**Software Requirement Specifications**

**Report**

Employee Management System (Software House)

|  |  |
| --- | --- |
| **Project Code** | Not Applicable |
| **Supervisor** | Ms. Sania Urooj |
| **Project Team** | Abdul Rehman Nazeer-K22-4078 |
| **University** | FAST NUCES Karachi Main Campus. |

**Document History**

| **Version** | **Name of Person** | **Date** | **Description of change** |
| --- | --- | --- | --- |
| 1.0 | Abdul Rehman Nazeer | 12-3-2024 | Initial Document Created |
|  |  |  |  |
|  |  |  |  |

**Document Sign-Off**

| **Version** | **Sign-off Authority** | **Sign-off Date** |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |

**Table of Contents**

1. **Introduction**
   * 1.1. Purpose of Document
   * 1.2. Intended Audience
   * 1.3. Abbreviations
   * 1.4. Document Convention
2. **Overall System Description**
   * 2.1. Project Background
   * 2.2. Project Scope
   * 2.3. Not In Scope
   * 2.4. Project Objectives
   * 2.5. Stakeholders
   * 2.6. Operating Environment
   * 2.7. System Constraints
   * 2.8. Assumptions & Dependencies
3. **External Interface Requirements**
   * 3.1. Hardware Interfaces
   * 3.2. Software Interfaces
   * 3.3. Communications Interfaces
4. **Functional Requirements**
   * 4.1. Functional Hierarchy
   * 4.2. Use Cases
     + 4.2.1. Title of use case
5. **Non-functional Requirements**
   * 5.1. Performance Requirements
   * 5.2. Safety Requirements
   * 5.3. Security Requirements
   * 5.4. User Documentation
6. **References**
7. **Appendices**

**1. Introduction**

**1.1. Purpose of Document**

The purpose of this document is to outline the requirements and specifications for the development of an Employee Management System (Software House). It serves as a comprehensive guide for the project team, stakeholders, and developers involved in the creation of the system.

**1.2. Intended Audience**

* Development team members responsible for implementing the Employee Management System (Software House).
* Project stakeholders such as software house owners, managers, and staff.
* Project supervisor involved in project review and approval.

**1.3 Abbreviations**

SRS: Software Requirements Specification GUI: Graphical User Interface HTML: Hyper-Text Markup Language CSS: Cascading Style Sheets JS: JavaScript Django: Django Web Framework

**1.4 Document Convention**

* Font name: Segoe UI
* Headings font size: 12pt (Bold and aligned to the left)
* Subheadings Font Size: 12pt (Bold and aligned to the left)
* Body Text Font Size: 12pt (Regular font, aligned to the left)

**2. Overall System Description**

**2.1. Project Background**

The Employee Management System (Software House) project aims to streamline operations within a software house using Django Python, HTML, CSS, and JavaScript.

**2.2. Project Scope**

The scope includes the development of a comprehensive software solution for managing employee information, tasks, and projects using Django Python, HTML, CSS, and JavaScript.

**2.3. Not In Scope**

Integration with external systems and complex user authentication mechanisms beyond basic login functionality are not within the scope of this project.

**2.4. Project Objectives**

Objectives include addressing operational inefficiencies, providing a comprehensive Employee Management System (Software House) using Django Python, HTML, CSS, and JavaScript, ensuring timely implementation and deployment, and collaborating closely with stakeholders.

**2.5. Stakeholders**

Stakeholders include software house owners, managers, staff, software developers, project managers, quality assurance/testers, and trainers.

**2.6. Operating Environment**

The system will be developed using Django Web Framework, Python programming language, HTML, CSS, and JavaScript.

**2.7. System Constraints**

Constraints include dependencies on Django, Python, HTML, CSS, and JavaScript, time constraints, hardware compatibility with standard desktop or laptop computers, and legal constraints regarding user data protection.

**2.8. Assumptions & Dependencies**

Assumptions include sufficient resources and stakeholder support, while dependencies involve hardware specifications and industry standards.

**3. External Interface Requirements**

**3.1. Hardware Interfaces:**

The Employee Management System does not have direct hardware interfaces. However, it is compatible with standard desktop or laptop computers running Windows, macOS, or Linux operating systems and it can also be used in a mobile-phone browser.

