Chapter 5

System & Database Design

**System Design**

Systems design is the process of defining elements of a system like modules, architecture, components and their interfaces and data for a system based on the specified requirements. It is the process of defining, developing and designing systems which satisfies the specific needs and requirements of a business or organization.

A systemic approach is required for a coherent and well-running system. Bottom-Up or Top-Down approach is required to take into account all related variables of the system. A designer uses the modelling languages to express the information and knowledge in a structure of system that is defined by a consistent set of rules and definitions. The designs can be defined in graphical or textual modelling languages. Some of the popular examples of graphical modelling languages are:

* Unified Modelling Language
* Flowchart etc.

Unified Modelling Language

UML stands for Unified Modeling Language. It’s a rich language to model software solutions, application structures, system behavior and business processes. The purpose of a use case diagram in UML is to demonstrate the different ways that a user might interact with a system. There are two main categories i.e. Structure diagrams and Behavioral diagrams.

* Structure diagrams show the things in the modeled system. In a more technical term, they show different objects in a system.
* Behavioral diagrams show what should happen in a system. They describe how the objects interact with each other to create a functioning system.

Design methods:

Design methods are procedures, techniques, aids, or tools for designing. They offer a number of different kinds of activities that a designer might use within an overall design process.

Some popular design methods:

* Architectural design
* Logical design
* Physical design

USE CASE DIAGRAM (Lu Exam Hive)

Use case diagrams give a graphic overview of the actors involved in a system, different functions needed by those actors and how these different functions interact. It’s a great starting point for any project discussion because you can easily identify the main actors involved and the main processes of the system.

