Noakhali Science & Technology University
Department of Computer Science & Telecommunication Engineering

Lab Report On:

Introduction and Requirement Analysis

Course Code: CSTE 3210

Submitted by:

Mohammad Borhan Uddin
ID: ASH2101008M

Mohammad Billal Hossain
ID: MUH2101028M

Submitted to:

Dr. Nazia Majadi
Associate Professor
Department of Computer Science & Telecommunication Engineering

Date : March 12, 2025

Contents

1 Introduction

The *Problem Fixer* system is a proposed complaint management platform designed to improve the efficiency of reporting and resolving issues within a university. The system will facilitate structured and transparent handling of complaints, ensuring that problems are addressed in a timely manner.

1.1 Project Scope

The *Problem Fixer* system will provide a structured approach for teachers to report issues within the university. These complaints may include malfunctioning office equipment, electrical issues, or sanitation concerns. Complaints will be categorized to ensure efficient handling by the administration. The admin will have the capability to assign employees to specific complaint categories and track resolutions. A notification system will be integrated to update users on the status of their complaints, ensuring smooth communication.

1.2 Objectives and Goals

The key objectives and goals of the *Problem Fixer* system include:

- Developing an **automated and structured** complaint management system.
- Allowing **teachers to submit complaints** and track their progress.
- Enabling **administrators to assign employees** to complaints for quick resolution.
- Ensuring **timely notifications** through in-system alerts and email updates.
- Providing users with the ability to **mark complaints as resolved** once satisfactorily addressed.
- Improving **efficiency and accountability** in complaint resolution.
- Designing the system with **scalability** to handle increasing users and complaints without performance degradation.
- Ensuring **modularity** in the system architecture, allowing easy updates, feature additions, and maintenance.
- Adhering to **software engineering best practices**, including clean code, version control, and thorough testing for robust system development.

2 Defining Requirements

This section defines the essential components, functionalities, and performance expectations for the *Problem Fixer* system.

2.1 System Requirements

The system will be composed of the following key modules:

- **User Module (Teachers):** Enables users to submit complaints, track their status, and mark them as resolved.
- **Admin Module:** Allows administrators to manage complaint categories, assign employees, resolve complaints, and send notifications.
- **Employee Management Module:** Enables the admin to add employees and assign them to specific complaint categories.
- **Notification System:** Sends real-time notifications and emails to users upon complaint updates.
- **Database Management:** Securely stores all complaint records, user details, and resolution history.

2.2 Functional Requirements

The following functionalities will be supported:

1. **User Features:**
    - Users can **log in** and **submit complaints**.
    - Users can **select a complaint category** (e.g., Computer, Electrical, Cleaning, AC).
    - Users can **track the status** of their complaints.
    - Users can **mark complaints as done** once resolved.
2. **Admin Features:**
    - Admin can **log in** and **manage complaint categories**.
    - Admin can **add departments**.
    - Admin can **add employees** and **assign them to specific complaint categories**.
    - Admin can **view and manage complaints** submitted by users.
    - Admin can **resolve complaints** by providing resolution details.
    - Admin can **assign an employee** to handle a complaint if necessary.
    - Admin can **send notifications and emails** to users when a complaint is resolved.

2.3 Non-Functional Requirements

1. **Performance Requirements:**
   - The system should be **scalable and capable of handling multiple complaints concurrently**.
   - Complaint submission and updates should be processed **in real-time**.
2. **Security Requirements:**
   - Secure authentication mechanisms must be implemented.
   - Only **authorized admins** should be able to manage complaints and assign employees.
3. **Usability Requirements:**
   - The interface should be **intuitive and easy to navigate**.
   - Complaint submission should be **simple and user-friendly**.
4. **Reliability and Maintainability:**
   - The system should ensure **high availability with minimal downtime**.
   - Data should be **regularly backed up** to prevent loss.

3 Requirement Analysis

The requirement analysis for the "Problem Fixer" project identifies key system components and user interactions. It focuses on understanding how admins, employees, and teachers interact with the system, automating processes, and managing complaints from submission to resolution. The following use case diagram and descriptions outline the primary system requirements.

3.1 Scenario-Based Models

3.1.1 Use Case Diagram

The Use Case Diagram for the "Problem Fixer" system models the interactions between the users and the system. The actors in the system include the following.

