

Software Architecture Document For ScheduFlex

Version 1.0

Prepared by Team BinaryTrio

Faculty of Computing/Sabaragamuwa University of Sri Lanka

15/04/2024

Table of Contents

Table of Contents-----	2
1. Introduction -----	3
1.1 Purpose -----	3
1.2 Scope-----	3
1.3 Definitions, Acronyms, and Abbreviations -----	3
1.4 References -----	4
2. System Analysis-----	4
2.1 Feasibility Study-----	4
2.1.1 Technical Feasibility-----	5
2.1.2 Operational Feasibility -----	6
3. Use Case Diagram -----	7
4. Database Diagram -----	8
5. ER diagram-----	9
6. System Design-----	9
6.1 Architectural Goals and Constraints-----	9
7. Class Diagram -----	11
8. Activity Diagram-----	12
8.1 Process of Admin Registration-----	12
8.2 Process of Admin Login to the System -----	13
8.3 Process of Adding Relief Teacher-----	14
8.4 Process of Teacher Registration -----	15
8.5 Process of Viewing the Updated Timetable of the Available Teacher -----	16
9. Sequence Diagram-----	17
9.1 Teachers Viewing the Tima Table -----	17
9.2 Admin Assigning the Reliefs-----	18
9.3 Admin Login -----	19
9.4 Register a Teacher -----	20
9.5 Register an Admin -----	21

Software Architecture Document

1. Introduction

In this document, we gathered the requirements of the client and started to design the system after identifying the issue. At first, UML diagrams were designed for ease of understanding the system better. After the Activity, Class, and Sequence diagrams were designed to further understanding.

1.1 Purpose

The purpose of this system analysis and design document is to provide a comprehensive overview of the architectural design and structure of the proposed ScheduFlex system for Sri Lankan government schools. This document outlines the system's complete architectural overview. The document, which consists of a number of different diagrams, including class diagrams, use case diagrams, ER diagrams, and activity diagrams, will help developers understand the system when they implement it in the real world. Apart from developers, this document serves as a reference for all stakeholders involved in the software development lifecycle.

1.2 Scope

This document covers the architectural aspects of the ScheduFlex for optimizing schedules in the schools. It serves a high-level view of architecture, components, data flow, and deployment strategy. Multiple different diagrams have been used including use case diagrams, class diagrams, sequence diagrams, ER diagrams for database and package diagrams.

1.3 Definitions, Acronyms, and Abbreviations

Admin:

Personnel responsible for overseeing the operation and management of the ScheduFlex, including school principals, vice-principals, and administrative staff.

Section:

A school divides all grades into manageable chunks. Generally, Sri Lankan government school has 3 main sections namely the Primary Section, Secondary Section, and Advanced Section.

Relief Period:

An absent teacher duty period is assigned when a teacher is released from his or her regular teaching duties.

Period:

A time period allotted to a teacher for teaching.

1.4 References

All diagrams were drawn using Figjam diagramming software.

Figjam, “Free UML Diagram Tool | FigJam”.

<https://www.figma.com/templates/uml-diagram-tool/>

2. System Analysis

System analysis is a structured process of analyzing a system in order to understand its functionality, behavior, and interactions with other systems or users. It involves studying both the user requirements and the technical aspects of a system to propose improvements or solutions. The key objectives of system analysis include Understanding Requirements, problem identification, and feasibility study. challenges. Before designing the system, we identified the functional and non-functional requirements in the system as our client mentioned.

2.1 Feasibility Study

The purpose of a feasibility study is to evaluate all aspects of a project to determine if it is viable, practical, and worth pursuing. The feasibility study for our project aims to assess its practicality and viability from both operational and technical perspectives. This study will help us make informed decisions and ensure the successful implementation of the project.

2.1.1 Technical Feasibility

Technical feasibility refers to the assessment of whether a proposed system can be successfully developed, implemented, and operated using current technology, and resources.

Infrastructure:

Evaluate the availability of servers, hosting platforms, and network infrastructure to support the web application's requirements, including data storage, security, and access control.

Resource Evaluation:

Assessing the availability and capability of resources such as skilled personnel, equipment, and facilities required for the proposed system.

Software and Development Tools:

Determine the availability and compatibility of software and development tools required for building the web application, such as programming languages, frameworks, and databases.

Security Measures:

Consider the technical measures required to ensure data security, user authentication, and protection against vulnerabilities and cyber threats.

Performance and Scalability:

Evaluate the technical feasibility of achieving the desired performance levels and accommodating potential future growth in terms of user load, concurrent usage, and data storage.

2.1.2 Operational Feasibility

Operational feasibility refers to the assessment of whether a proposed system will be operationally viable once it is implemented within government schools. It focuses on evaluating the practicality and effectiveness of the project from an operational standpoint, considering the impact on existing processes, people, and resources.

Compatibility:

Evaluate the compatibility of the web application with various devices, browsers, and operating systems commonly used by school non-academic staff (system users).

Scalability:

Determine if the system can handle increasing volumes of data, ensuring smooth performance and responsiveness.

User Acceptance and adoption:

Assessing the willingness of stakeholders (teachers, non-academic staff, and school management) to adopt and utilize the proposed system effectively.

Training and Support:

Assess the feasibility of providing training and support resources to ensure that users can understand and use the system efficiently.

Operational Risks and Challenges:

Identifying potential operational risks, challenges, or barriers that may hinder the successful deployment and utilization of the project. This includes factors such as regulatory compliance, cultural considerations, and dependencies on external factors.

In this document, we will focus on the diagrams that we designed to visualize the functional and non-functional requirements. We have. There are several UML diagrams we have used,

1. Use case diagram.
2. Database diagram.
3. ER diagram.

3. Use Case Diagram

The relationship between the system and the outside users is depicted in the use case diagram. It provides a quick overview of which user interacts with which system functionality. Use case diagrams were created based on modules. The Intended Use Case diagrams for our system are shown below.

