IT 314 Lab 6
Grp 1 (Employee Management System)

Domain analysis models provide a systematic way to analyze the key components, concepts, and relationships within a specific domain. For an employee management system, we can create several domain analysis models to capture different aspects of the domain. Here are a few examples:

User Categories:

- Employee: The lower layer of users, who get a few selected feature access and for only himself/herself.
- Manager: Higher level of users, who get additional information and features for other users too.

User Needs:

- Employee
  - View his/her profile.
  - View his/her attendance history.
  - View his/her performance metrics.
  - View his/her salary status.
  - Apply for leave and see the status of his/her leave application.
  - View department details
- Manager
  - Create an Employee profile.
  - Approve or reject leave application.
  - View attendance of employees.
  - All other features and perks offered to employees.

Use Case Model:

A use case model defines the various actions or tasks that users can perform within the system. For an employee management system, some potential use cases might include:

- Registering a new employee profile.
- Show employee profile page.
- Show salary status page.
- Managing the employee attendance history.
- Applying for a leave (leave application form).

Information Resources:

- Hardware or software support that marks the attendance of employees.
- API that will generate the PDF of the performance report.

Database Requirements:

- NoSQL database that stores and retrieves the data at the highest possible speed.
- Various models for data storage are required.

Security Requirements:

- Password encryption is required for safety.
- Cookies or session generation for creating user specific features.
- Security and login for users.

Identify Boundary, Entity and Control Object.

- Boundary:
    - Boundary for the application would be the interface of mobile application which separates the system and the user.
- Entity:
    - There are several entities associated with the application. This includes the categories of users like employee and Manager.
    - Additionally the entities would include the hardware and software supporting the system.
- Control object:
    - Control objects would be the options and features that generate data or control the database features.

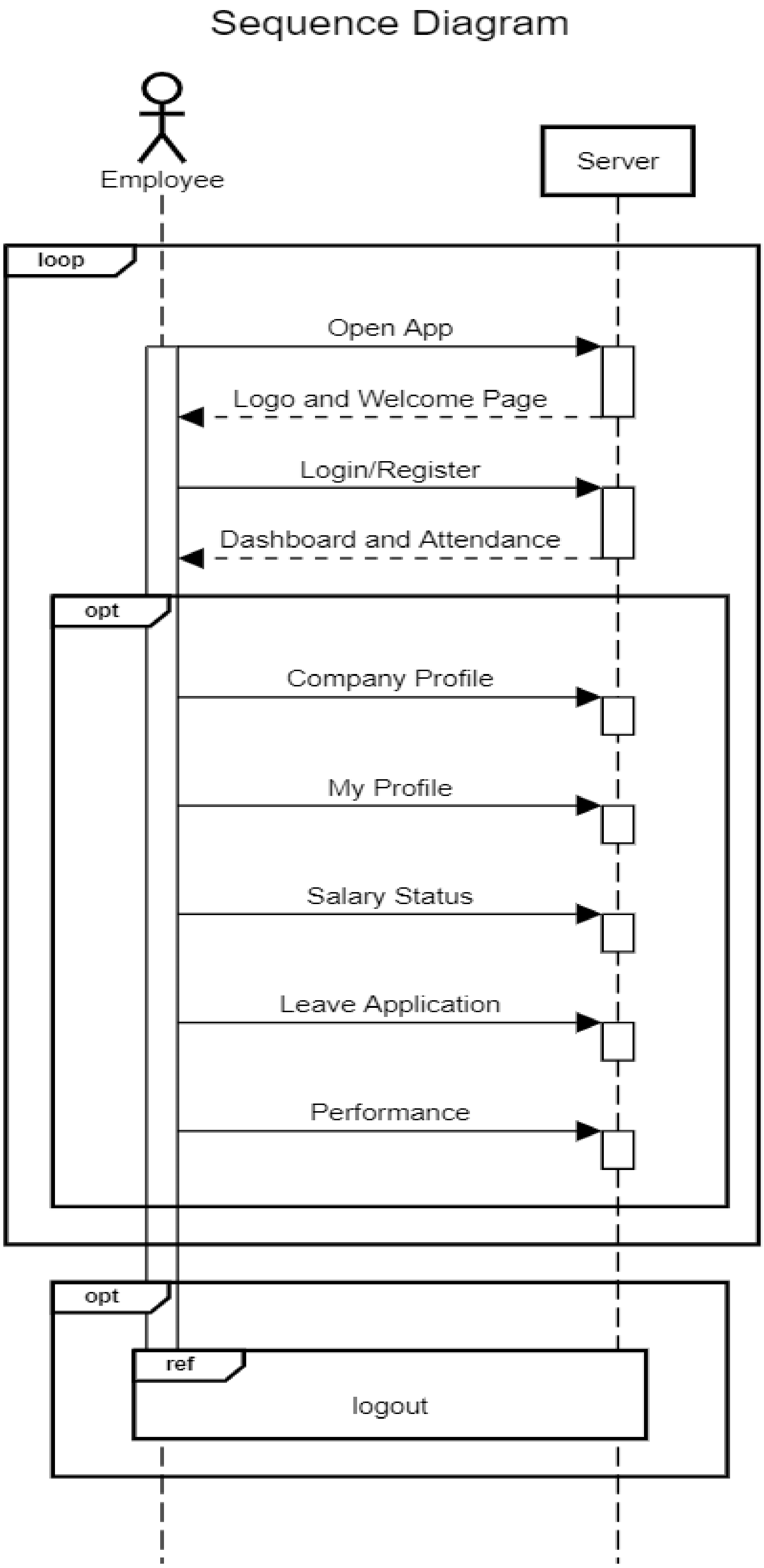
Design Goals:

The design goals for an employee management system may vary depending on the organization's specific needs, but some common design goals include:

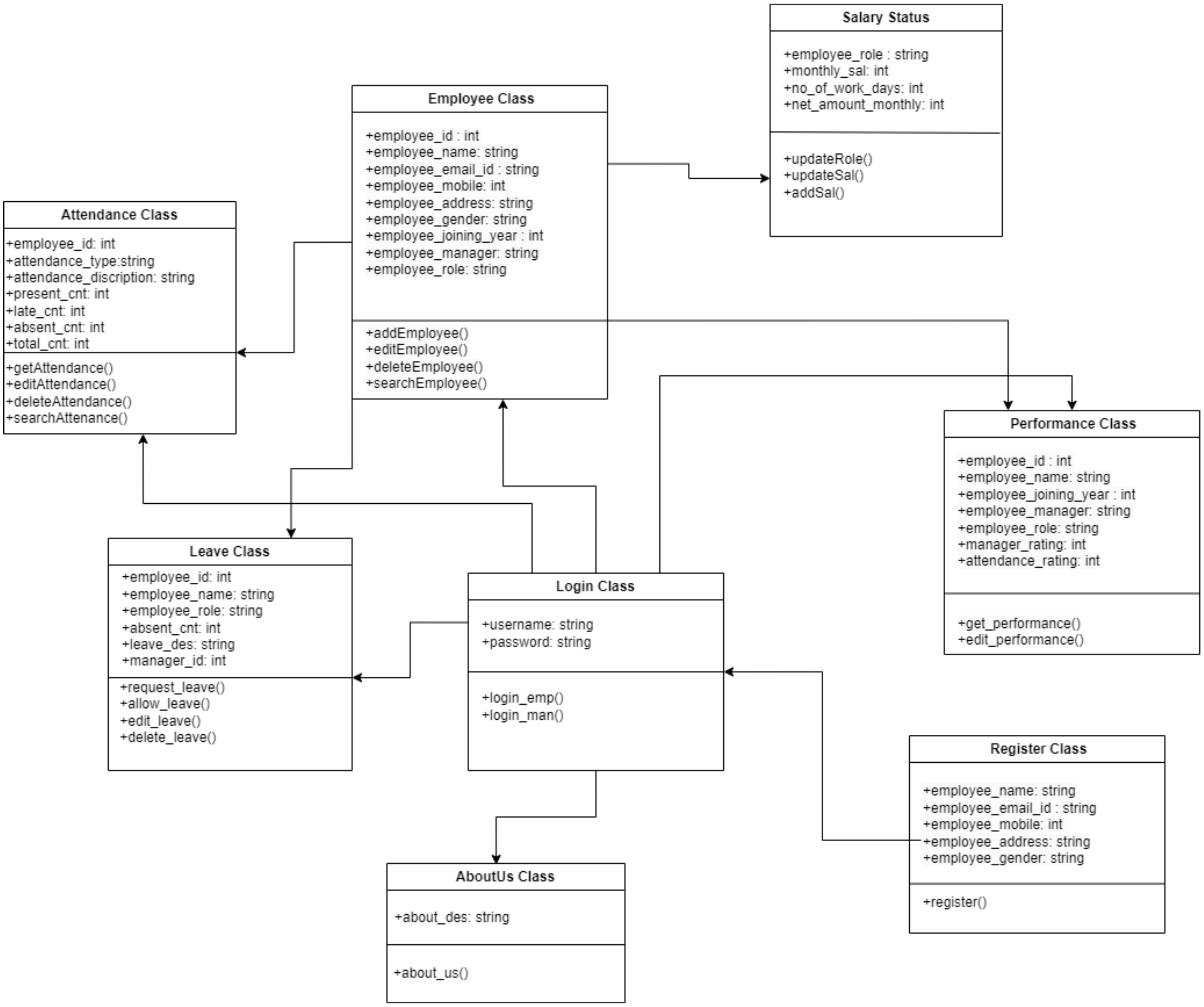
- **Efficient Data Management:** The system should be designed to efficiently manage employee data, including personal details, job roles, attendance history, and performance information.
- **Streamlined Processes:** The system should automate routine processes such as onboarding, leave requests, and payroll, to reduce the burden on HR personnel and improve overall efficiency.