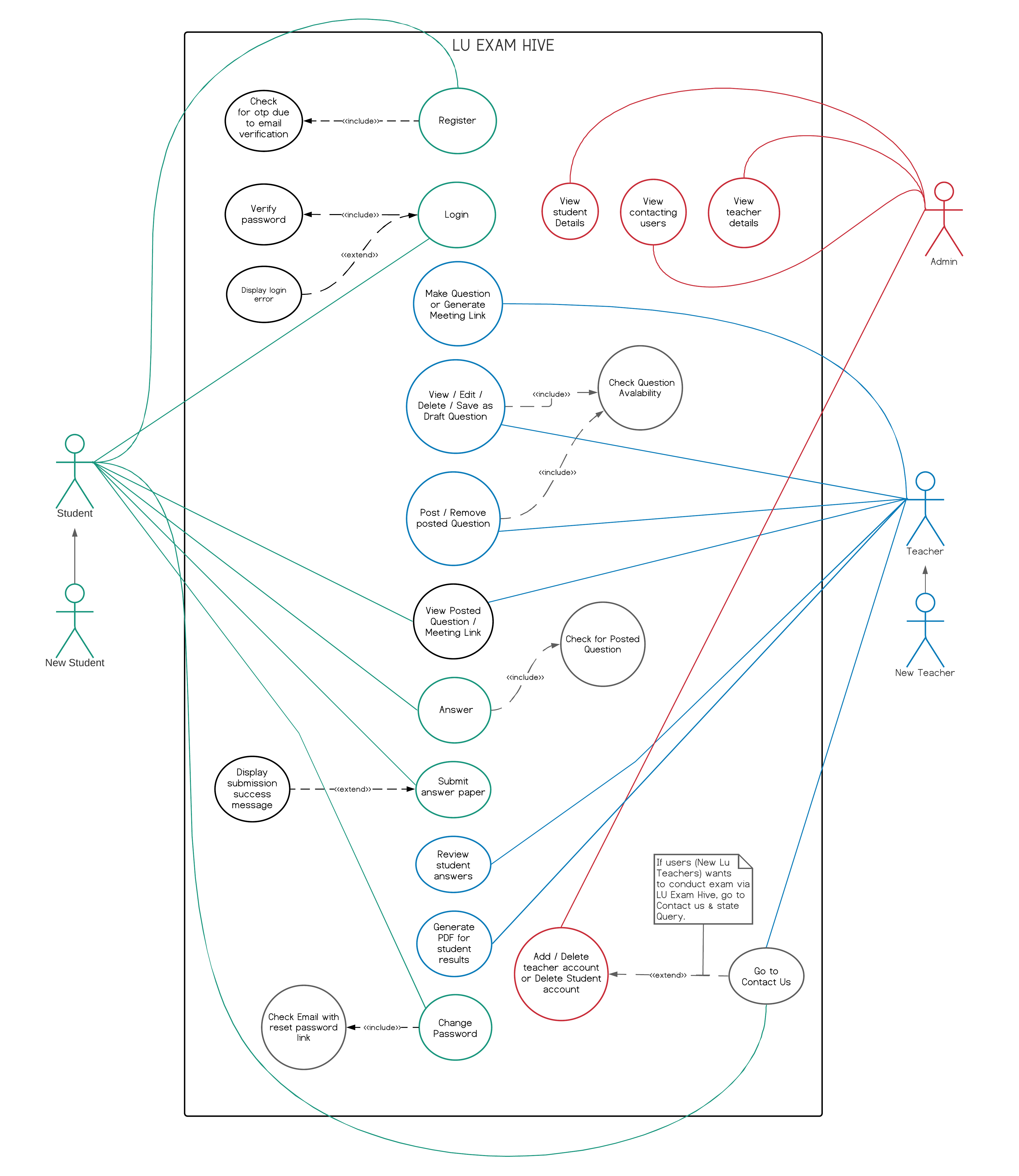


Figure 3: Use Case Diagram for Lu Exam Hive

Activity Diagram (Lu Exam Hive)

Activity diagrams represent workflows in a graphical way. They can be used to describe the business workflow or the operational workflow of any component in a system. Sometimes activity diagrams are used as an alternative to State machine diagrams. Activity diagrams are the perfect UML solution for visualizing process flows. Activity diagrams in UML are a great solution to visualize the actions, outcomes, and flows within a specific process and the behaviors that pair with them. An activity diagram is used to create a simple overview of any process to better identify areas for improvement or model software architecture to help better understand what’s going on.

