**Software Requirements and Design Document**

**for**

**Student Societies Management System Application**

**Prepared by Ahmed Murtaza Malik, Ahmed Hannan, and Zain bin Asad**

**FAST NUCES**

**10th November, 2024**

**Table of Contents**

**Table of Contents ii**

**1.** **Introduction 1**

1.1 Purpose 1

1.2 Product Scope 1

1.3 Title 1

1.4 Objectives 1

1.5 Problem Statement 1

**2.** **Overall Description 1**

2.1 Product Perspective 1

2.2 Product Functions 2

2.3 List of Use Cases 2

2.4 Extended Use Cases 2

2.5 Use Case Diagram 2

**3.** **Other Nonfunctional Requirements 2**

3.1 Performance Requirements 2

3.2 Safety Requirements 2

3.3 Security Requirements 2

3.4 Software Quality Attributes 2

3.5 Business Rules 3

3.6 Operating Environment 3

3.7 User Interfaces 3

**4.** **Domain Model 3**

**5.** **System Sequence Diagram 3**

**6.** **Sequence Diagram 3**

**7.** **Class Diagram 4**

**8.** **Package Diagram 4**

**9.** **Deployment Diagram 4**

# **Introduction**

## **Purpose**

Student Societies Management System Application stands to create a seamless interface for societies and administration of the institute in order to streamline event scheduling. While also providing a viable portal for the students to discuss matters and for societies to inform students of any events or changes. Fixing issues that plague the management of all student societies today. Designed for university students, it deals with their day-to-day problems in managing events and handling hassles that occur due to miscommunication.

## **Product Scope**

Designed to aid society registration within FAST University, event scheduling and announcements, it will improve efficiency and reduce delays. Furthermore, having details of all events in a single place will also prevent clashes amongst societies. Ensuring success of events that are essential to the public image of FAST NUCES whilst decreasing the hassle that administration has to endure for all proceedings.

## **Title**

A one stop solution to all external and internal issues within a society.

## **Objectives**

Amongst making the society joining process easier for students, it aims to:

* Reduce delays caused by administration in time crucial events.
* Prevent clashes in reservations
* Streamline society approval and registration process
* Reduce workload on FAST administration regarding society matters
* Maintaining records of successful events to prove credibility of societies

## **Problem Statement**

The need for a one stop solution to society problems has become apparent over the years as more and more events get organized by an ever-increasing number of societies. Leading to an increased workload for the administration in ensuring a smooth process. A software solution that is not compute heavy, it can be run on existing architecture and will cut down delays and clashes for events and bookings by reducing manual work.

# **Overall Description**

## **Product Perspective**

An independent complete software solution to the common problems hindering societies from functioning as smoothly as possible. It augments the FAST university student database with a detail of societies and their events. An aided view of the functionalities will be highlighted in the diagrams attached below, such as the class diagram.

## **Product Functions**

* Reduced administrative wait times: Created an efficient process to guarantee quick approvals and resolutions for urgent situations.
* Implemented a centralized scheduling system to avoid double bookings of venues and resources.
* Established a platform that is user-friendly to simplify and speed up society registration and approval processes in an automated manner.
* Decreased administrative tasks: Automated everyday society-related responsibilities, lessening the workload for FAST administration.
* Event record management: Implemented a system for storing detailed records of previous events, which aids organizations in showcasing their trustworthiness.

## **List of Use Cases**

* Student comments on a post
* Student applies to a society
* Student leaves a society
* Student/Society makes a post
* Student/Society updates their profile
* Society organizes an event
* Society makes an announcement
* Society manages members
* Society registers itself
* Society dissolves itself
* Society books a venue

## **Extended Use Cases**

Explained in detail in the file uploaded alongside this report, kindly refer to the file to avoid extra repetition.

## **Use Case Diagram**

As directed by the report itself, this diagram is in the deliverable submitted alongside the report itself.

# **Other Nonfunctional Requirements**

## **Performance Requirements**

Requiring Java, preferably the latest release, MySQL 9.0.1 and MySQL Connector jar file for 9.0.1, it has no other requirements and is not compute intensive, making it friendly for installation in the hardware available to FAST management.

## **Safety Requirements**

As the program is meant to keep track of students in societies and track their interactions. It would be advisable for FAST management to log in with admin details, and ensure no student has access to those details so that access is not given to unwanted users.