- **Security and Privacy:** The system should have robust security features to protect employee data and prevent unauthorized access, as well as comply with data privacy regulations.
- **User-Friendly Interface:** The system should have a user-friendly interface that is easy to navigate, intuitive, and accessible to all users.
- **Scaling:** Large number of employees might be using the system.
- Reliability
- Backward compatibility.
- Rapid Development.
- Understandability.
- Flexibility of requirements.

Sequence Diagram:



Class Diagram:



High level system design:

This architecture consists of a presentation tier, an application tier, and a data storage tier.

Presentation Tier:

The presentation tier is responsible for handling user interface and user input. This tier includes the following subsystems:

- User Interface: This subsystem includes the web pages and forms that allow users to interact with the website.
- Client-side Logic: This subsystem includes the Dart and other client-side code that handles user input validation and other user interactions.

Application Tier:

The application tier is responsible for handling the business logic and processing user input. This tier includes the following subsystems:

- Controller: This subsystem is responsible for managing the flow of data between the user interface and the data storage tier.
- Business Logic: This subsystem is responsible for implementing the business logic of the website, such as user registration, post creation, and ranking evaluation.
- Security: This subsystem is responsible for implementing security features, such as user authentication and authorization.

Data Storage Tier:

The data storage tier is responsible for storing and managing data. This tier includes the following subsystems:

- Data Access Layer: This subsystem is responsible for accessing and retrieving data from the database.
- Database: This subsystem is responsible for storing all of the website's data, such as user information, forum data, topic data, post data, tag data, and ranking data.
- **Security:** The system should be secure and protect sensitive employee data. Access to the system should be restricted based on user roles and permissions. The system should also be backed up regularly to prevent data loss in case of a system failure.
- **Integration:** The system should be able to integrate with other HR systems and tools, such as applicant tracking systems, performance management software, and benefits administration platforms.

Subsystems:

- Employee Attendance Management System
    - Handles the attendance marking and management system.
- Employee Profile system
    - Everything related to the employee details is handled by this subsystem.
- Leave application management system
    - Everything related to the leave application and its approval.
- Infrastructure and security subsystem
    - Password security and backend security as a whole.