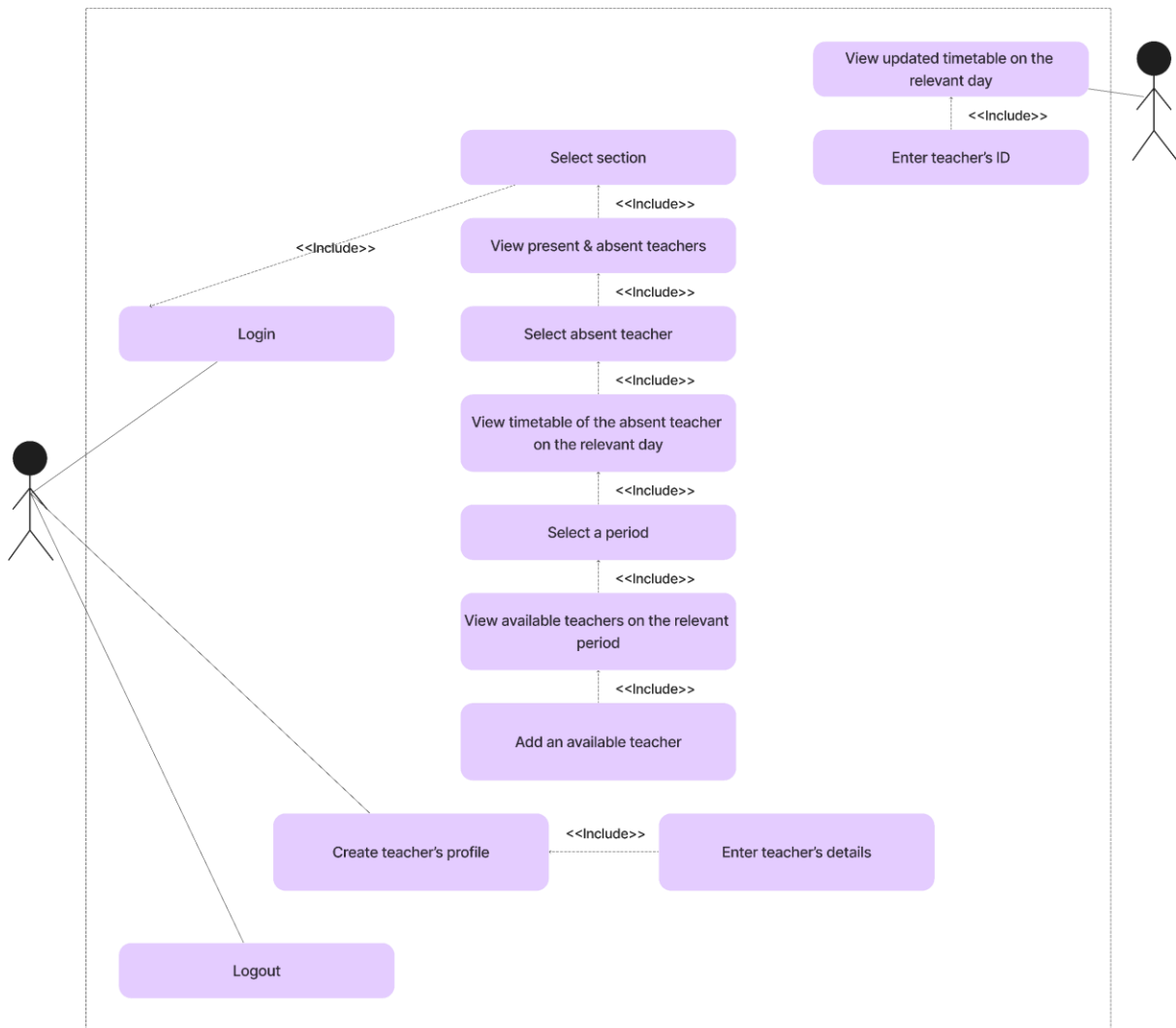


Figure 3.1 illustrate the relationship between the system and the outside users.

4. Database Diagram

Database Diagram is a visual representation of the structure and relationships within a database. It provides a graphical overview of the database schema, including tables, columns, relationships, and constraints. Database diagrams are commonly used during the design phase of a database project to plan and organize how data will be stored, accessed, and manipulated.

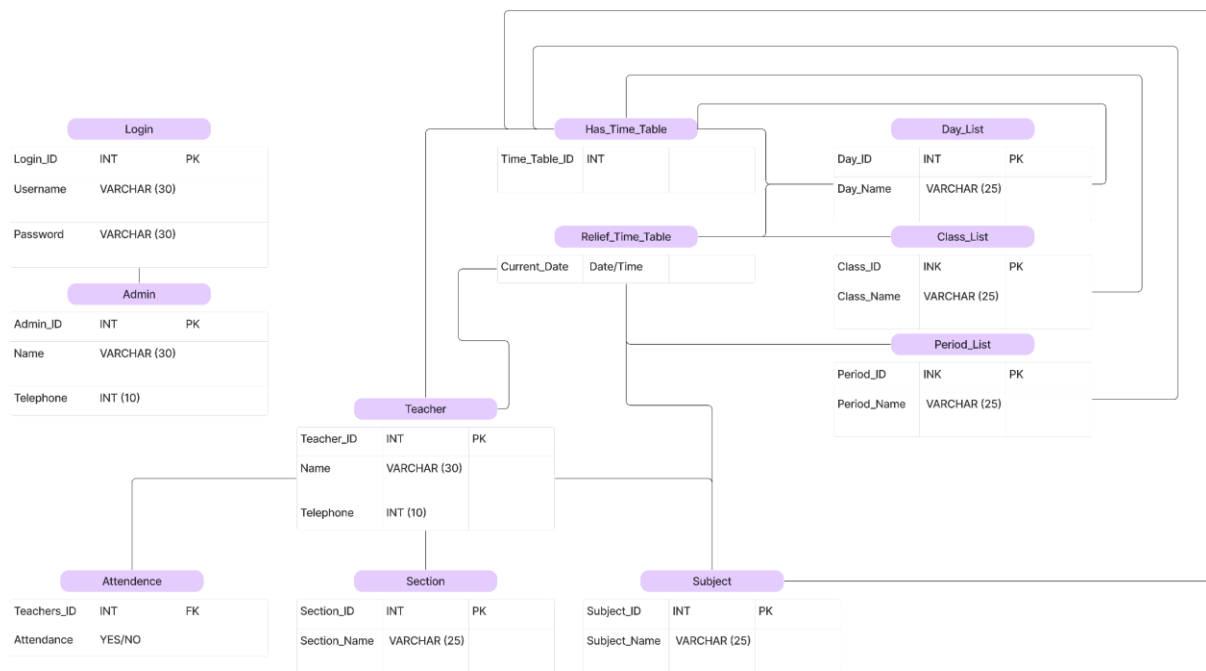


Figure 4.1 illustrates a visual representation of the structure and relationships within a database.