## **Security Requirements**

With no current internet connectivity, its safety can be ensured as no external party can spoof the data generated, furthermore, MySQL’s own encryption and safety procedures ensure another layer of data security preventing any unwanted changes.

## **Software Quality Attributes**

Following GoF and GRASP patterns it ensures that the code is easy to understand and changes when needed can be implemented without having to change all other functionalities as well. Making it easy to append more functions as required. Furthermore, it has a graphical user interface to ensure that it is easy to use. Model View Controllers ensure robustness and aid the testability of the entire project as well.

## **Business Rules**

Societies first need to register themselves, after which they can make announcements and proceed to make edits to said announcements. It can organize an event, book a venue, update its own profile and accept students into itself or remove students from itself.   
The administration has full control over these actions and can choose whether a post can be added or a society can even be registered in the first place.

Students can apply to join societies, leave societies and update their own profiles as needed.

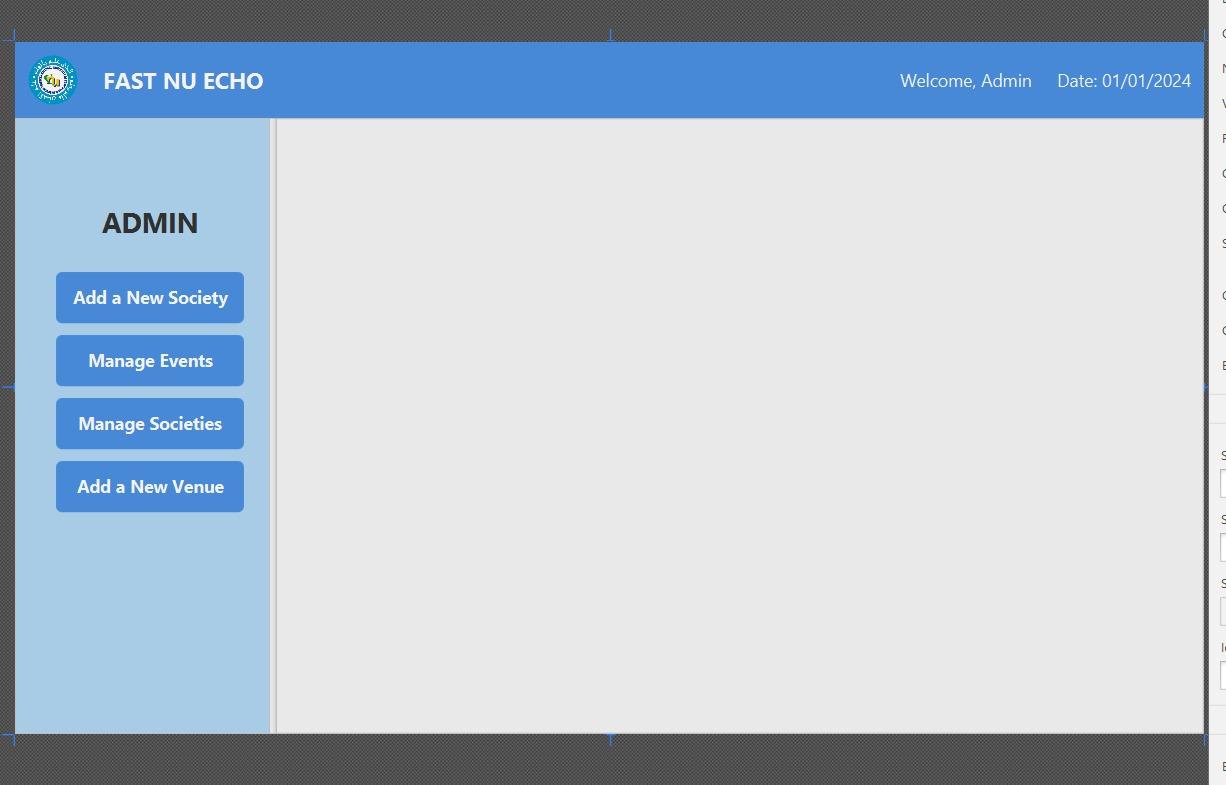
## **Operating Environment**

Requiring MySQL 9.1.0 it can operate on any device where that is installed and can be integrated seamlessly with the java code where the MySQL connector is present for version 9.1.0. This allows it to operate on any operating system with minimum hardware requirements.

