Record:** Action: Submit a new employee record in the system form.

**Update Attendance:** Action: Update attendance for a specific date.

## End Messages

The system provides internal feedback messages to guide the developer or end-user:

- **Success Messages:**

```
<?php  
  
$response = array(  
  
    "status" => "success",  
  
    "message" => "Task Scheduling Successful"  
  
); ?>
```

- **Error Messages:**

```
<?php  
  
$response = array(  
  
    "status" => "error",  
  
    "message" => "Error Scheduling"  
  
);  
  
?>
```

### **3.2 Functions**

The main functions that must be performed in the Web-Based Employee Management System for the Faculty at Taguig City University (TCU) are outlined in the functional requirements. These requirements describe how inputs will be received, processed, and outputs will be produced by the system.

#### **System Functionalities**

The system shall:

##### **Validity Checks on the Inputs:**

- Ensure correct data formats for faculty details (names, ID numbers, contact information).
- Ensure all input fields follow the required format (text, numbers, dates).
- Validate login credentials using a secure authentication mechanism for faculty members and administrators.
- Check for duplicate or missing records during data entry for faculty profiles, schedules, and leave records.

**Exact Sequence of Operations:** The system shall follow a defined sequence of operations to maintain workflow consistency:

##### **Faculty Members:**

- Log in to the system.
- View personal profiles, schedules, and leave records.
- Submit leave requests or update schedule data.

##### **Administrators:**

- Log in to the system.
- Manage faculty records, including adding, updating, and deleting information.
- Generate and export reports (e.g., schedule summaries, leave balances).

- Update system parameters such as leave or working hours.

**Responses to Abnormal Situations:**

**Overflow:** Handle excessive data entries (e.g., bulk upload of records).

**Communication Facilities:** Notify users of server or connection issues.

**Error Handling and Recovery:**

- Display error messages for invalid inputs or failed operations.
- Log errors in the system for administrative review.
- Provide a mechanism to recover incomplete transactions or data.

**Effect of Parameters:** Enable system parameters to be configured by administrators, including:

- Maximum number of leave days per faculty member.
- Working hours and schedules.
- Faculty and administrator authority to access specific data.

**Relationship of Outputs to Inputs:** The system shall process inputs and generate appropriate outputs, including:

**Input/output Sequences:**

- Input of schedule records > Generate schedule summaries.
- Submit leave requests > Update leave balances and notify administrators.

**Formulas for Input to Output Conversion:**

- Calculate total working hours or overtime based on input schedules.
- Deduct leave days based on approved leave requests.

### **3.3 Performance Requirements**

This subsection specifies the numerical requirements for system performance under both normal and limited hardware conditions, such as a PC with 4GB RAM.

**1. User Capacity:**

- The system shall support up to 50 simultaneous users on a PC with 4GB RAM.

**2. Transaction Speed:**

- 90% of all transactions shall be processed in under 2 seconds on a PC with 4GB RAM.

**3. Data Handling:**

- The system shall manage up to 10,000 faculty records.
- The system shall process up to 200 leave requests per day.

**4. Availability:**

- The system shall maintain 95% uptime on limited hardware.

**5. Scalability:**

- The system shall support future upgrades to accommodate up to 500 simultaneous users on higher-spec machines.

### **3.4 Logical Database Requirements**

This section specifies the logical database requirements for the Web-Based Employee Management System for the Faculty at Taguig City University (TCU). The requirements focus on the data to be stored, its structure, and the relationships that govern the system's operation.

## **Types of Information Used by Various Functions (Data Dictionary):**

The database will manage the following key types of information:

- **Faculty Information:** Personal and professional details such as faculty ID, full name, department, position, and contact details.
- **Leave Records:** Details of leave applications, including request ID, faculty ID, leave type, start date, end date, approval status, and comments from the administrator.
- **Schedule Logs:** Records of faculty schedules, capturing assigned courses, time slots, and associated faculty IDs.
- **Schedules:** Data about assigned teaching schedules, including schedule ID, faculty ID, course details, and time slots.