- **Primary Actor:** User (Teacher) - The individual who submits complaints and marks them as resolved.
- **Secondary Actor:** Admin - The person who manages departments, categories, employees, and resolves complaints.

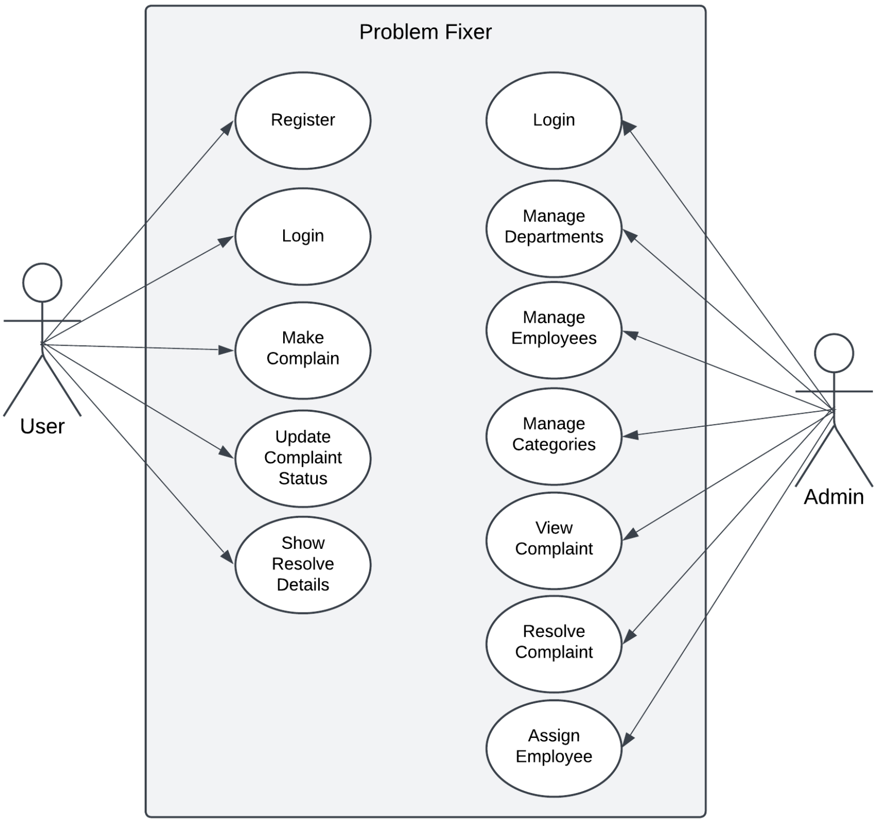

Figure 1: Use Case Diagram

3.1.2 Use Case Descriptions

Use Case 1: Submit a Complaint

- **Actor:** User (Teacher)
- **Description:** The user submits a complaint related to a broken item, service request, or facility issue (e.g., a broken office computer). The user selects a category for the complaint (e.g., Computer, Electrical, Cleaning) and provides a description of the issue.
- **Preconditions:** The user is logged into the system.
- **Postconditions:** The complaint is successfully submitted and recorded in the system.

Use Case 2: Manage Employees, Departments, and Categories

- **Actor:** Admin
- **Description:** The admin manages employees, departments, and categories to ensure smooth operation of the system. The admin assigns employees to specific departments, creates categories for complaints, and ensures that each department has the appropriate resources to handle complaints efficiently.
- **Preconditions:** The admin is logged into the system and has access to management features.
- **Postconditions:** The employees, departments, and categories are successfully managed, and resources are allocated to handle complaints effectively.

Use Case 3: Assign Employees to Complaints

- **Actor:** Admin
- **Description:** The admin assigns employees to complaints based on their expertise in relevant categories (e.g., a computer technician for computer-related issues).
- **Preconditions:** The complaint is in the "due" state, and there is an employee available to handle it.
- **Postconditions:** The employee is successfully assigned to the complaint, and they can begin working on resolving the issue.

3.2 Behavioral Models

Behavioral models help visualize the dynamic interactions within the system for better understanding.

3.2.1 Activity Diagram

The Activity Diagram depicts the flow of activities in two key processes of the system, showing how various actions and decisions occur in a sequential manner.