5. ER diagram

Entity-Relationship diagram is a visual representation of the entities (or objects) within a database system and the relationships between those entities. It's a fundamental tool used in database design to model the structure of a database and define how data should be organized.

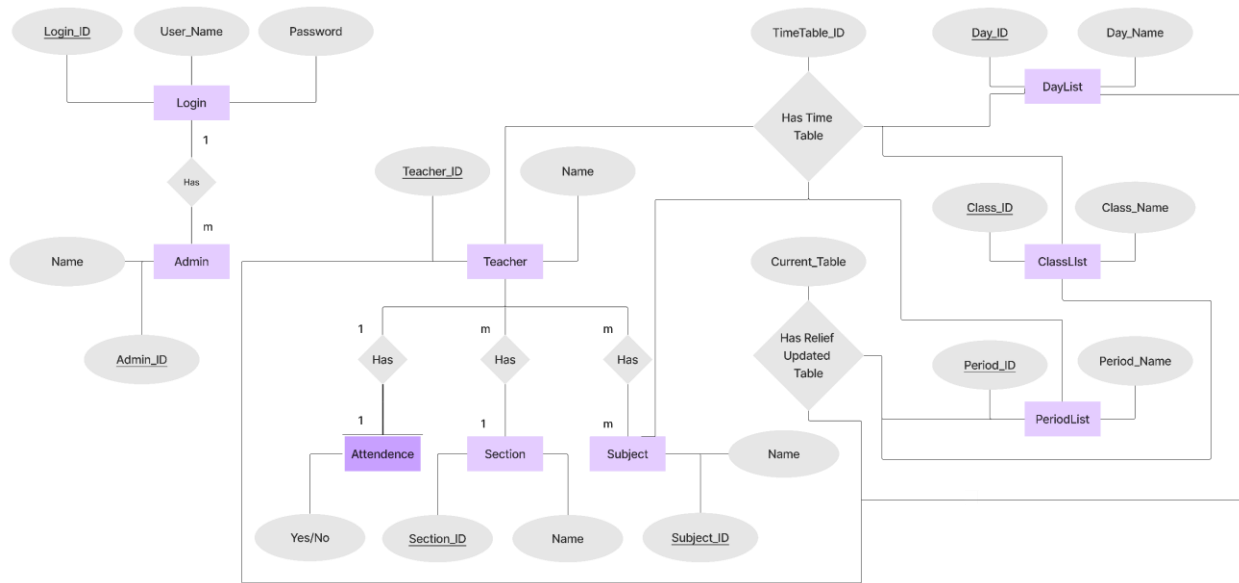


Figure 5.1 illustrates a visual representation of the entities (or objects) within a database system and the relationships between those entities.

6. System Design

6.1 Architectural Goals and Constraints

The architectural foundation of the ScheduFlex is rooted in key goals and constraints that drive its design and implementation.

Time-Conscious Development:

The system is designed to meet objectives within a defined timeline, balancing feature-rich implementation with project completion.

Scalability and Extensibility:

Given the dynamic nature of the education field, the system should be designed to accommodate growth and evolving requirements. Scalability and extensibility are paramount, enabling seamless integration of new functionalities and efficient expansion to meet changing user needs.

Portability and Accessibility:

To cater to diverse user preferences and device choices, the system should be designed for cross-device compatibility. It ensures consistent performance across various platforms.