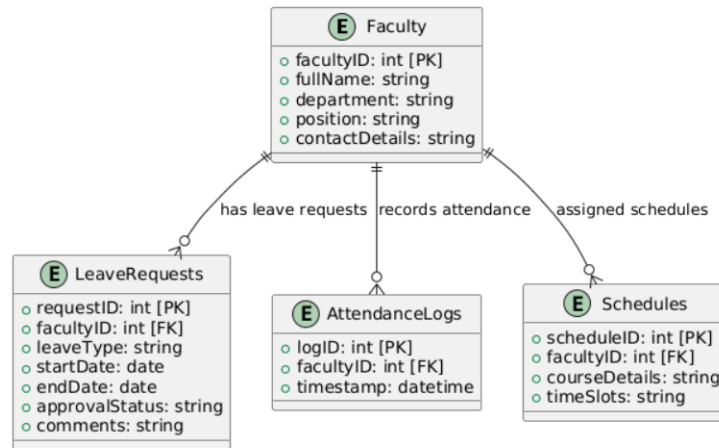
Each data type will be described in a data dictionary, specifying attributes such as data type, size, constraints, and relationships.

## **Data Entities and Their Relationships:**

Key data entities and their relationships include:

- **Faculty:** Central to the system, this entity connects to leave requests, schedules, and schedule logs.
- **Leave Requests:** Linked to the Faculty entity through the faculty ID, tracking individual leave applications.
- **Schedules:** Associated with the Faculty entity, representing teaching assignments and time slots.

An Entity-Relationship (ER) diagram will visualize these relationships, showing how faculty members interact with leave applications, schedules, and schedule logs.



### Integrity Constraints:

The system will enforce the following integrity constraints:

- **Primary Key Constraints:** Unique identifiers such as faculty ID, request ID, and schedule ID to ensure no duplicate records exist.
- **Foreign Key Constraints:** Referential integrity between related entities (e.g., faculty ID in Leave Requests referencing the Faculty entity).
- **Validation Rules:** Ensure data correctness, such as preventing overlapping leave dates or invalid time slots in schedules.

### Data Retention Requirements:

Retention policies will be defined for efficient database management and regulatory compliance:

- **Leave Records:** Stored for a minimum of five years for auditing and reference purposes.
- **Schedule Logs:** Retained for one academic year before archiving.
- **Schedules:** Maintained for the duration of the academic year and archived afterward.

### 3.5 Design Constraints

This section outlines the constraints influencing the design of the Web-Based Employee Management System for the Faculty at Taguig City University (TCU). These constraints may arise from existing standards, hardware limitations, or institutional policies.

### 3.5.1 Standards Compliance

The EMS will adhere to the following standards and regulations to ensure consistency, reliability, and compliance:

- **Report Format:** All system-generated reports will follow the university's formatting standards, ensuring readability and professionalism. This includes standardized headers, footers, fonts, and layouts for official documents like leave summaries and schedule reports.
- **Data Naming:** Database field names will adhere to a consistent and descriptive naming convention (e.g., "faculty\_id" for the faculty identifier) to promote clarity and maintainability.
- **Accounting Procedures:** Any financial aspects, such as faculty reimbursements or payroll adjustments, will align with institutional accounting standards. The system will accurately record and track financial data, ensuring compliance with TCU's financial policies.
- **Audit Tracing:** Changes to critical data (e.g., leave requests or schedules) will be logged, capturing details like the user making the change, timestamps, and before-and-after values. This ensures traceability and accountability.
- **Regulatory Standards:** The system will comply with the Data Privacy Act of 2012 in the Philippines, ensuring that faculty data is securely stored and accessible only to authorized personnel.

By incorporating these constraints, the EMS will deliver a robust, compliant, and user-centric platform for TCU's faculty and administrative staff.

## 3.6 Software System Attributes

### 3.6.1 Reliability

- The system is designed to meet a high level of reliability to support continuous operations:
  - Under typical operating conditions, the system seeks to achieve a **Mean Time between Failures (MTBF)** of 50 hours. This guarantees that there won't be any



unforeseen disruptions to the system's regular operation during regular business hours.

- Error-handling procedures have been implemented to ensure that minor malfunctions (such as incorrect input or inconsistent data) do not result in a system crash.
- Up to 50 people can access the system simultaneously without experiencing any performance issues, according to testing.

### 3.6.2 Availability

- Built to ensure reliable access for faculty and administrative staff:
  - Throughout the school year, the system will be available around-the-clock with a planned uptime of **99%**.
  - To ensure that faculty operations are not severely disrupted in the event of a failure, the system must be recoverable within **30-60 minutes**.
  - After a failure, users will be able to restart the application with the current operation still running or a maximum of **10 characters** of input lost.

