**Time Management Software**

**Software Requirements Specification**

**Version<3.0>**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| <04/02/25> | <1.0> | SRS 1.0 | Group-2 |
| <11/02/25> | <2.0> | SRS 2.0 | Group-2 |
| <04/03/25> | <3.0> | SRS 3.0 | Group-2 |

**Table of Contents**

1. Introduction

1.1 Purpose

1.2 Scope

1.3 Definitions, Acronyms, and Abbreviations

1.4 References

1.5 Overview

2. Overall Description

2.1 Product Perspective

2.2 Product Features

2.3 User Classes and Characteristics

2.4 Operating Environment

2.5 Design and Implementation Constraints

2.6 Assumptions and Dependencies

3. Functional Requirements

3.1 User Registration and Authentication

3.2 Schedule Management

3.3 Automatic Meeting Slot Detection

3.4 Email Notifications

3.5 Leave and Engagement Management

3.6 Conflict Resolution

3.7 Graphical User Interface (GUI)

3.8 Remote Access

3.9 Statistics and Reporting

3.10 System Integration

4. Non Functional Requirements

4.1 Usability

4.2 Reliability

4.3 Performance

4.4 Security

4.5 Scalability

4.6 Compatibility

4.7 Availability

4.8 Maintainability

4.9 Supportability

4.10 Portability

5. Data Dictionary

5.1 User Registration and Authentication

5.2 Schedule Management

5.3 Meeting Slot Detection

5.4 Email Notifications

5.5 Engagement & Leave Management

5.6 Statistics & Reporting

5.7 System Integration

6. Data Flow Diagrams

6.1 Level 0 DFD

6.2 Level 1 DFD

7. ER Diagram

8. Structure Chart

9. Use Case Diagram

9.1 Time Management Software System

9.2 Schedule Meeting

10. Class Diagram

11. Activity Diagram

**Software Requirements Specification**

### 1. Introduction

The introduction of the Software Requirements Specification (SRS) provides an overview of the Time Management Software (TMS) by detailing the purpose, scope, definitions, acronyms, abbreviations, references, and an overview of the document. The aim of this document is to define the problem statement for TMS, analyzing the software's requirements and capabilities. It will focus on the needs of the stakeholders, such as executives and secretaries, and how the system can be designed to efficiently manage time for its users.

This document outlines the core functionalities that the TMS should provide, which include scheduling meetings, managing appointments, finding open slots for multiple executives, sending email notifications, and tracking time spent on meetings. Furthermore, the software must handle both individual and collective scheduling needs while offering intuitive, easy-to-use features suited for executives with varying levels of computer proficiency. The TMS should also generate statistics on time allocation for meetings and projects, assisting executives in understanding their time management patterns.

The detailed requirements of the TMS, including both functional and non-functional aspects, are discussed in the subsequent sections. These requirements are intended to ensure that the software aligns with the needs of the company, helping executives and secretaries maximize their time efficiency through seamless scheduling, monitoring, and reporting.

#### 1.1 Purpose

The purpose of this document is to outline and analyze the requirements for the Time Management Software (TMS). It defines the functionalities needed by executives and secretaries to manage their schedules, register meetings, and receive notifications. The goal is to present a clear understanding of the system’s objectives, intended users, and the interactions the software will support. This document serves to provide guidance for the design, development, and implementation of the system by detailing its functional and non-functional requirements.

In short, this SRS aims to provide a comprehensive overview of the TMS, including its user interface, functionalities, hardware and software requirements, and how the product will be used by stakeholders to achieve better time management.

#### 1.2 Scope

The scope of this SRS document pertains to the development of the Time Management Software (TMS), which will be used primarily by executives and secretaries. It outlines the key features of the system, which include the ability to register daily schedules, organize meetings, send email notifications, track leave and personal engagements, and generate statistical reports on time allocation. TMS will be designed to ensure that executives can efficiently manage their schedules and stay informed about upcoming meetings and appointments.

The system will support multiple users, handle scheduling conflicts, and provide a graphical user interface (GUI) to ensure ease of use. Additionally, it will allow for remote access, integration with email systems for notifications, and real-time updates of schedules.