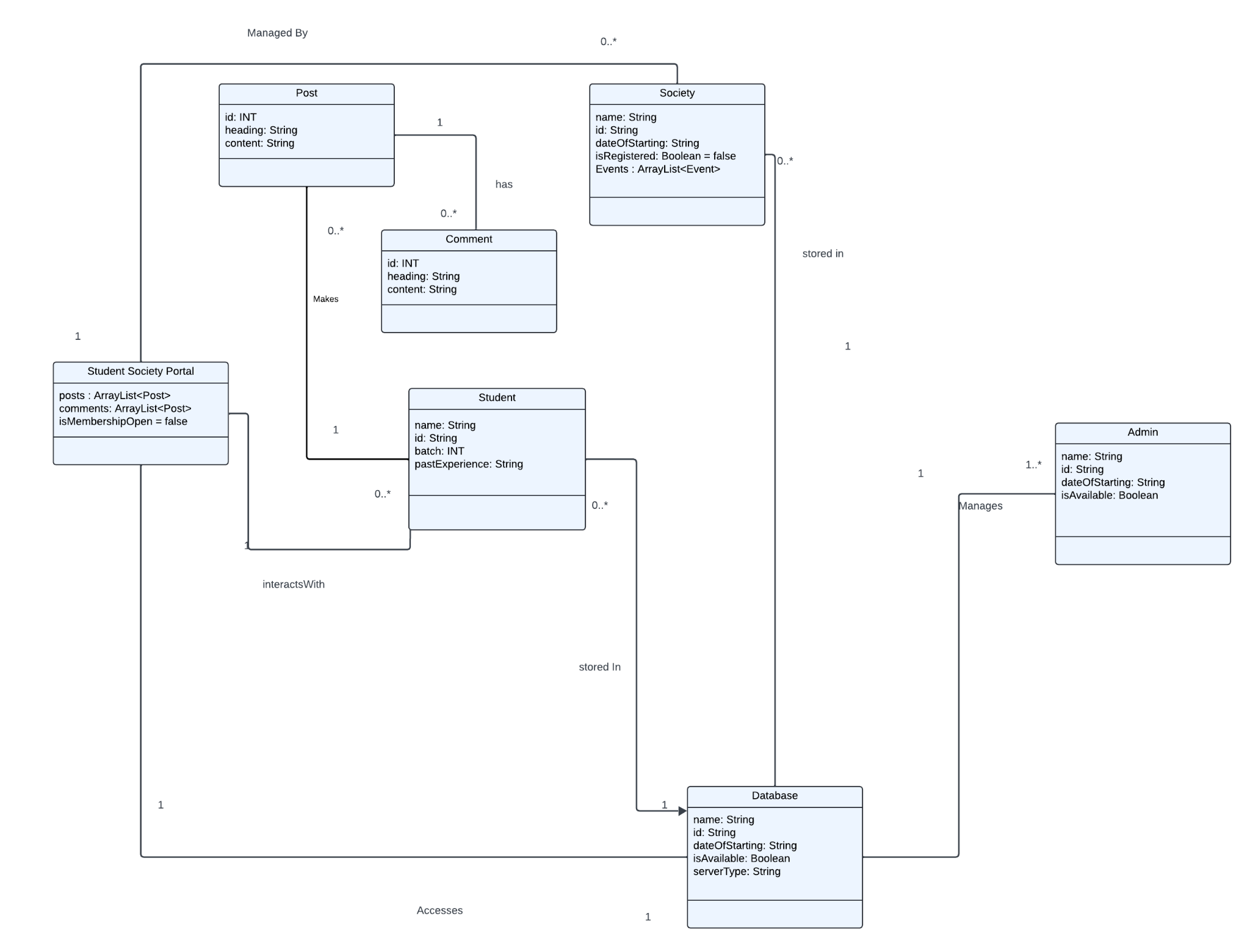
## **User Interfaces**

Designed for FAST NUCES it follows the design choices of Flex, one of the most integral software used by students throughout the university, a dashboard exists on the left with details for the specific user and allows navigation to multiple screens.