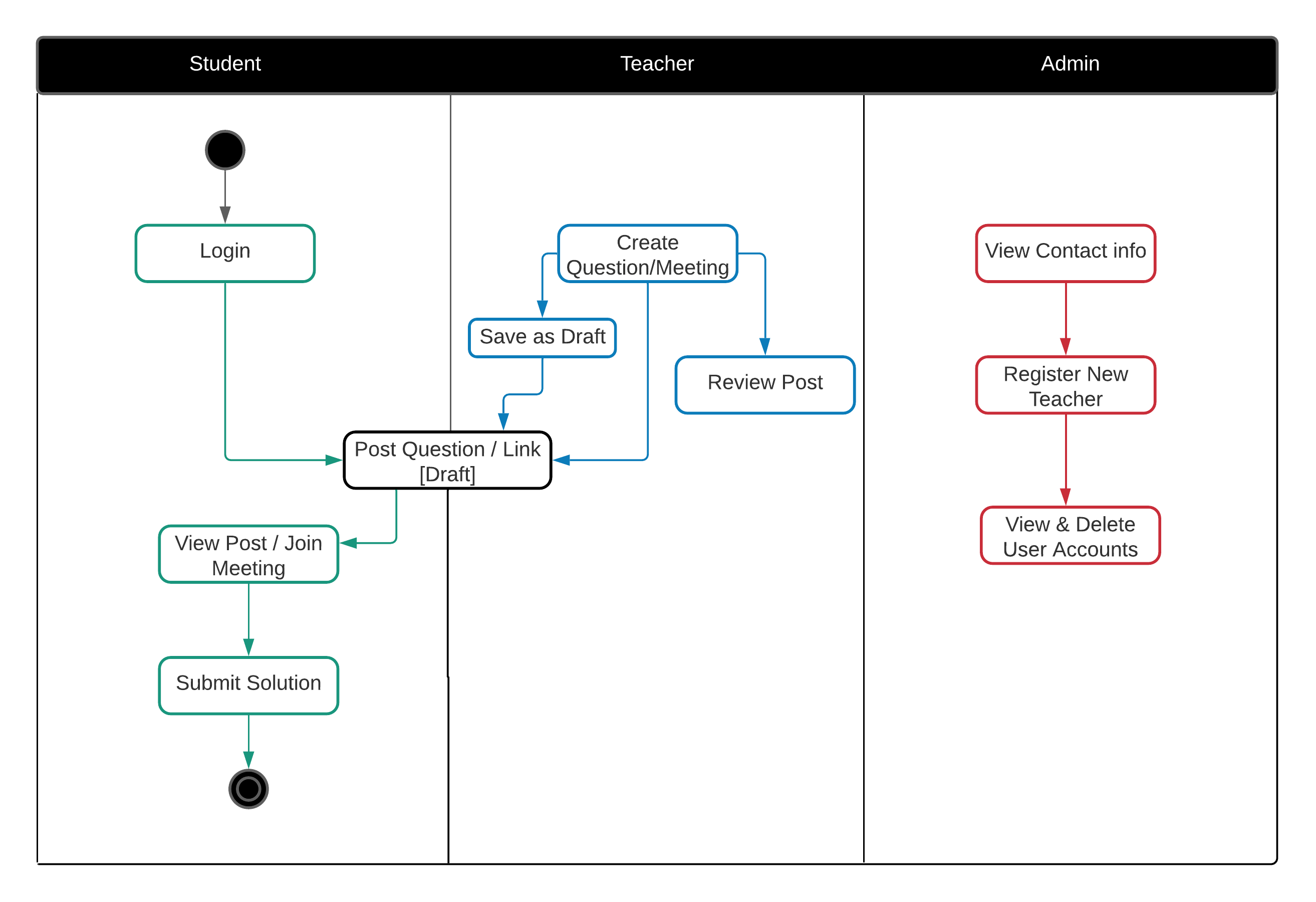


Figure 4: Activity Diagram for Lu Exam Hive

System Sequence Diagrams (Lu Exam Hive)

A system sequence diagram is a type of sequence diagram in UML. These charts show the details of events that are generated by actors from outside the system.