This document also serves as a guide for the software’s development, detailing the functional and non-functional requirements, constraints, and assumptions that will guide the software's design and implementation.

#### 1.3 Definitions, Acronyms, and Abbreviations

|  |  |
| --- | --- |
| TMS | Time Management Software |
| GUI | Graphical User Interface |
| Email Notification | Automated email reminders sent to executives regarding their schedules |
| Man-Hours | Total hours spent by all executives in meetings |
| Executive | A user who schedules and attends meetings |
| Secretary | A user who manages and adjusts executive schedules |

#### 1.4 References

The references for the TMS are:

* Company Policies on Time Management
* Industry Standards for Scheduling Software
* User Interface Design Guidelines

#### 1.5 Overview

The remaining sections of this document provide a comprehensive description of the TMS, including user characteristics, system functionalities, constraints, and assumptions. Section 2 of this document discusses the overall description of the system, the target users, and the system's requirements. Section 3 will delve into the detailed functional requirements, data requirements, and design constraints of the system. It will also outline the assumptions made during the design phase. Section 4 will provide additional supporting information such as use cases, interface designs, and other supplementary materials necessary for development and deployment.

**2. Overall Description**

### ****2.1 Product Perspective****

The **Time Management Software (TMS)** will serve as a centralized tool for managing the daily schedules of executives and their secretaries. It will streamline meeting organization, ensure efficient use of time, and improve coordination among executives. The software will integrate with existing systems and be accessible remotely, supporting multiple users simultaneously. TMS is designed to be easy-to-use, even for novice computer users, while offering powerful features for advanced time management.

### ****2.2 Product Features****

* **Appointment Registration**
  + Allows executives to register meeting details such as attendees, location, time, duration, and purpose.
* **Automatic Meeting Scheduling**
  + Finds common open time slots for multiple executives and schedules meetings accordingly.
* **Leave and Task Management**
  + Enables executives to mark leave periods and list important tasks for the day.
* **Daily Engagement Email**
  + Sends daily appointment summaries and task lists to each executive via email.
* **Conflict Resolution and Rescheduling**
  + In case of scheduling conflicts, suggests alternatives and helps rearrange appointments.
* **Time Tracking and Reporting**
  + Provides reports on time spent in meetings, project-related activities, and overall meeting statistics.
* **User Interface**
  + Simple and intuitive graphical interface, designed to be accessible to non-technical users.
* **Remote Access**
  + Supports multiple concurrent users, allowing secretaries and executives to access the system from different locations.

### ****2.3 User Classes and Characteristics****

* **Executives**
  + Primarily responsible for scheduling and attending meetings.
  + May have limited technical knowledge, so the system is designed for ease of use.
* **Secretaries**
  + Responsible for managing the schedules of multiple executives, handling conflicts, and rescheduling appointments.
  + May have more experience with scheduling tools and system features.
* **System Administrators**
  + Responsible for configuring the system, managing user access, and ensuring system security.
  + Requires technical knowledge to maintain and troubleshoot the system.

### ****2.4 Operating Environment****

* **Software**:
  + TMS will be a desktop-based application, compatible with major operating systems (Windows, macOS).
  + The system should also support web-based access for remote users.
* **Hardware**:
  + Desktops or laptops for executives and secretaries.
  + Server infrastructure to handle centralized data storage and user requests.
* **Network**:
  + Requires an internet connection for remote access, email notifications, and real-time updates.

### ****2.5 Design and Implementation Constraints****

* **Ease of Use**
  + The software must be simple and intuitive to accommodate novice users.
* **Scalability**
  + The system must be able to scale as the company grows, accommodating an increasing number of users and requests.
* **Data Security**
  + Strong security measures must be in place to protect sensitive scheduling information.
* **Real-Time Updates**
  + The system must update all calendars and schedules in real time when changes are made by any user.
* **Integration**
  + The software should integrate with email systems and other business tools used by the company.

### ****2.6 Assumptions and Dependencies****

* **Assumptions**
  + Executives and secretaries have access to desktop computers or laptops with internet connectivity.
  + Executives are familiar with basic scheduling and calendar management.
  + The company will provide necessary resources, including server infrastructure and network connectivity.
* **Dependencies**
  + Email systems must be configured to send and receive notifications.
  + The system depends on reliable internet connectivity for remote access and real-time synchronization.
  + Integration with existing software tools (e.g., project management tools, email) is assumed to be available and supported.