7. Class Diagram

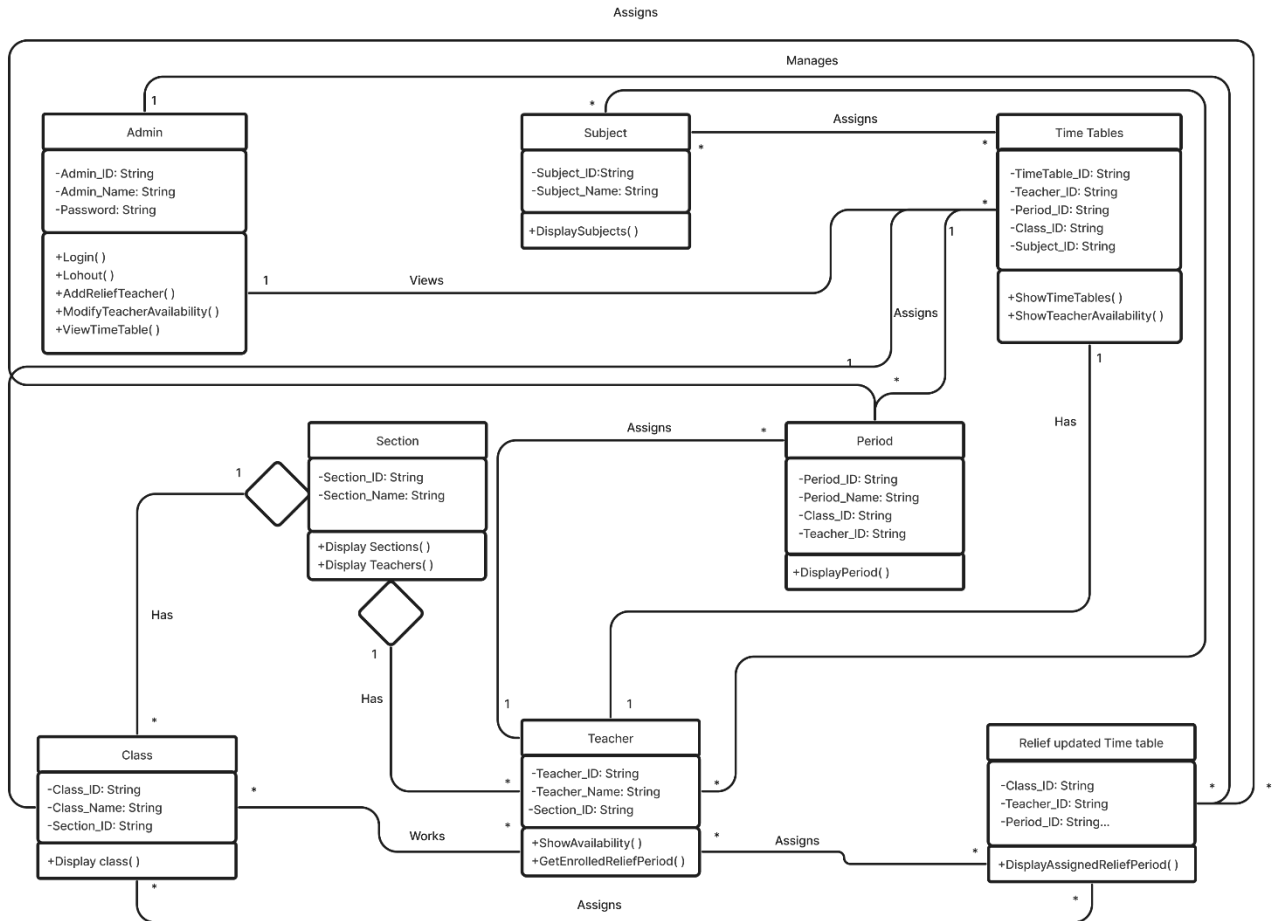


Figure 7.1 Class diagram

8. Activity Diagram

8.1 Process of Admin Registration

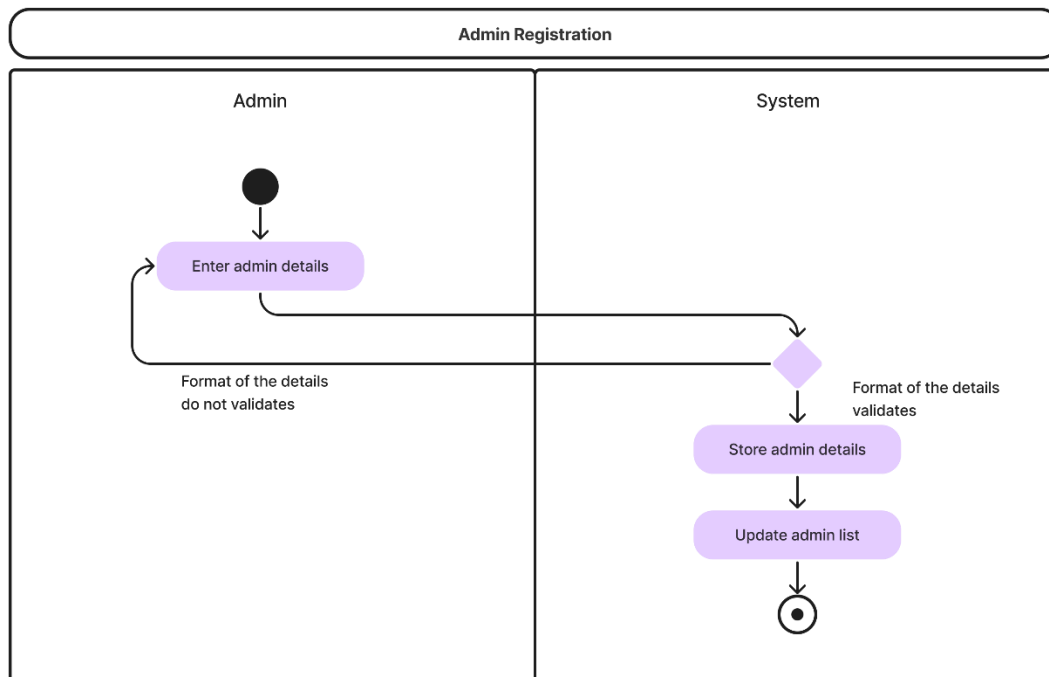


Figure 8.1 Process of admin registration.

8.2 Process of Admin Login to the System

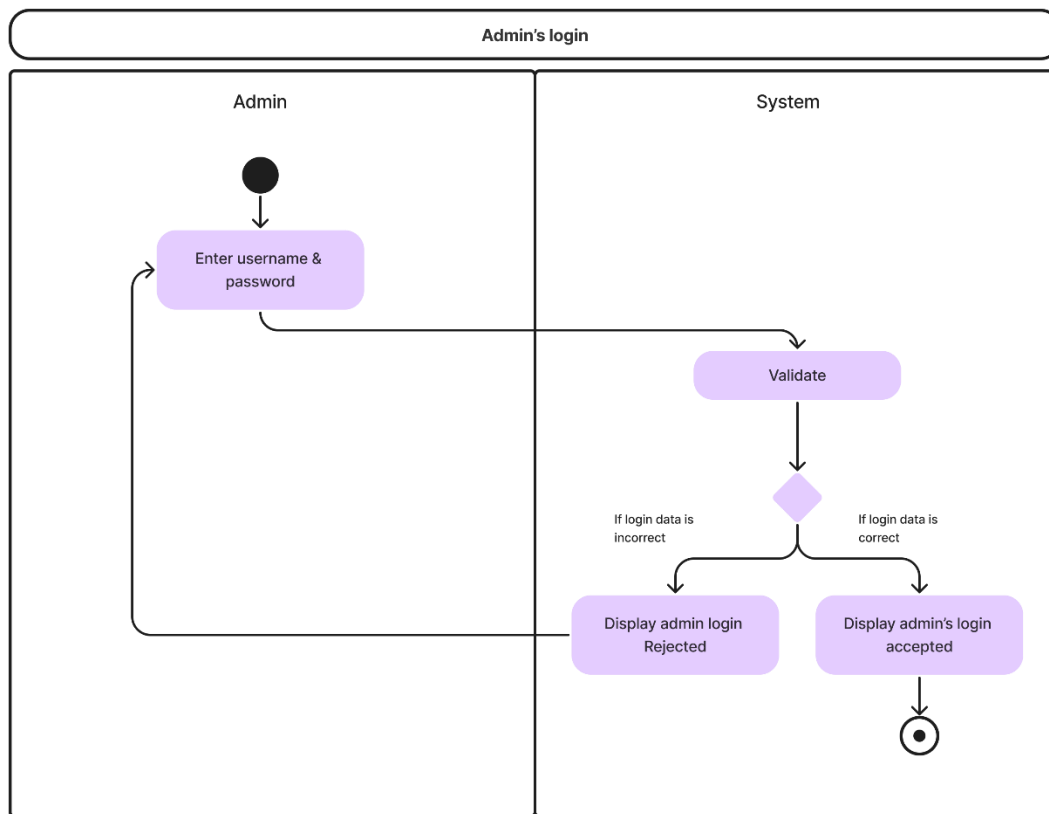


Figure 8.2 Process of admin login to the system.