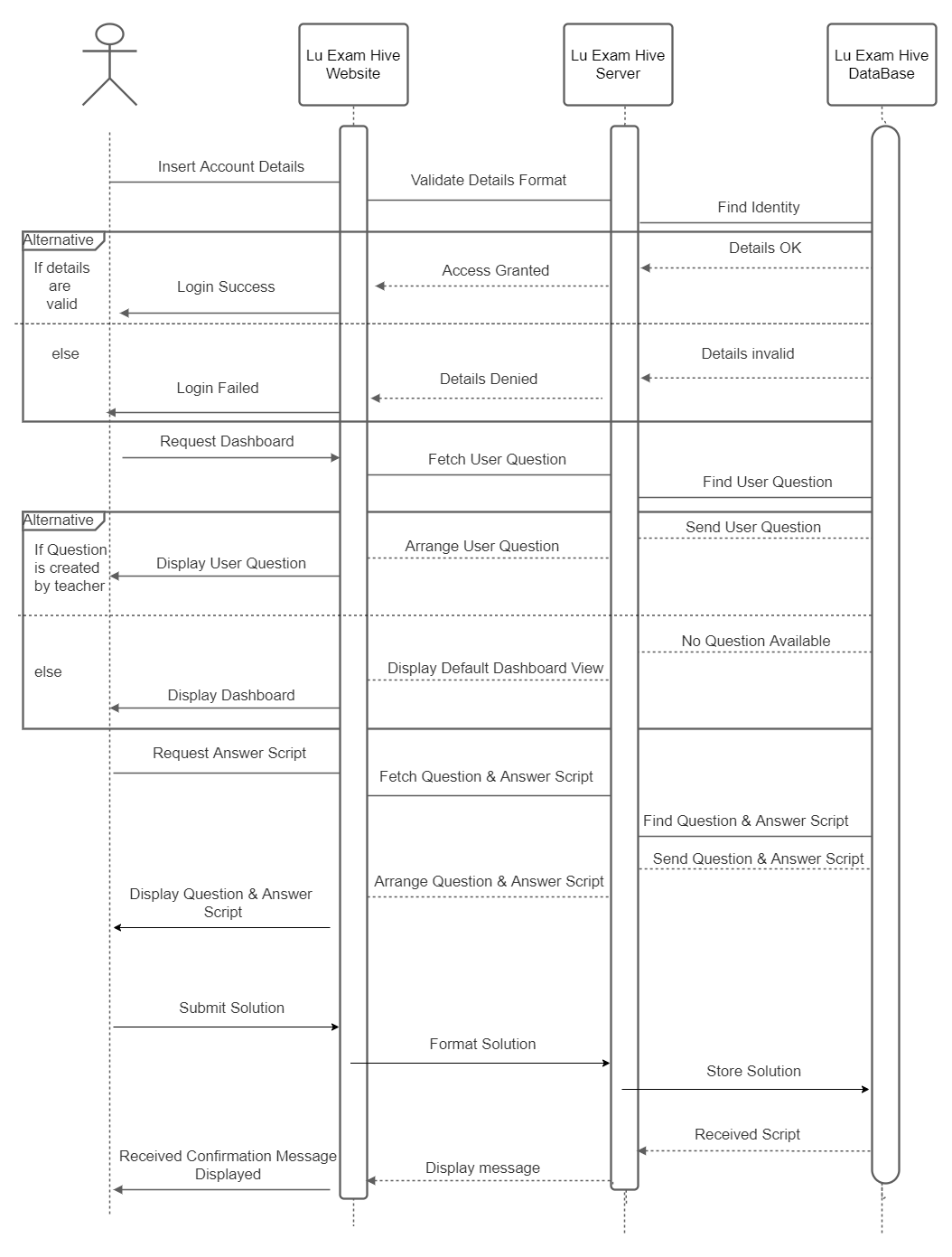


Figure 5: Sequence Diagrams for Lu Exam Hive

Class Diagram (Lu Exam Hive)

Class diagrams are the foundation for all other UML structure diagrams. Class diagrams are the main building block of any object-oriented solution. It shows the classes in a system, attributes, and operations of each class and the relationship between each class.