### 3. Functional Requirements

* 1. **User Registration and Authentication**
  + The system should allow executives and secretaries to register and log in with secure credentials.
  + Each user should have a unique account with role-based access (e.g., executives and secretaries).

**3.2 Schedule Management**

* + Executives should be able to create, edit, and delete their daily appointments and meetings.
  + The system should store details of each appointment, such as:
    - Invited persons
    - Meeting venue
    - Time and duration
    - Purpose of the meeting (e.g., project discussion)

**3.3 Automatic Meeting Slot Detection**

* + The system should automatically check for overlapping appointments and suggest common available slots for multiple executives.
  + If a suitable slot is not available, the system should notify the secretary to adjust the schedules.
  1. **Email Notifications**
  + Executives should receive automatic email notifications for their meetings and appointments.
  + The system should send daily reminders with the executive’s schedule for the day.
  + Executives should be notified via email if a meeting is successfully scheduled or if any changes are made to their schedule.

**3.5 Leave and Engagement Management**

* + Executives should be able to mark periods of time for which they will be on leave.
  + The system should allow executives to post important tasks or jobs for a specific day and integrate these into their daily list of engagements.

**3.6 Conflict Resolution**

* + The system should offer suggestions to rearrange appointments for executives if no common slot is found, with the help of the secretary.

**3.7Graphical User Interface (GUI)**

* + The system should provide an intuitive and easy-to-use graphical interface to allow executives to view their daily schedules.
  + The interface should display schedules in a clear, time-blocked view (e.g., calendar or agenda format).

**3.8 Remote Access**

* + The system should be able to handle simultaneous remote requests, allowing executives and secretaries to access their schedules from their desktops.

**3.9 Statistics and Reporting**

* + The system should generate reports on various statistics such as:
    - Time spent by each executive on meetings (e.g., man-hours).
    - The total duration of meetings for specific projects.
    - The fraction of time spent in meetings over a specific period.

**3.10 System Integration**

* + The system should integrate with email services to send notifications and updates.
  + The system should provide data export options for reporting purposes (e.g., exporting meeting logs, statistics).

### 4. Non -Functional Requirements

* 1. **Usability**
  + The system should be easy to use, especially for users with limited computer proficiency.
  + The graphical interface should be intuitive, with simple navigation and minimal training required for users to operate effectively.

**4.2 Reliability**

* + The system should ensure high reliability and minimize downtime.
  + Email notifications should be reliably sent and received without delays.

**4.3 Performance**

* + The system should support multiple simultaneous requests from different executives and secretaries.
  + The scheduling feature should be able to identify conflicts and find available slots quickly, even when dealing with multiple executives.
  1. **Security**
  + The system should ensure secure authentication and authorization mechanisms for different users (executives and secretaries).
  + Data security should be a priority, particularly when dealing with confidential meeting details and personal schedules.
  + Sensitive information, such as meeting details and executive schedules, should be encrypted both in transit and at rest.

**4.5 Scalability**

* + The system should be scalable to accommodate future growth, including adding more executives, secretaries, and meeting data.
  + It should handle increasing amounts of data (such as appointments and meeting statistics) as the company grows.

**4.6 Compatibility**

* + The software should be compatible with the desktop systems of executives and secretaries.
  + The system should be compatible with common email services for sending notifications (e.g., Gmail, Outlook).

**4.7 Availability**

* + The system should be available and accessible to users during working hours, with minimal maintenance downtime.
  + Backup mechanisms should be in place to restore the system in case of failures.

**4.8 Maintainability**

* + The system should be designed for ease of maintenance, allowing for quick updates and bug fixes.
  + Clear documentation and logs should be provided to facilitate the maintenance process.

**4.9 Supportability**

* + The system should include support tools or help documentation to assist users who face issues or have questions about using the software.
  + Technical support should be available in case of system failure or issues with functionalities.