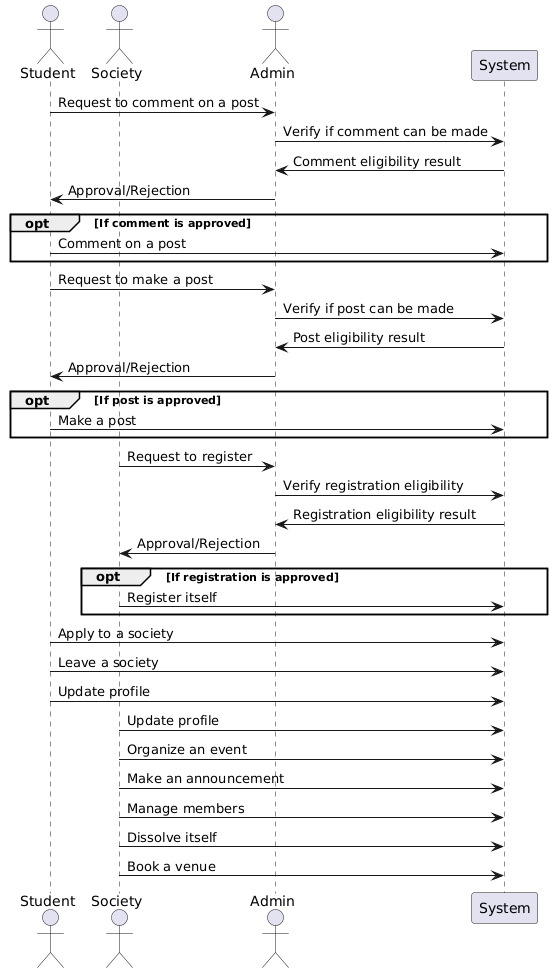
For all of the options in the left panel, the right side of the screen opens up details and options for that respective panel. This design style is kept consistent for all three users: admin, student and society.