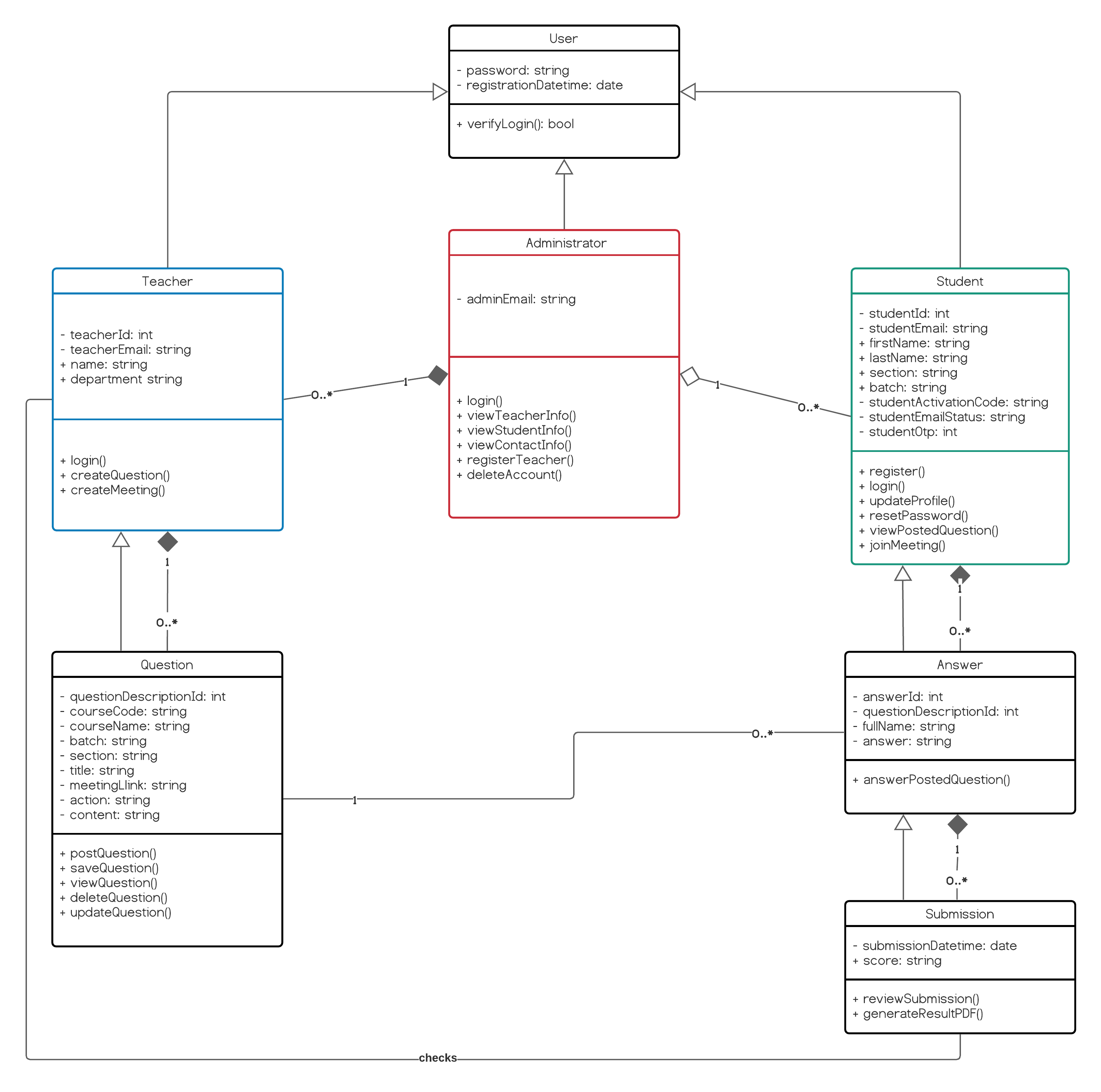


Figure 6: Class Diagram for Lu Exam Hive

**Database Design**

Database Design is a collection of processes that facilitate the designing, development, implementation and maintenance of enterprise data management systems. Properly designed database are easy to maintain, improves data consistency and are cost effective in terms of disk storage space. The database designer decides how the data elements correlate and what data must be stored. The main objectives of database designing are to produce logical and physical designs models of the proposed database system.

The logical model concentrates on the data requirements and the data to be stored independent of physical considerations. It does not concern itself with how the data will be stored or where it will be stored physically. The physical data design model involves translating the logical design of the database onto physical media using hardware resources and software systems such as database management systems (DBMS).

There are two types of Database Techniques:

* Normalization
* ER Modeling

Normalization

Normalization is a database design technique that reduces data redundancy and eliminates undesirable characteristics like Insertion, Update and Deletion Anomalies. Normalization rules divides larger tables into smaller tables and links them using relationships. The purpose of Normalization in SQL is to eliminate redundant (repetitive) data and ensure data is stored logically.

ER-Diagram

An Entity Relationship (ER) Diagram is a type of flowchart that illustrates how “entities” such as people, objects or concepts relate to each other within a system. ER Diagrams are most often used to design or debug relational databases. Also known as ERDs or ER Models, they use a defined set of symbols such as rectangles, diamonds, ovals and connecting lines to depict the interconnectedness of entities, relationships and their attributes. They mirror grammatical structure, with entities as nouns and relationships as verbs.