**4.10 Portability**

* + The system should be able to run on different operating systems commonly used by executives and secretaries (Windows, macOS, etc.).
  + It should also have the ability to scale across different devices if needed (laptops, tablets, etc.).

**Data Dictionary**

#### ****1. User Registration and Authentication****

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Data Type** | **Description** |
| UserID | Integer | Unique identifier for the user (executive or secretary). |
| Username | String | User's login name |
| Password | String | User's password, stored securely. |
| Role | String | Role of the user (Executive/Secretary). |
| Email | String | User’s email address for notifications. |

#### ****2. Schedule Management****

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Data Type** | **Description** |
| AppointmentID | Integer | Unique identifier for the appointment. |
| ExecutiveID | Integer | Unique identifier for the executive involved. |
| Date | Date | The date the appointment is scheduled for. |
| StartTime | Time | The start time of the appointment. |
| EndTime | Time | The end time of the appointment. |
| Venue | String | The location of the meeting. |
| Purpose | String | The reason for the meeting (e.g., project discussion). |

#### ****3. Meeting Slot Detection****

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Data Type** | **Description** |
| SlotID | Integer | Unique identifier for the time slot. |
| ExecutiveIDs | List of Integer | |  | | --- | | List of executives whose schedules are being checked. |  |  | | --- | |  | |
| StartTime | Time | Start time of the common available slot. |
| EndTime | Time | End time of the common available slot. |

#### ****4. Email Notifications****

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Data Type** | **Description** |
| NotificationID | Integer | Unique identifier for the notification. |
| UserID | Integer | The user to whom the email notification is sent. |
| Subject | String | Subject line of the email notification. |
| Body | String | Subject line of the email notification. |
| Timestamp | DateTime | The date and time when the email was sent. |

|  |  |  |
| --- | --- | --- |
|  |  |  |

#### ****5.**** Engagement & Leave Management

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Data Type** | **Description** |
| EngagementID | Integer | Unique identifier for the engagement/task. |
| ExecutiveID | Integer | Unique identifier for the executive. |
| Date | Date | The date the engagement is scheduled for. |
| StartTime | Time | The start time of the engagement. |
| EndTime | Time | The end time of the engagement. |
| TaskDescription | String | Description of the task or job to be done. |
| Priority | String | Priority level of the task (e.g., High, Medium, Low). |
| LeaveID | Integer | Unique identifier for the leave period (if applicable). |
| Type | String | Type of leave or engagement (e.g., sick leave, task, personal work). |