### 3.6.3 Security

- Security is essential for preventing unauthorized access, use, or alteration of faculty data:
  - To guarantee safe data transfer, **HTTPS** will be used for all user-system communications.
  - The administrative and faculty credentials are secured with **encrypted passwords**.
  - To ensure accountability, **audit logs** will be kept to monitor user activity, such as leave requests, data modifications, and login attempts.
  - Data integrity checks are put in place to stop important faculty-related variables from becoming corrupted while procedures are underway.

### 3.6.4 Maintainability

- Developed with maintainability in mind to facilitate easy updates and long-term support:

- The system is **modular** in nature, with separate modules for scheduling, leave requests, and other essential features. This makes it possible to apply updates or repairs separately without affecting the entire system.
- Standard programming practices and thorough documentation of code, database structures, and workflows reduce code complexity.
- Error logs are used to assist with effective problem diagnosis and minimize downtime while troubleshooting.
- Developer documentation will be made available to help future maintenance teams comprehend and expand the capabilities of the system.

### 3.6.5 Portability

- Designed to operate across various platforms and devices to ensure flexibility and ease of deployment:
  - Only a web browser (such as Google Chrome, Mozilla Firefox, or Microsoft Edge) and internet access are needed for the system to function, making it platform-independent.
  - The system is compatible with common web servers and operating systems because it uses **MySQL** for the backend and **HTML5, CSS, and JavaScript** for the frontend.
  - Less than 5% of the codebase's components need host-specific changes, meaning that the majority of it remains host-independent.
  - The system will be set up on only the **local servers**.
  - Without the need for platform-specific customization, the **responsive design** guarantees that the system functions flawlessly on **PCs, laptops, tablets, and smartphones**.

ID	Characteristic	H/M/L	1	2	3	4	5	6	7	8	9	10	11	12
1	Correctness													
2	Interoperability													
3	Flexibility													
4	Integrity/Security													
5	Efficiency													
6	Maintainability													
7	Reliability													
8	Portability													
9	Availability													
10	Reusability													
11	Usability													
12	Testability													

### 3.7 External Actor Descriptions for Web-Based Employee Management System

#### 3.7.1 Human Actors

##### 1. Faculty Members:

- Primary users of the system who perform actions such as logging schedules, submitting leave applications, and viewing their schedules and profiles.
- Faculty members rely on the system for accurate record-keeping and quick responses to their requests.

##### 2. System Administrator:

- Oversees system maintenance, manages user accounts, performs backups, and troubleshoots technical issues.
- Ensures data security and compliance with university policies.
- Responsible for managing faculty records, approving or rejecting leave requests, resolving schedule conflicts, and assigning substitute faculty when needed.
- Administrators ensure the system operates smoothly by maintaining accurate data and addressing operational concerns.

### 3.7.2 Hardware Actors

#### 1. User Devices:

- Faculty and Administrative Devices: These include desktops, laptops, tablets, and smartphones that users employ to access the system via a web browser.
- Minimum specifications include:
  - Internet-enabled device with a compatible web browser.

#### 2. Local Servers:

- Hosts the web-based application and database, providing secure storage and processing of faculty-related data.
- Ensures system availability and efficient handling of concurrent requests from multiple users.

#### 3. Networking Equipment:

- Routers, switches, and internet connections that facilitate communication between user devices and the server hosting the system.

### 3.7.3 Software System Actors

#### 1. Web Browsers:

- Faculty and administrative staff use browsers such as Google Chrome, Mozilla Firefox, or Microsoft Edge to access the system.
- The system's responsive design ensures compatibility with these browsers for both desktop and mobile usage.

#### 2. Operating Systems:

- The system is platform-independent and compatible with standard operating systems like Windows.

#### 3. Database Management System (DBMS):

- The system uses **MySQL** to store, retrieve, and manage faculty data, including leave requests, schedules, and schedule logs.
- The DBMS ensures data integrity and supports automated backups for reliability.