**3.2. Software Interfaces:**

The Employee Management System interfaces with the following software components:

* Django: Python-based web framework used for backend development.
* HTML/CSS/JavaScript: Frontend technologies used for user interface design and interaction.

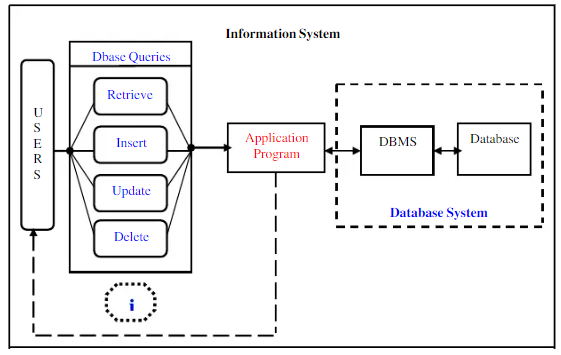
**3.3. Communications Interfaces:**

The Employee Management System utilizes standard HTTP/HTTPS protocols for communication between the client (web browser) and the server (Django application).

**4. Functional Requirements**

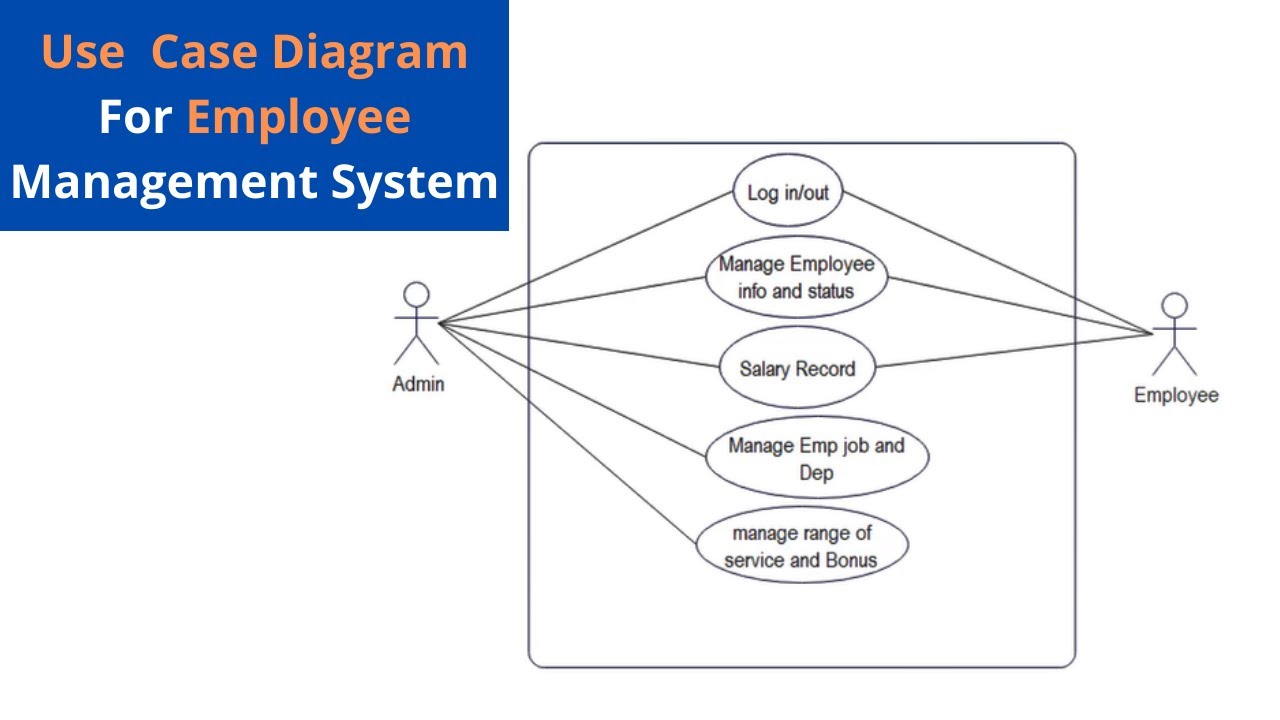
**4.1. Functional Hierarchy**

Below is an example of how the system would be working from the user side of things also I would like to mention that we will be using Django for web framework and not any database.