#### ****6. Statistics & Reporting****

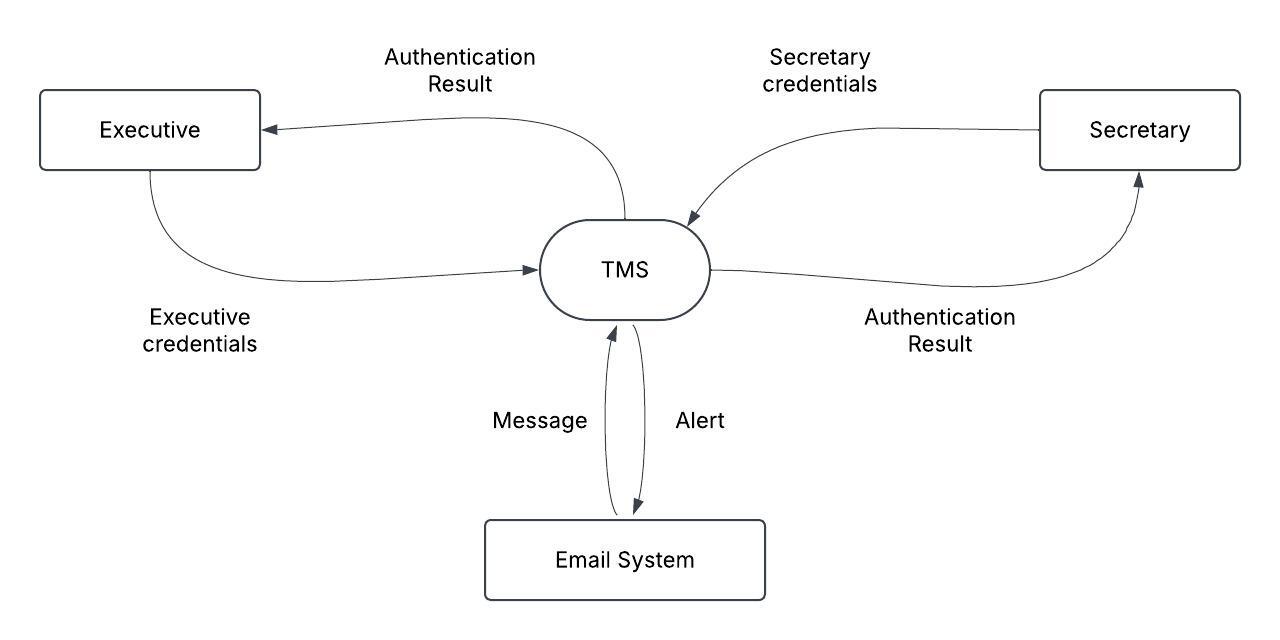
|  |  |  |
| --- | --- | --- |
| **Field Name** | **Data Type** | **Description** |
| |  | | --- | | ReportID | | |  |  | | --- | --- | |  | Integer | | |  | | --- | | Unique identifier for the report. | |
| |  | | --- | | ExecutiveID | | |  |  |  | | --- | --- | --- | |  | Integer | . | | |  | | --- | | The executive for whom the report is generated. | |
| |  | | --- | | TotalMeetings | | |  |  |  | | --- | --- | --- | |  | Integer | . | | |  | | --- | | Total number of meetings attended by the executive. | |
| |  | | --- | | TotalManHours | | |  |  | | --- | --- | | Float | . | | |  | | --- | | Total man-hours spent on meetings by the executive. | |
| |  | | --- | | DateRange | | |  |  | | --- | --- | |  | String | | |  |  | | --- | --- | | The time range for the report (e.g., weekly, monthly). |  | |
| ProjectID | Integer | The ID of project related to the meetings |
| TotalProjectMeetings | Integer | Total meeting related to specific project |
| TotalProjectManHours | Float | Total man-hours spent on meeting for specific project |
| TimeSpentInMeeting | Float | Time spent in a meeting |

#### ****7. System Integration****

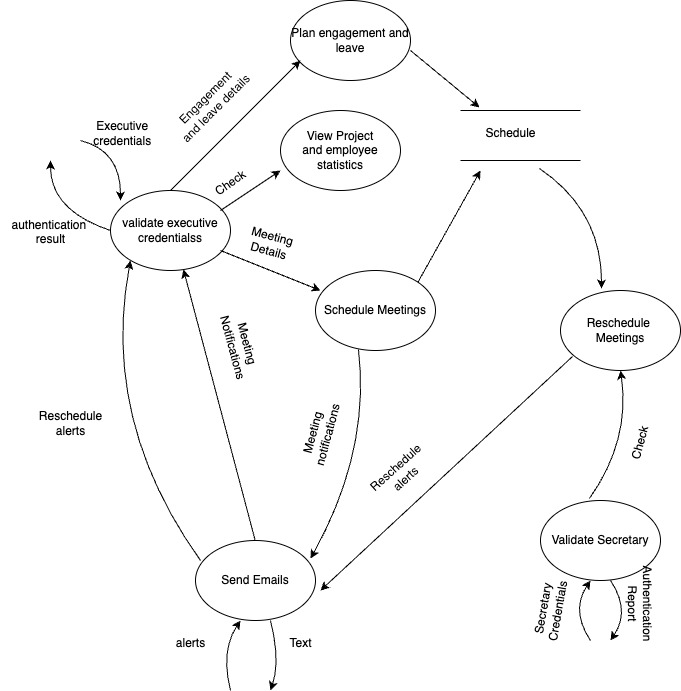
|  |  |  |
| --- | --- | --- |
| **Field Name** | **Data Type** | **Description** |
| |  |  |  | | --- | --- | --- | | EmailID |  |  | | |  |  |  | | --- | --- | --- | |  | Integer |  | | |  |  |  | | --- | --- | --- | |  |  | Unique identifier for email-related integration. | |
| |  |  |  | | --- | --- | --- | | ServiceName |  |  | | |  |  |  | | --- | --- | --- | |  | String |  | | |  |  |  | | --- | --- | --- | |  |  | The name of the email service (e.g., Gmail, Outlook). | |
| |  |  |  | | --- | --- | --- | | IntegrationStatus |  |  | | |  |  |  | | --- | --- | --- | |  | String |  | | |  |  |  | | --- | --- | --- | |  |  | Status of the integration (e.g., active, inactive). | |
| |  |  |  | | --- | --- | --- | | APIKey |  |  | | |  |  |  | | --- | --- | --- | |  | String |  | | |  |  |  | | --- | --- | --- | |  |  | The API key used for email integration (if applicable). | |