8.3 Process of Adding Relief Teacher

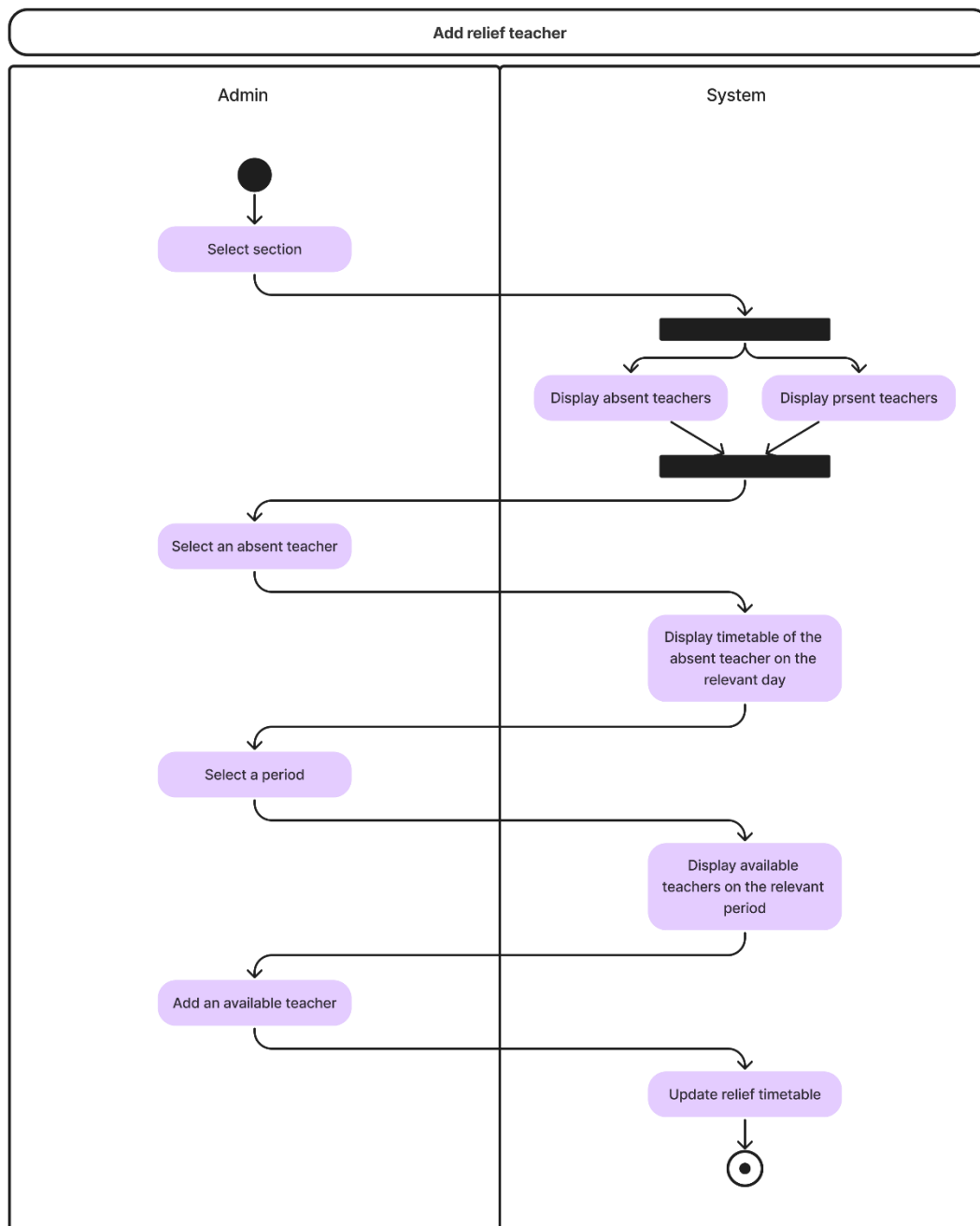


Figure 8.3 Process of adding relief teacher.