****

**4.2. Use Cases**

Below is an example of how the things will be working on the whole system i.e. on the side of both User and Admin.



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<Use case Id: name>** | | | | |
| **Use case Id:** | | Use case Id: SH-EMS-001 | | |
| **Actors:**  Administrator, Manager, Employee | | | | |
| **Feature:** Employee Information Management | | | | |
| **Pre-condition:** | | The user must be authenticated and authorized with appropriate permissions to access employee information. | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | Administrator logs into the system. | | | System verifies credentials and grants access. |
| **2.** | Administrator selects "Manage Employees" from the dashboard. | | | System navigates to the employee management module. |
| **3.** | Administrator chooses to add a new employee. | | | System prompts for employee details. |
| **4.** | Administrator enters employee details (name, position, department, contact info, etc.). | | | System validates the information and saves the new employee record. |
| **5.** | Administrator confirms the addition of the new employee. | | | System displays a confirmation message. |
| **6.** | Administrator logs out of the system. | | | System logs out the administrator and returns to the login screen. |
| **7.** | Manager logs into the system. | | | System verifies credentials and grants access. |
| **8.** | Manager selects "View Employees" from the dashboard. | | | System displays a list of employees. |
| **9.** | Manager filters employees by department. | | | System shows employees within the selected department. |
| **10.** | Manager selects an employee to view details. | | | System displays the selected employee's information. |
| **11.** | Manager requests to update an employee's information. | | | System prompts for necessary changes. |
| **12.** | Manager submits the changes. | | | System updates the employee details and displays a confirmation message. |
|  |  | | |  |
| **Alternate Scenarios:** | | | | |
| **4a**: If the entered employee details are invalid or incomplete, the system prompts the administrator to correct them.  **9a**: If there are no employees in the selected department, the system displays a message indicating no employees found.  **10a**: If the selected employee does not exist or cannot be accessed, the system displays an error message. | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **1.** | Administrators, managers, and employees can log in and access the employee management features. | | | |
| **2.** | New employees can be added, and existing employee information can be viewed and updated. | | | |
| **3.** | Employee information is securely stored and can be managed by authorized users. | | | |
| **Use Case Cross referenced** | | | Not Applicable. | |

**5. Non-functional Requirements**

**5.1. Performance Requirements**

* **System Responsiveness:** The system would respond to user interactions within 5 seconds under normal load conditions.
* **Scalability:** The system would be able to handle a minimum of 500 concurrent users without significant performance degradation.
* **Data Processing Speed:** The system would process and retrieve employee data for reports within 5 to 10 seconds.
* **System Availability:** The system would have an uptime of at least 99.9% to ensure continuous availability during business hours.

**5.2. Safety Requirements**

* **Data Backup:** The system automatically backs up employee data daily to ensure data safety and integrity.
* **Data Recovery:** In the event of a system failure, the system would be able to recover data from the latest backup within 24 hours.(NOT APPLICABLE)

**5.3. Security Requirements**

* **Authentication:** Users would be required to authenticate themselves using a username and password to access the system.
* **Access Control:** Role-based access control (RBAC) would be implemented to restrict access to sensitive employee information based on user roles.
* **Data Encryption:** Employee data would be encrypted during transmission over the network and when stored in the database to prevent unauthorized access.

**5.4. User Documentation**

* **User Manuals:** Comprehensive user manuals would be provided to guide users on how to navigate and use different features of the system.
* **Online Help:** Context-sensitive online help would be available within the system to provide users with assistance while using specific functionalities.
* **Leave Management:**
  + Include details on how employees can request leave.
  + Provide instructions for managers on how to review and respond to leave requests.
* **Salary Adjustment:**
  + Offer guidelines for HR on adjusting salaries based on leave records.

**6. References**

UML Diagram Source:

<https://www.academia.edu/75155355/Employee_Management_System>

<https://www.youtube.com/watch?v=lRFMNAiqtkY>

**7. Appendices**

**Feedback and Iterations:** Will gather feedback from users during the implementation phase and iterate on the leave management and salary adjustment features accordingly.

**Test Case:**

**Test Case Description: Verify that an employee can successfully submit a leave request through the Employee Management System.**

**Preconditions:**

**The Employee Management System is accessible.**

**The employee is logged into their account with appropriate permissions.**