(a) Activity Diagram for Making a Complain Process by User

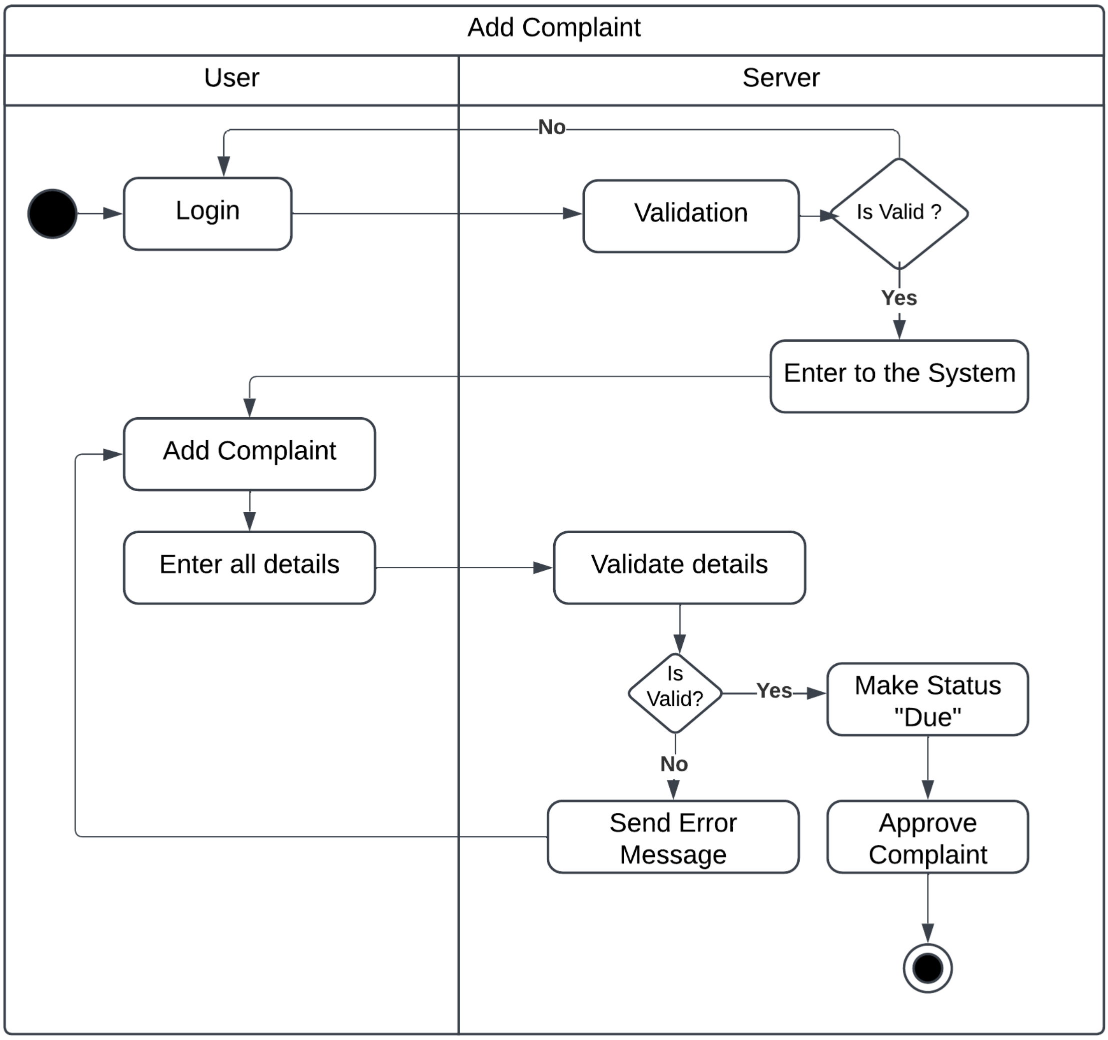

Figure 2: Activity Diagram of Making Complain Process by User

(b) Activity Diagram of the Complaint Management Process by Admin

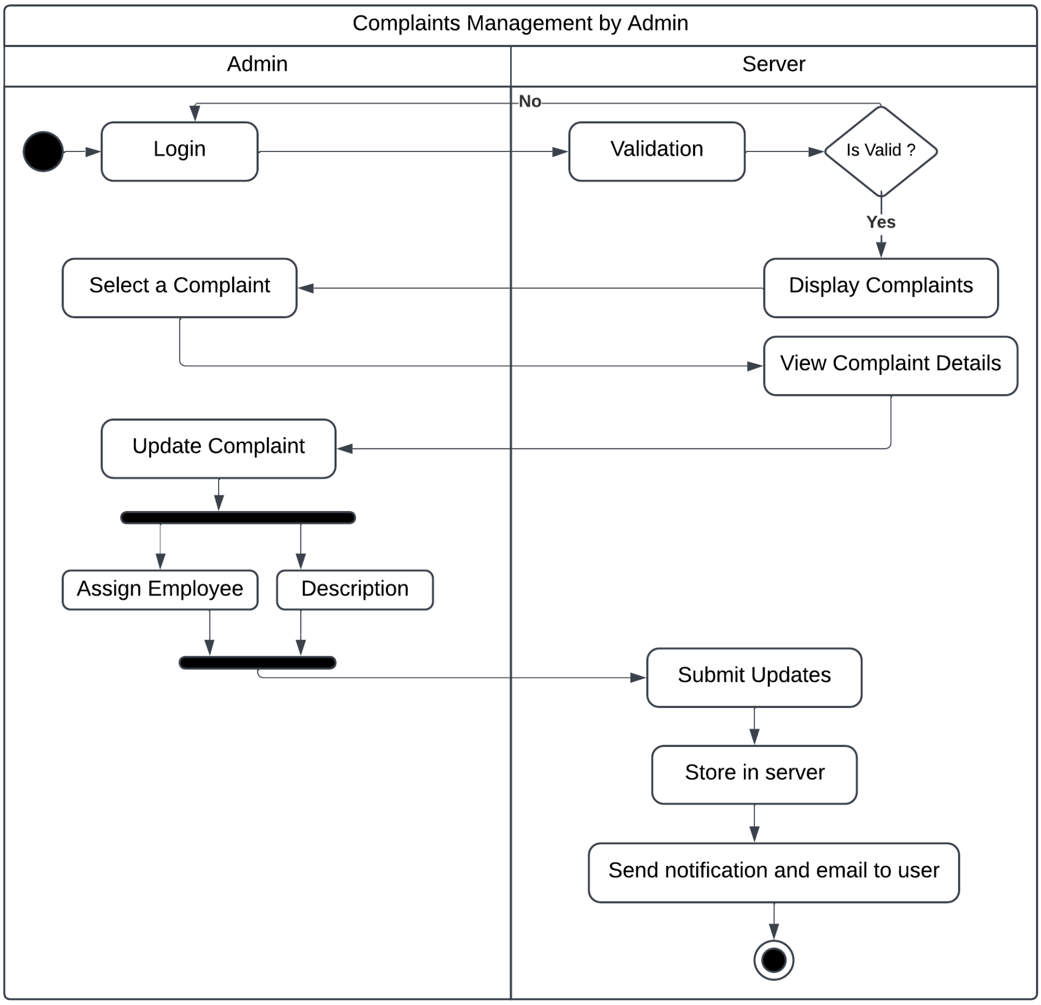

Figure 3: Activity Diagram of Complaint Management Process by Admin

3.2.2 Sequence Diagram

The Sequence Diagram illustrates the interaction between system components for two important processes, showing the order of messages exchanged between the objects in the system.