8.4 Process of Teacher Registration

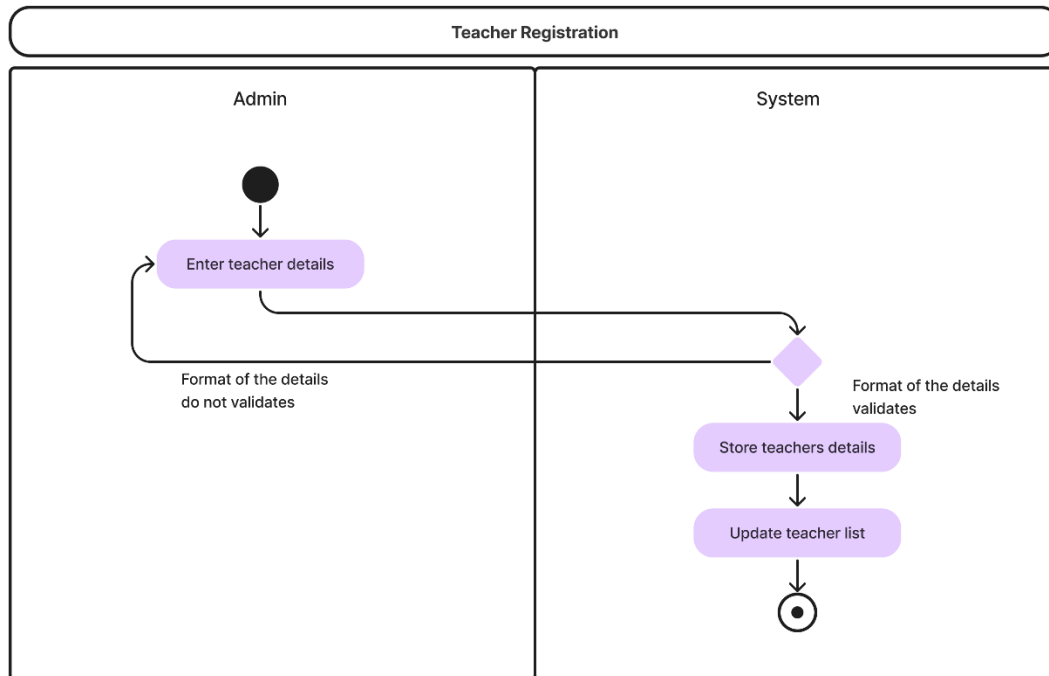


Figure 8.4 Process of teacher registration.

8.5 Process of Viewing the Updated Timetable of the Available Teacher

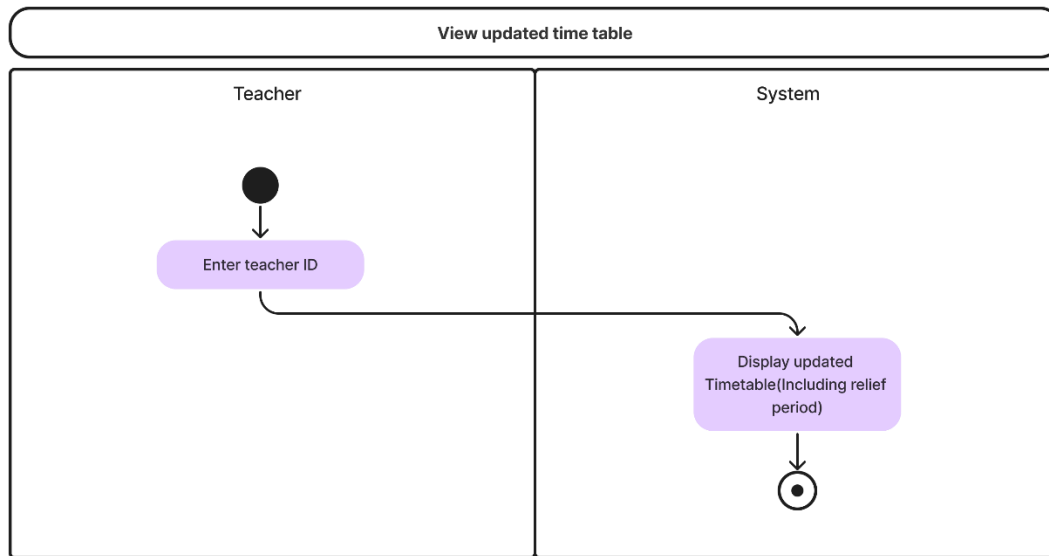


Figure 8.5 Process of viewing the updated timetable of the available teacher.

9. Sequence Diagram

9.1 Teachers Viewing the Tima Table

This sequence diagram illustrates how Teachers view their updated timetable. The User Interface collects the teacher Id and processes the data. The Database shows the updated timetable of specific teacher. This interaction ensures a smooth and efficient view experience within the system.

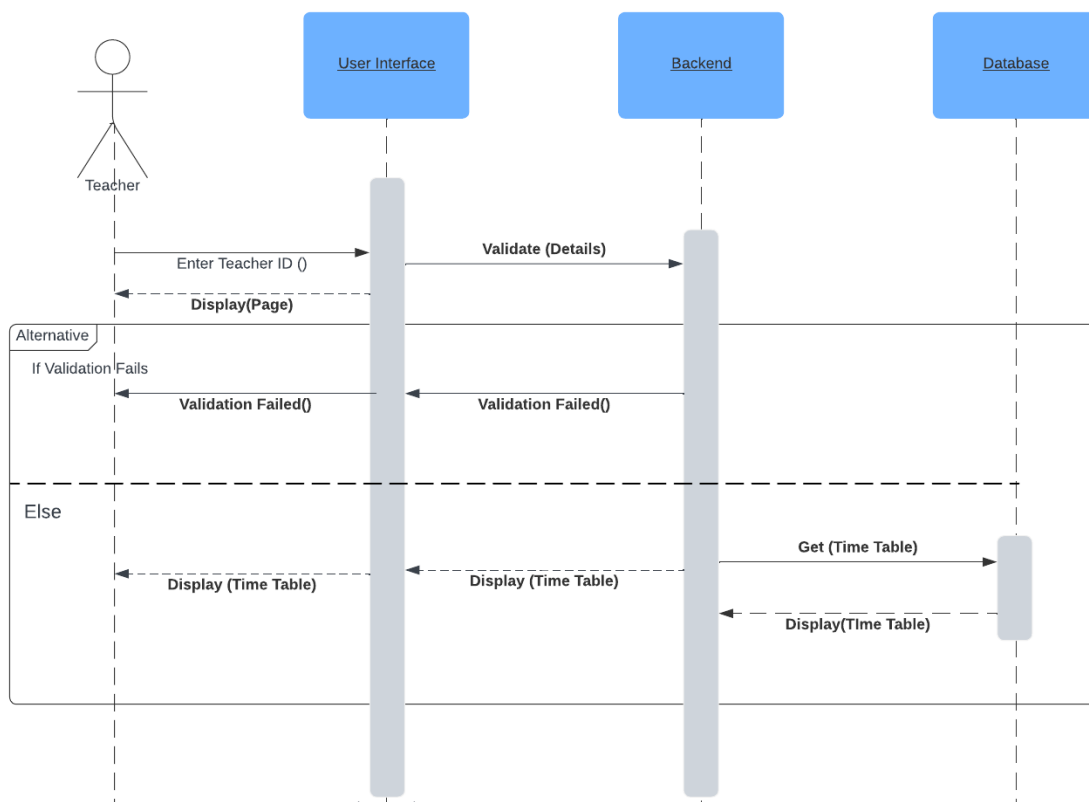


Figure 9.1 Illustrates how Teachers view their updated timetable.