ER-Diagram (Lu Exam Hive)

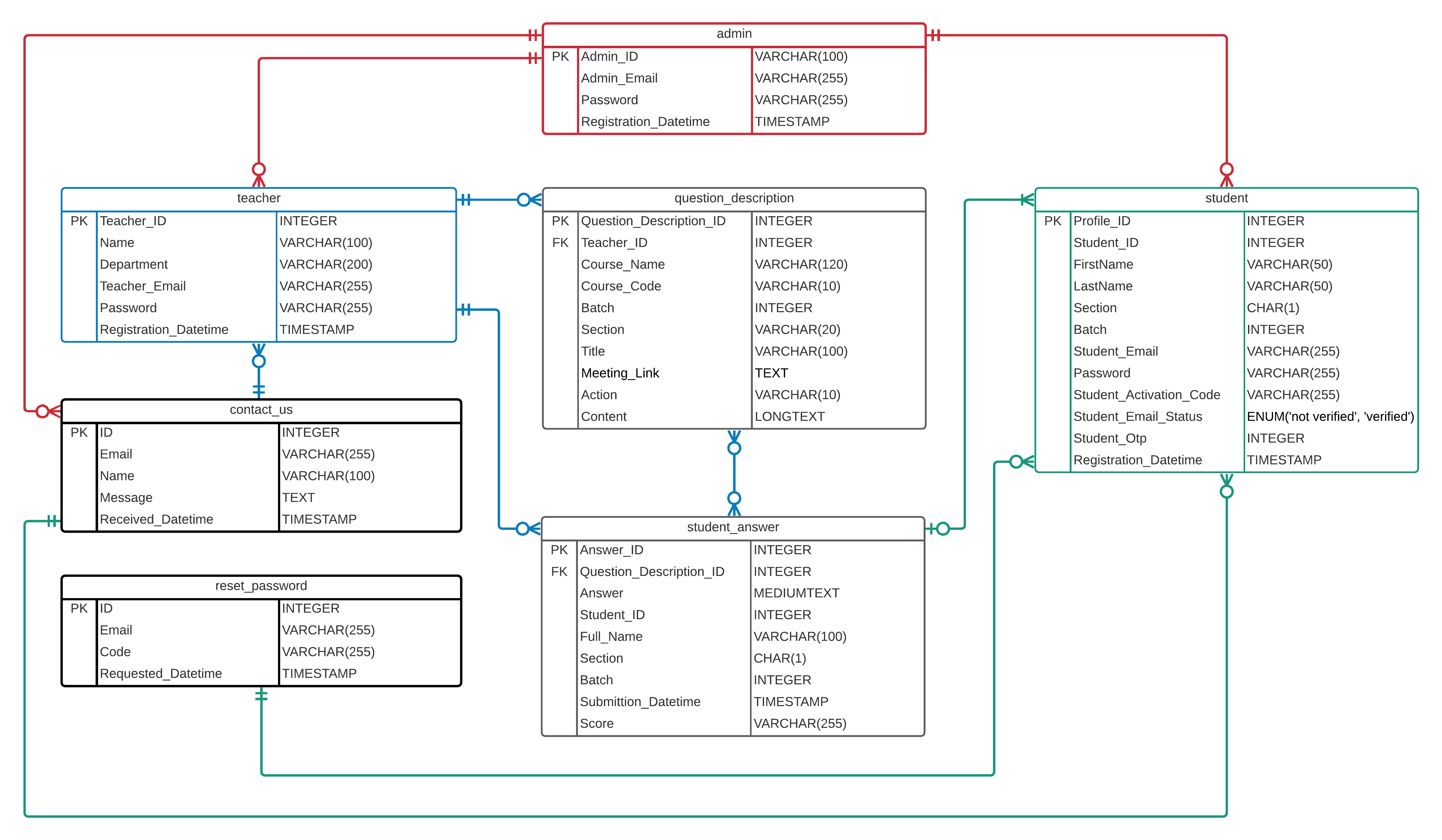


Figure 7: ER Diagram for Lu Exam Hive