(a) Sequence Diagram of Registration Process

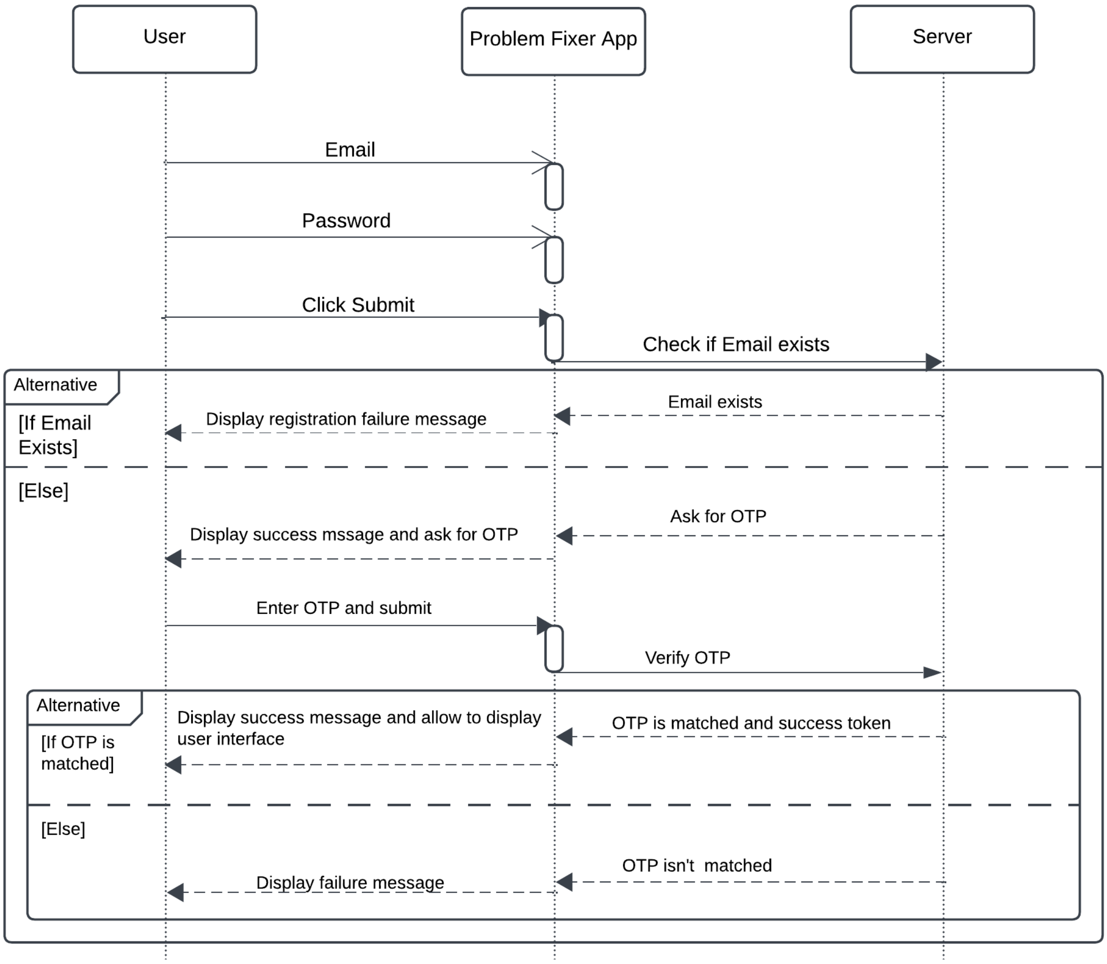

Figure 4: Sequence Diagram of Registration Process

(b) Sequence Diagram of Adding Employee

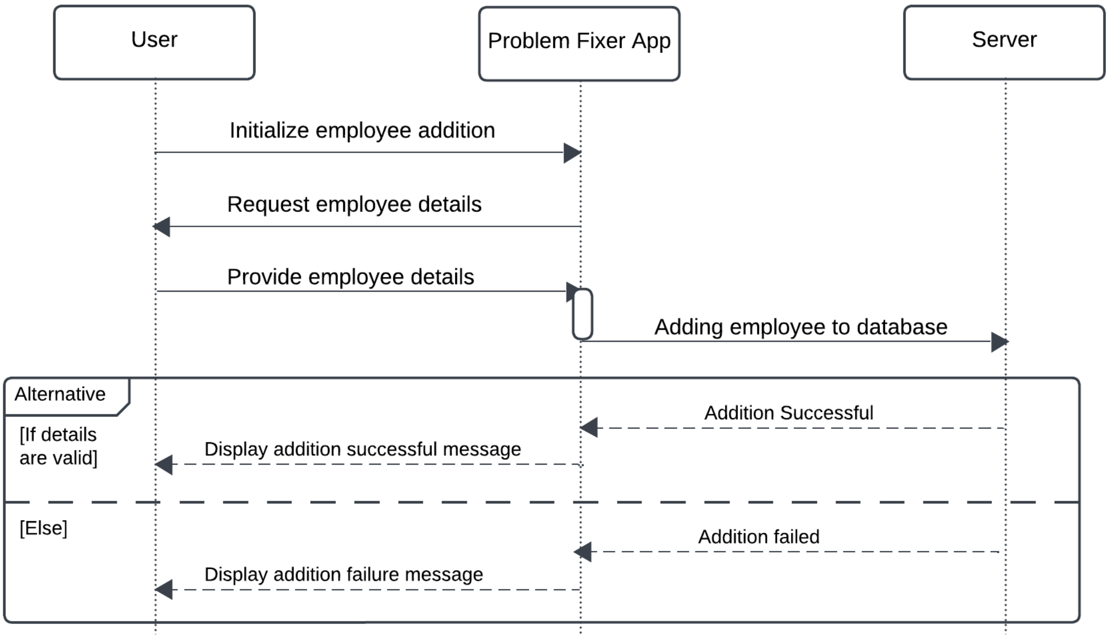

Figure 5: Sequence Diagram of Adding Employee

3.3 Flow Models

Flow models represent how data and control flow through the system, providing a clear picture of data processing and transformation.