**Data Flow Diagram**

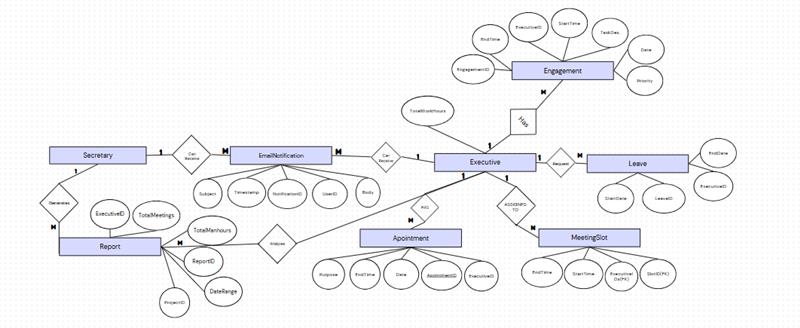
**Level 0 DFD for Time Management Software System**



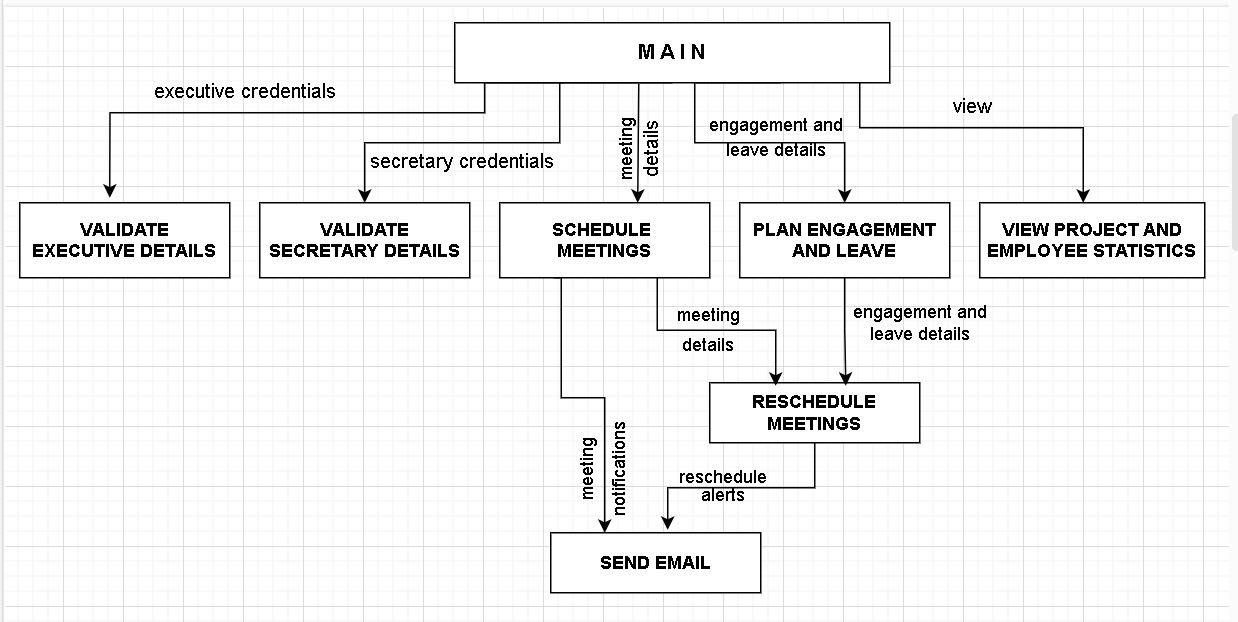
**Level 1 DFD for Time Management Software System**