9.2 Admin Assigning the Reliefs

This sequence diagram showcases the process of assigning the reliefs within the system. First, admin selects the section, and the system displays a list of present and absent teachers. When admin clicks a teacher, the system will validate the information and the database will display the current timetable. When admin selects a specific period, the system validates, the database filters a list of free teachers to the specific period and displays it to the interface. After assigning the reliefs the system validates the information and updates it to a time sensitive database. The Backend processes the information, updates the Database, and generates a display message.

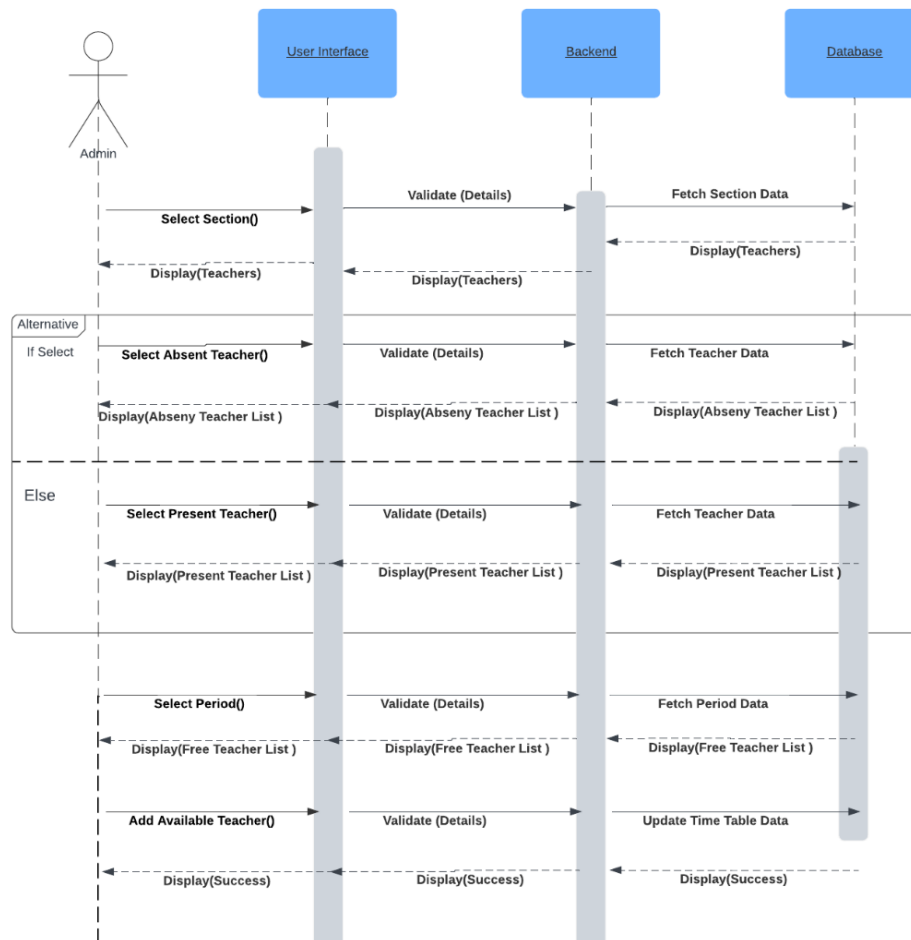


Figure 9.2 Illustrates the process of assigning the reliefs within the system.

9.3 Admin Login

This sequence diagram outlines the steps when admin login to the system. The system provides an interface to login and when admin enters the username and password it will validate the login information and display message.

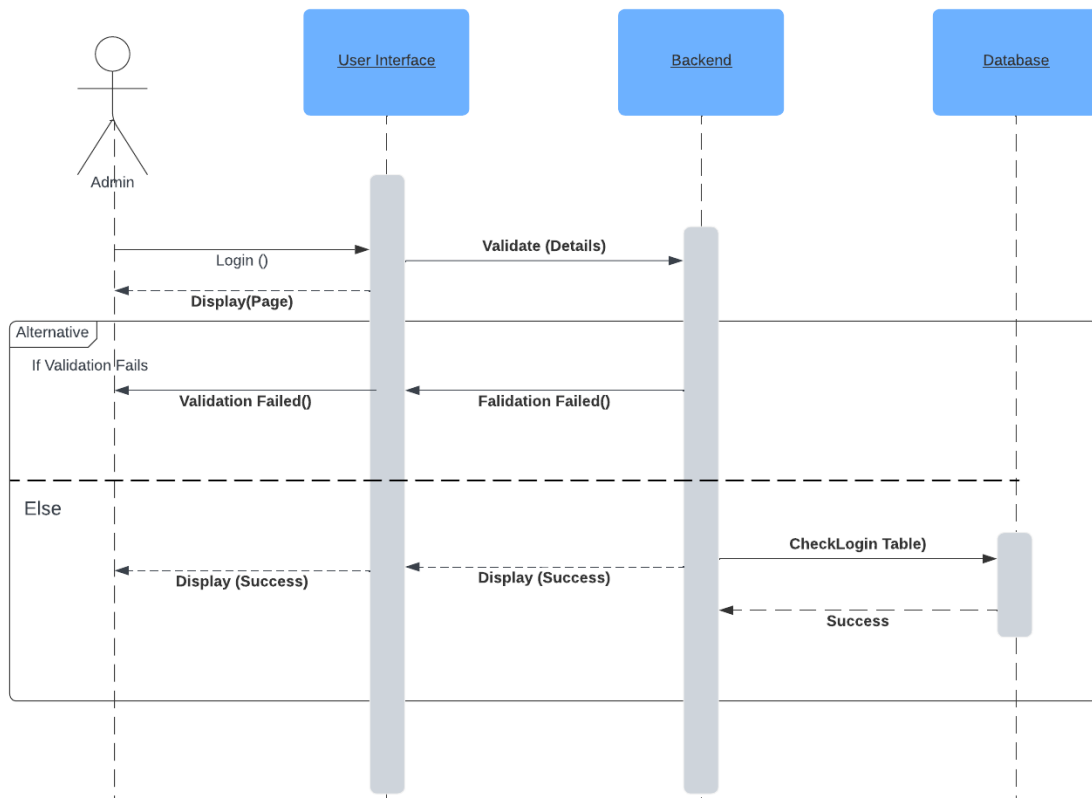


Figure 9.5 Illustrates the steps when admin login to the system.