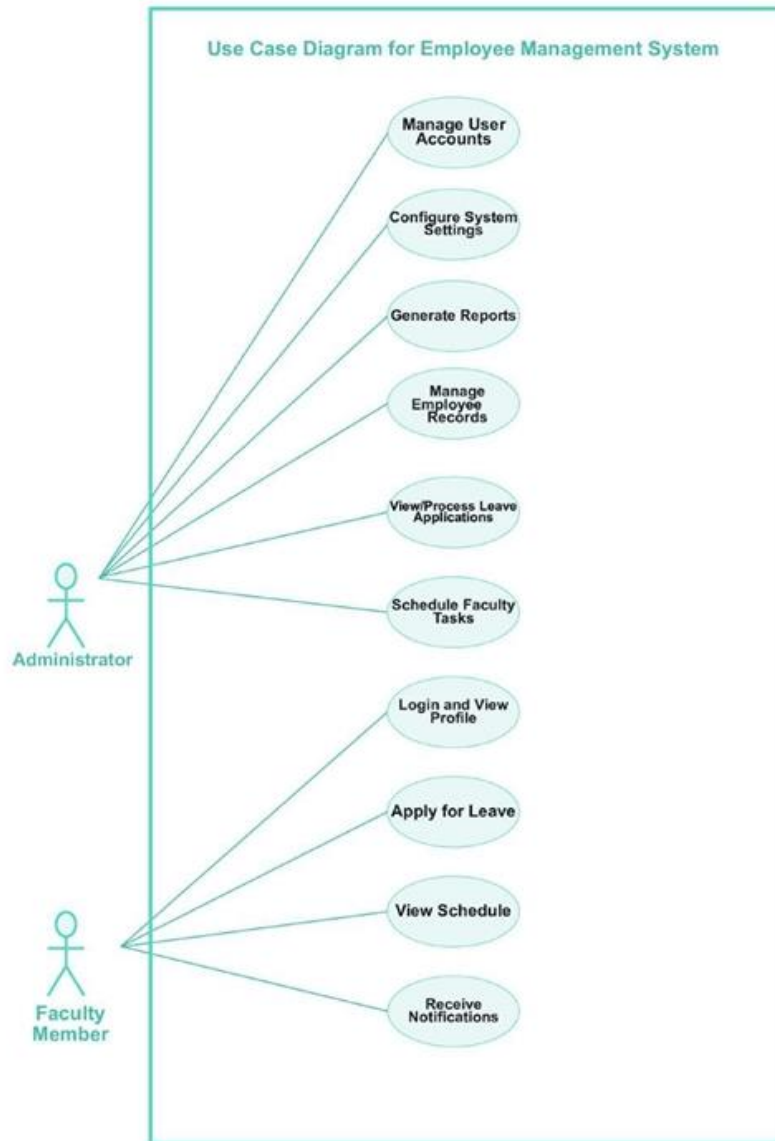
#### 4. Security Middleware:

- Handles user authentication (e.g., encrypted login credentials) and secure communication through **HTTPS**, ensuring data confidentiality and integrity.

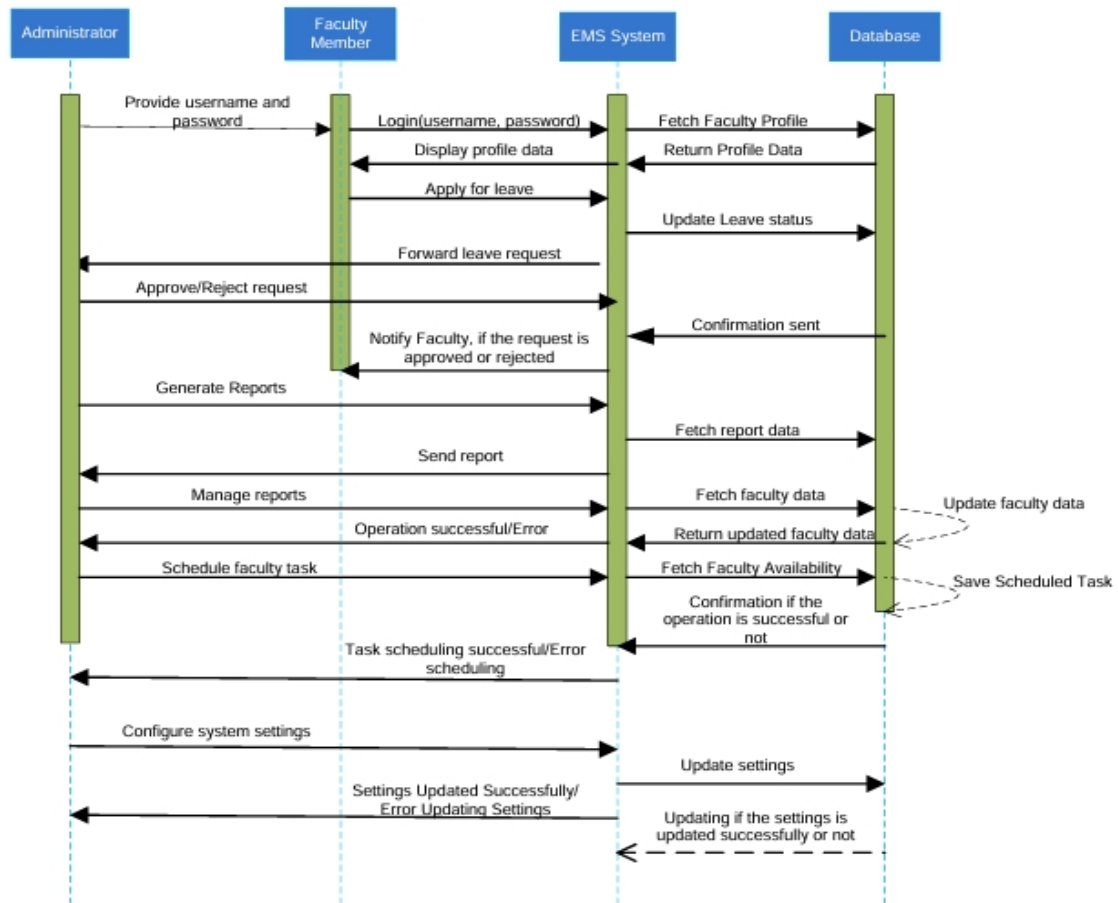
#### 5. Backup and Recovery Tools:

- Automated tools or scripts manage periodic backups of the system's database to safeguard against data loss and support disaster recovery efforts.

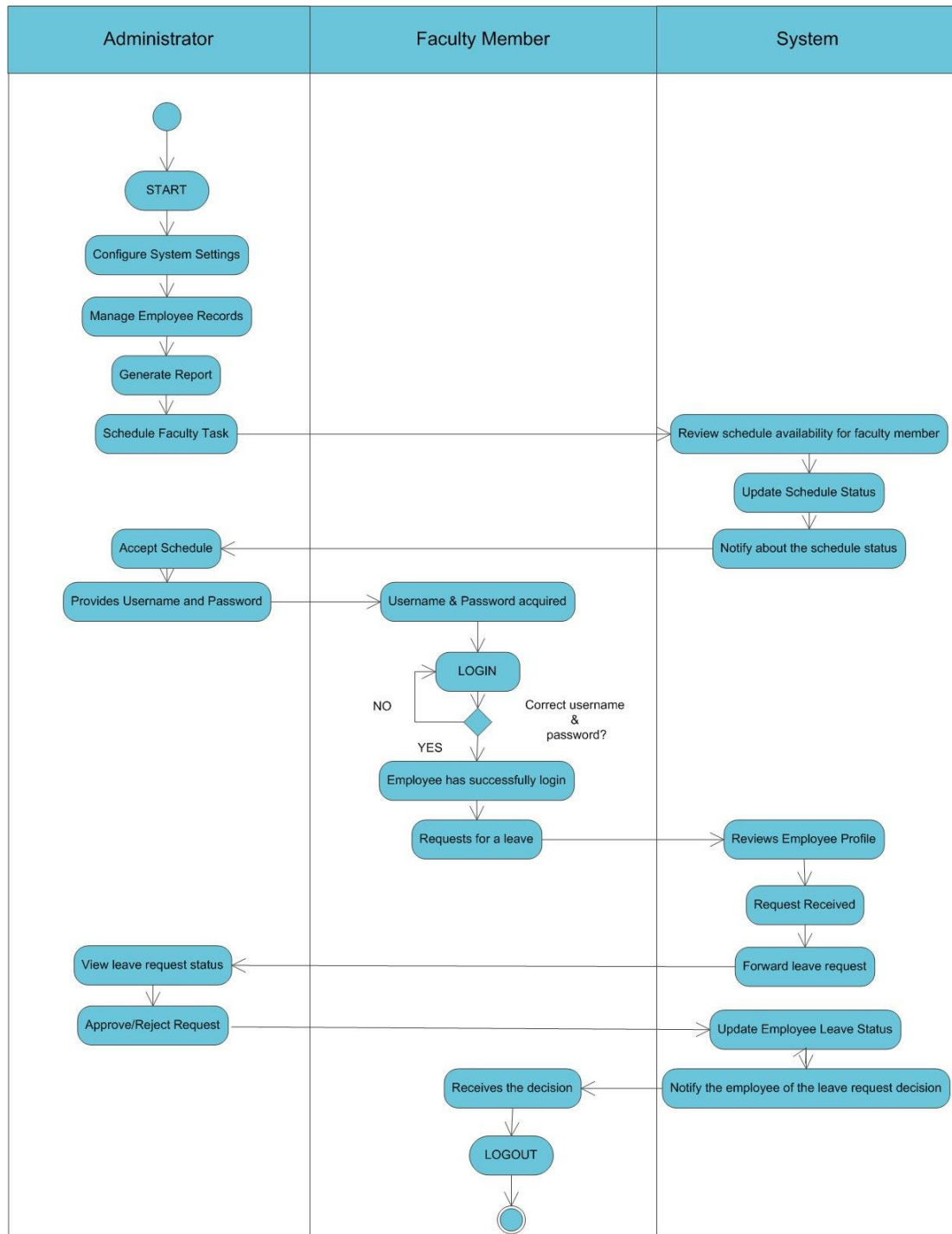
### 3.7.4 Use-Case Descriptions



### 3.7.5 