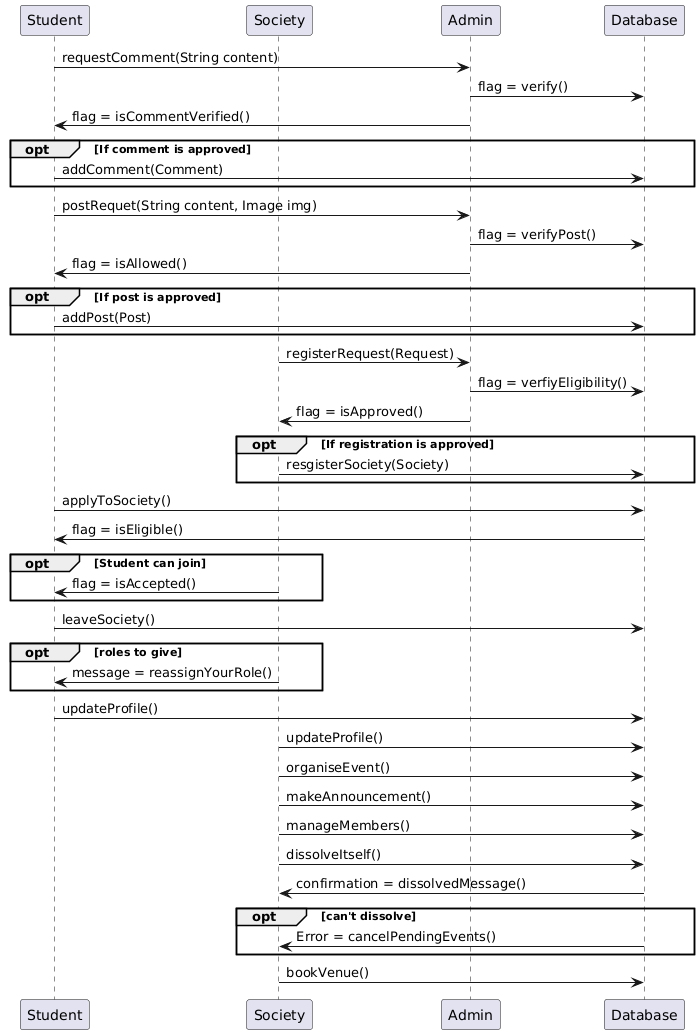
# **Domain Model**



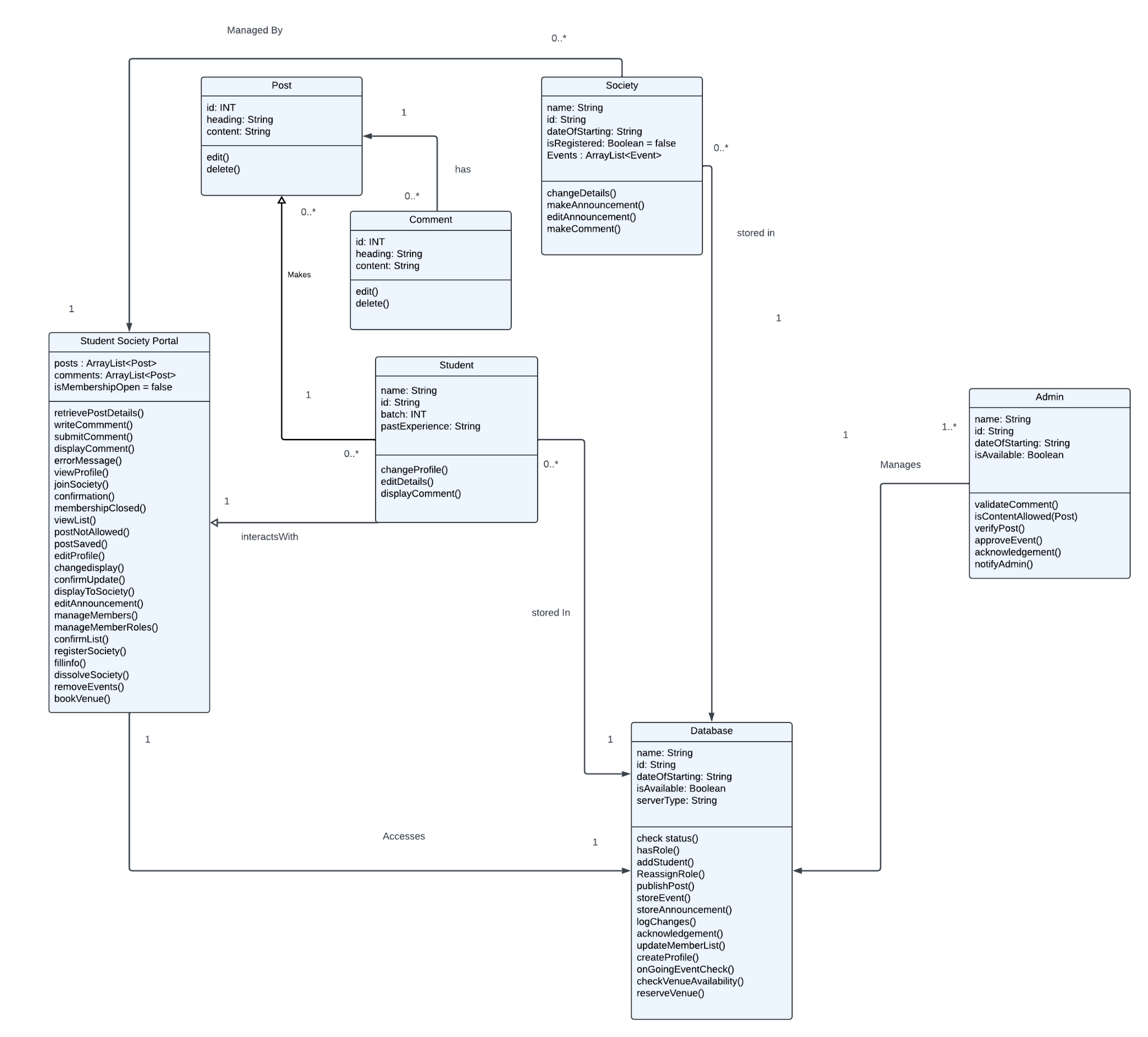
# **System Sequence Diagram**



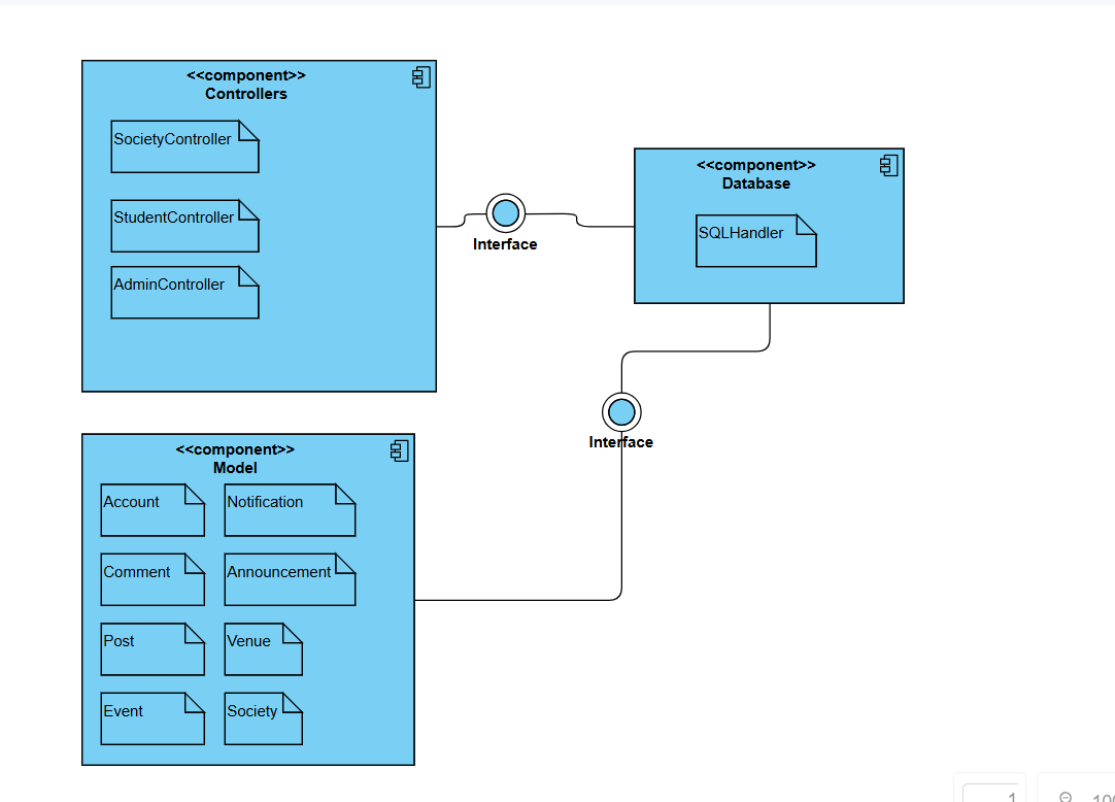