**ER Diagram**

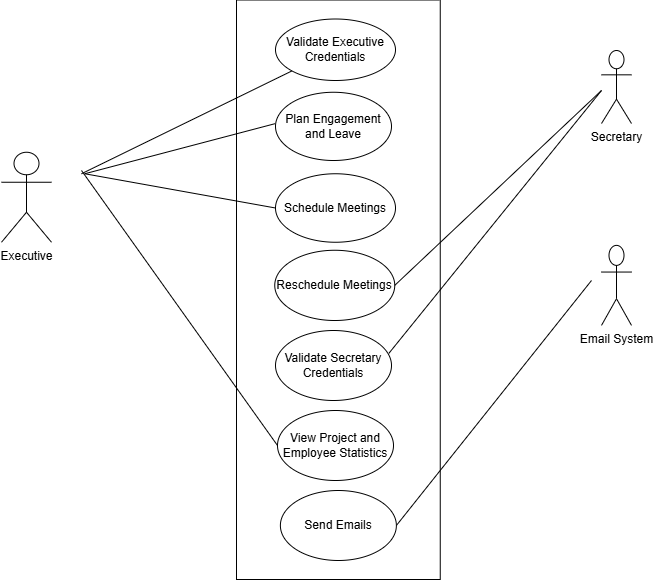
****

**Structure Chart**

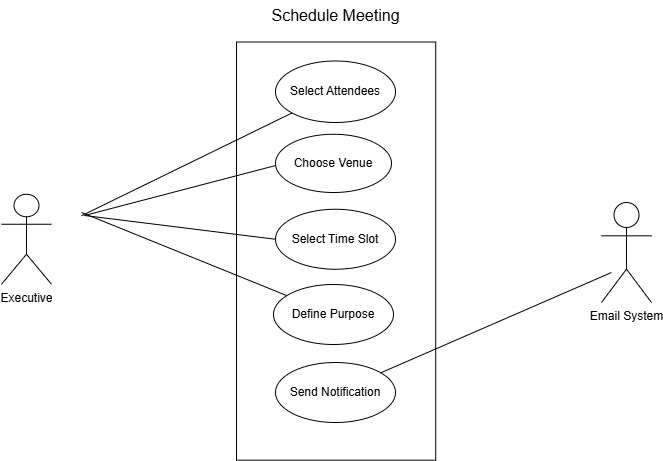
****

**Use Case Diagram**

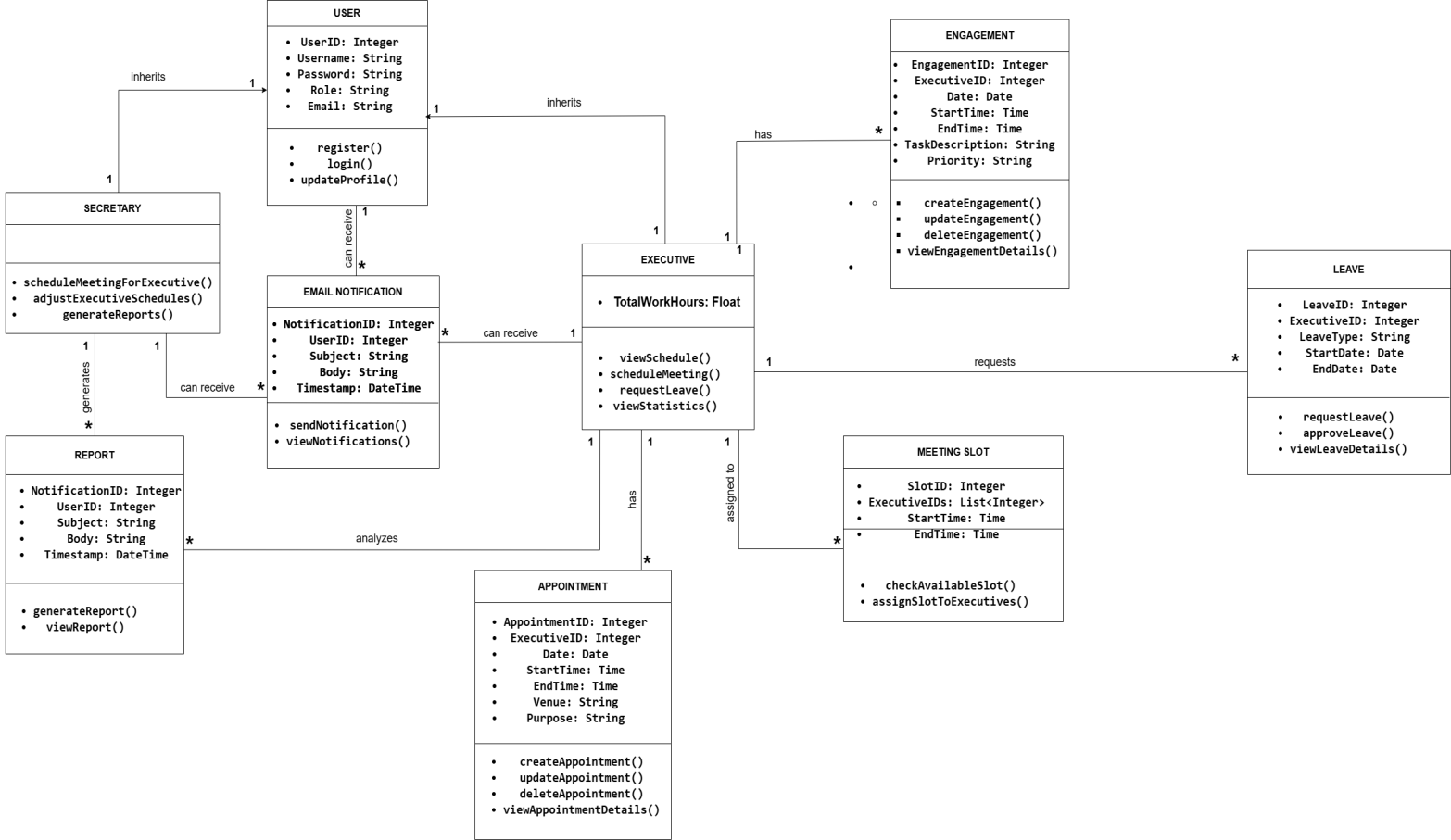