Sequence Diagram



## 3.7.6 Activity Diagram



### 3.7.7 Dataflow Diagram

#### Context Diagram

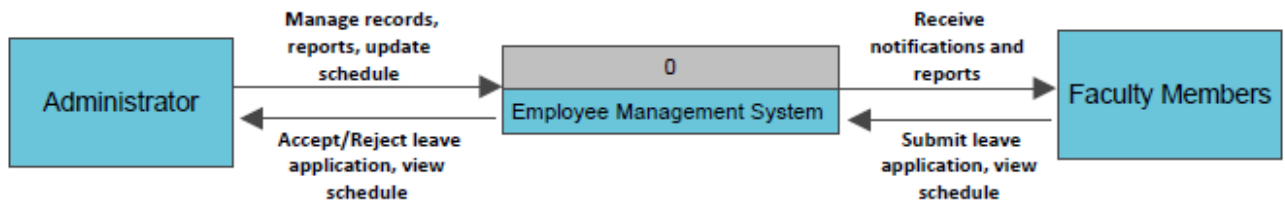
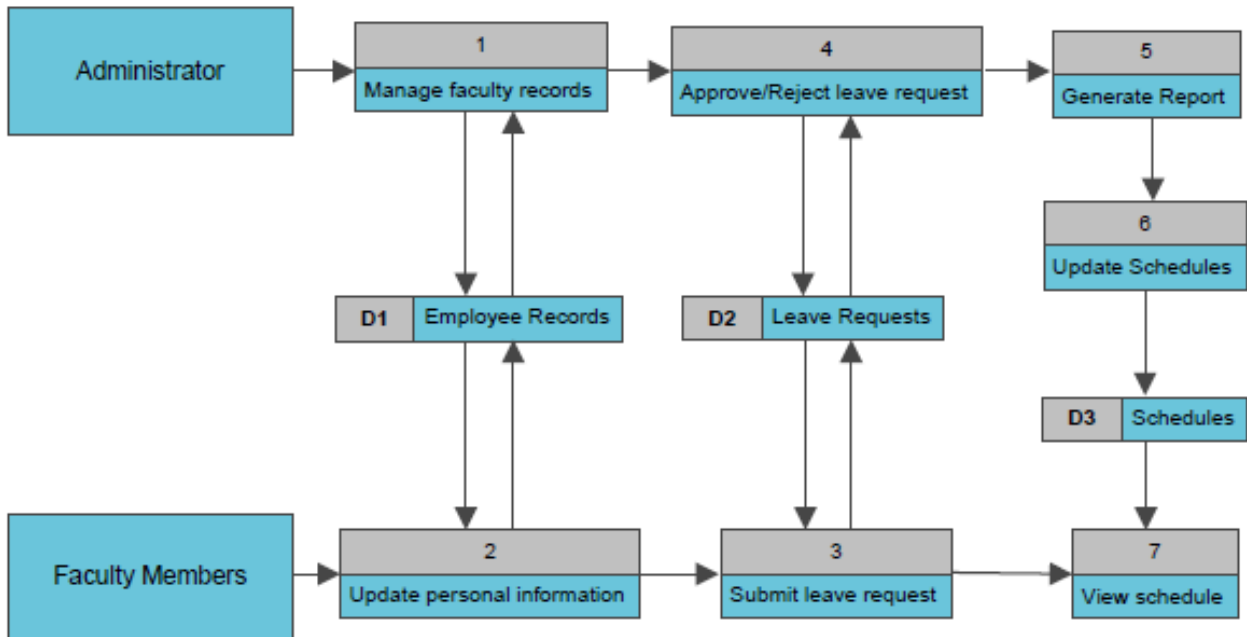


Diagram 0 DFD





## **4. Supporting Information**