3.3.1 Level 0 DFD

The Level 0 Data Flow Diagram (DFD) provides an overview of the system's data flow and high-level processes. It represents the system as a single process, showing interactions with external entities and data inputs/outputs.

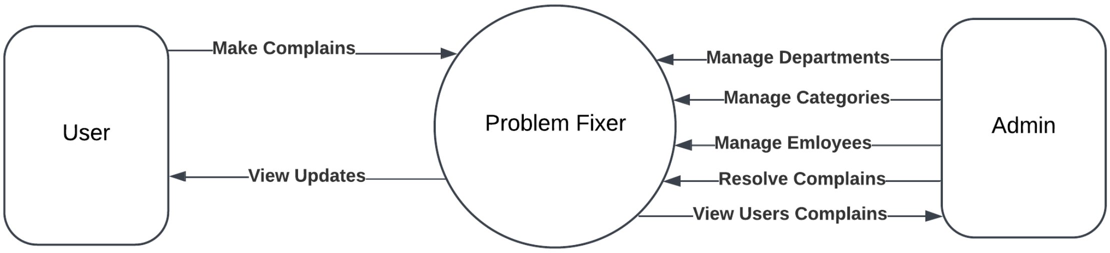

Figure 6: Sequence Diagram of Registration Process

3.3.2 Level 1 DFD

The Level 1 DFD breaks down the major processes from the Level 0 DFD into more detailed components, illustrating how data flows within the system at a deeper level.

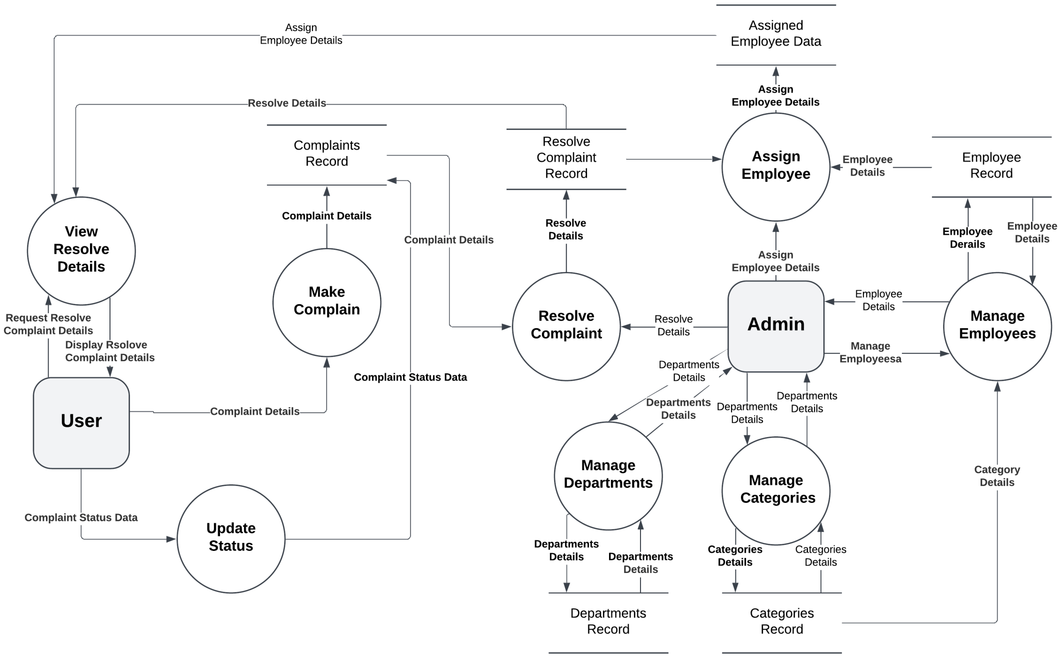

Figure 7: Sequence Diagram of Registration Process