# **Sequence Diagram**



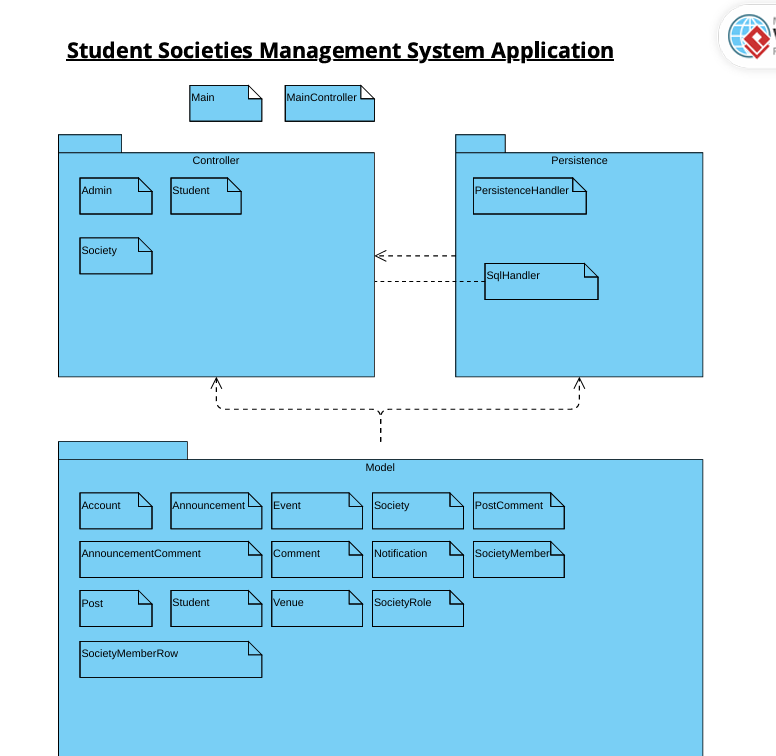
# **Class Diagram**



# **Component Diagram**



# **Package Diagram**

****

# **Deployment Diagram**