**Use Case Diagram for Time Management Software System**

****

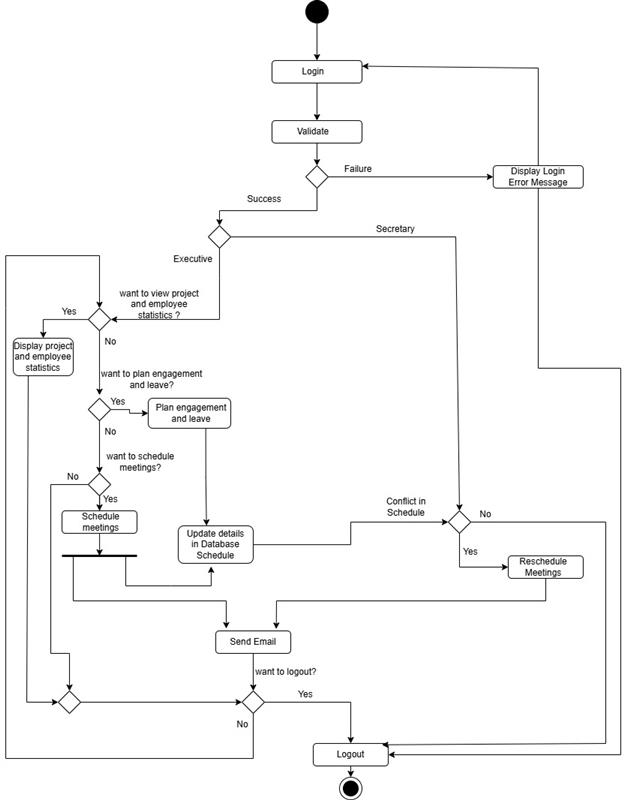
**Use Case Diagram to Schedule Meeting**

****

**Class Diagram**

****

**Activity Diagram**

****