9.4 Register a Teacher

This sequence diagram illustrates the procedure for registering the teachers. Admin inputs the details to the given interface, then the system will validate the information and updates the databases. The Backend validates and processes the information, updating the Database.

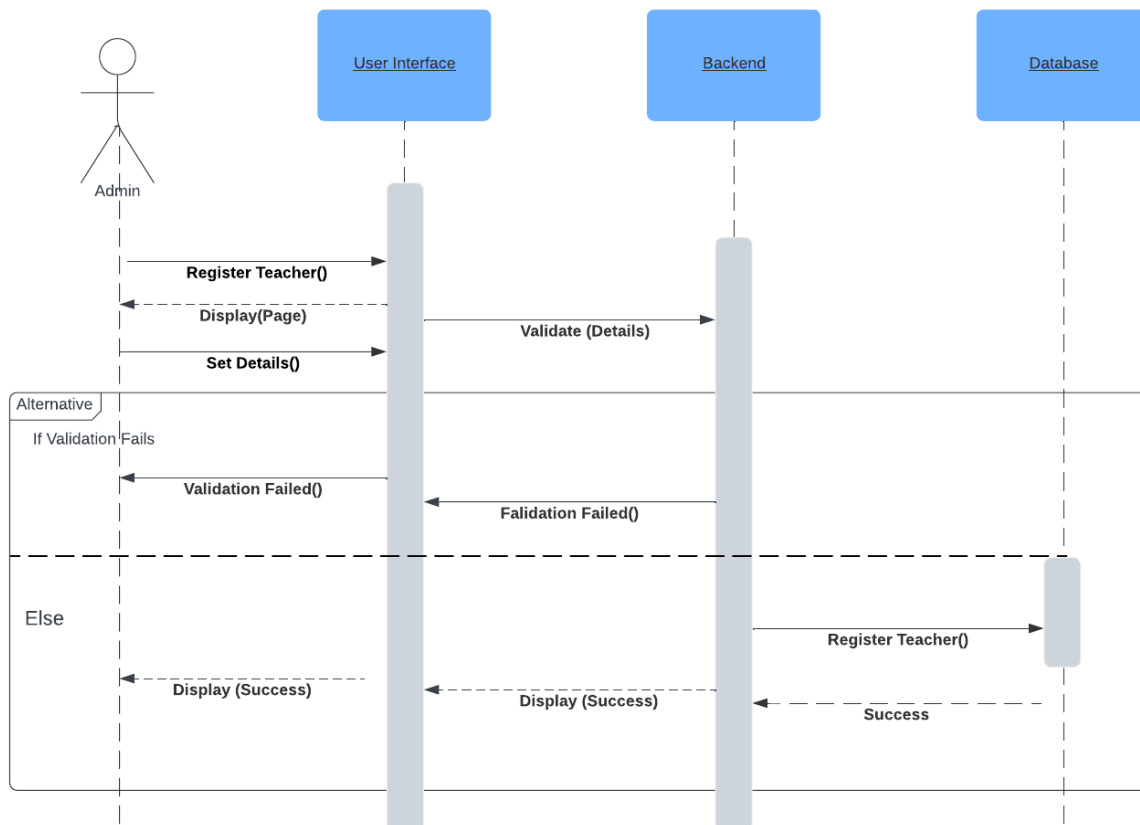


Figure 9.4 illustrates the procedure for registering the teachers.

9.5 Register an Admin

This sequence diagram illustrates the step to register an admin. Admin inputs the details to the given interface, then the system will validate the information and updates the databases. The Backend validates and processes the information, updating the Database.

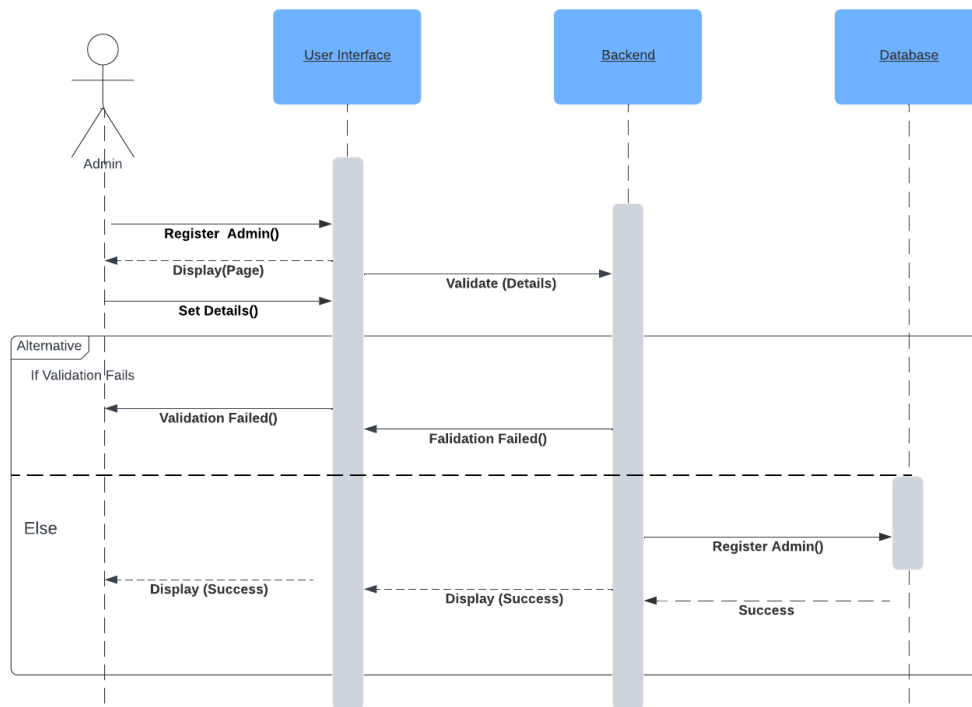


Figure 9.5 Illustrates the step to register an admin.