Assignment 6
Sequence Model Diagram

AIM - Design and implementation of the sequence model diagram.

PROBLEM STATEMENT -
Prepare a Sequence Model.
Identify 5 major scenarios and evolve for your system.
Draw Sequence Model and UML2.0 Notations.

OBJECTIVE -
To study and use communications.
To Draw a Sequence Model.
To implement the Sequence Model.

THEORY -

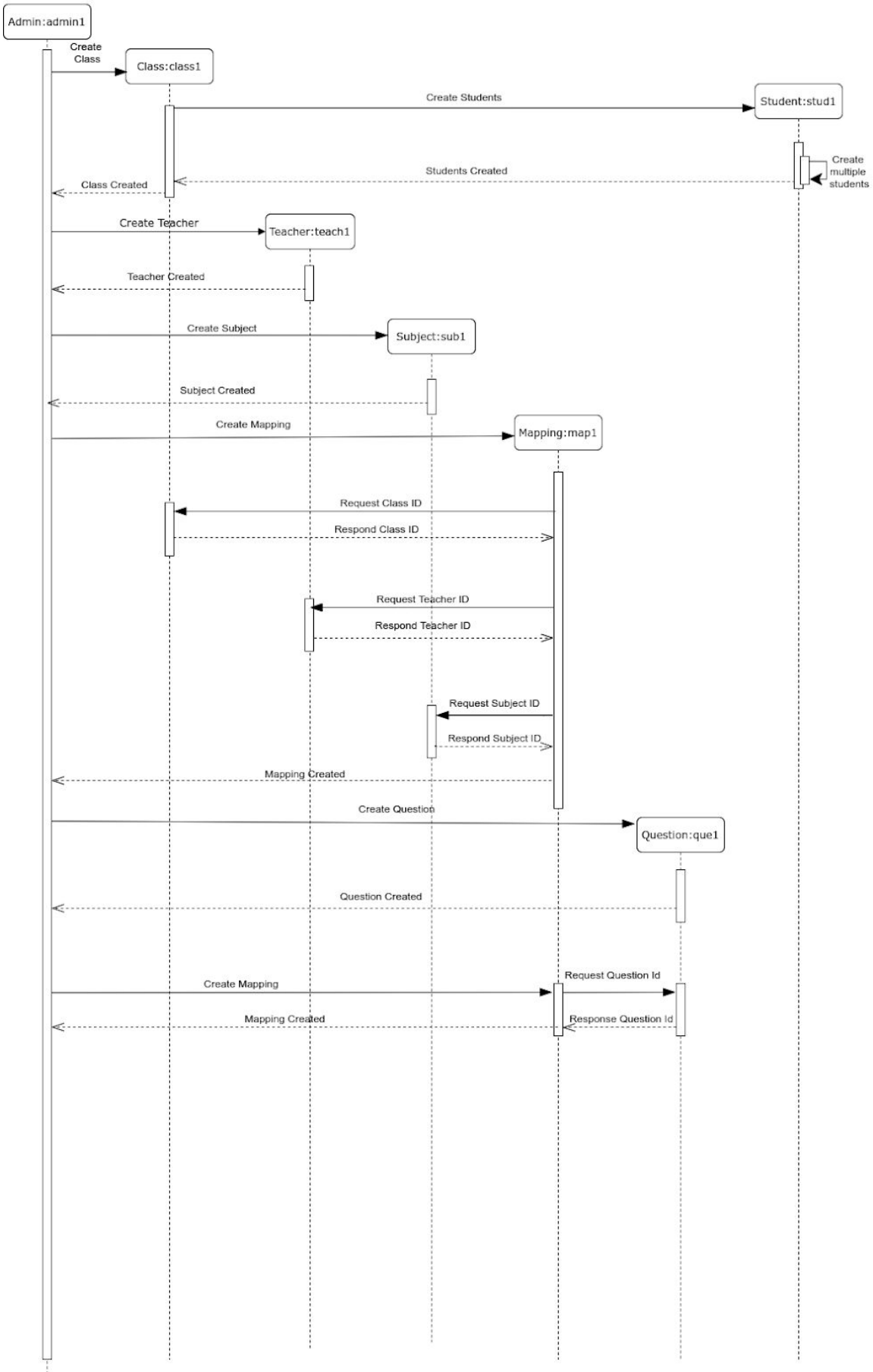
A sequence diagram is the most common kind of interaction diagram, which focuses on the message interchange between a number of lifelines, it shows object interactions arranged in time sequence. It depicts the objects involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development. Sequence diagrams are sometimes called **event diagrams** or **event scenarios**. A sequence diagram describes an interaction by focusing on the sequence of messages that are exchanged, along with their corresponding occurrence specifications on the lifelines.

A sequence diagram shows, as parallel vertical lines - lifelines, different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchanged between them, in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical manner.

The following nodes and edges are typically drawn in a UML sequence diagram:
lifeline, execution specification, message, combined fragment, interaction use, state invariant, continuation, destruction occurrence.

APPLICATIONS -
Sequence diagrams are used to explore any real application or a system.
Sequence diagrams are used to represent message flow from one object to another object.
Sequence diagrams are easier to maintain and generate.

Data Entry Scenario



In the Data Entry Scenario, the Admin class object admin1 is the first object created by the system. This object creates as well as controls all other objects of the system. It first creates the Class object which in turn creates the student object that is responsible to add multiple students in a given range of roll nos to the database. Then the teacher object is created. The Admin object creates the subject-object. Mapping objects are created by the combination of Teacher id, Subject id, and Class id. Admin then creates the Question object. And finally, this question object is mapped with the Mapping.

Generate Feedback/Password Scenario

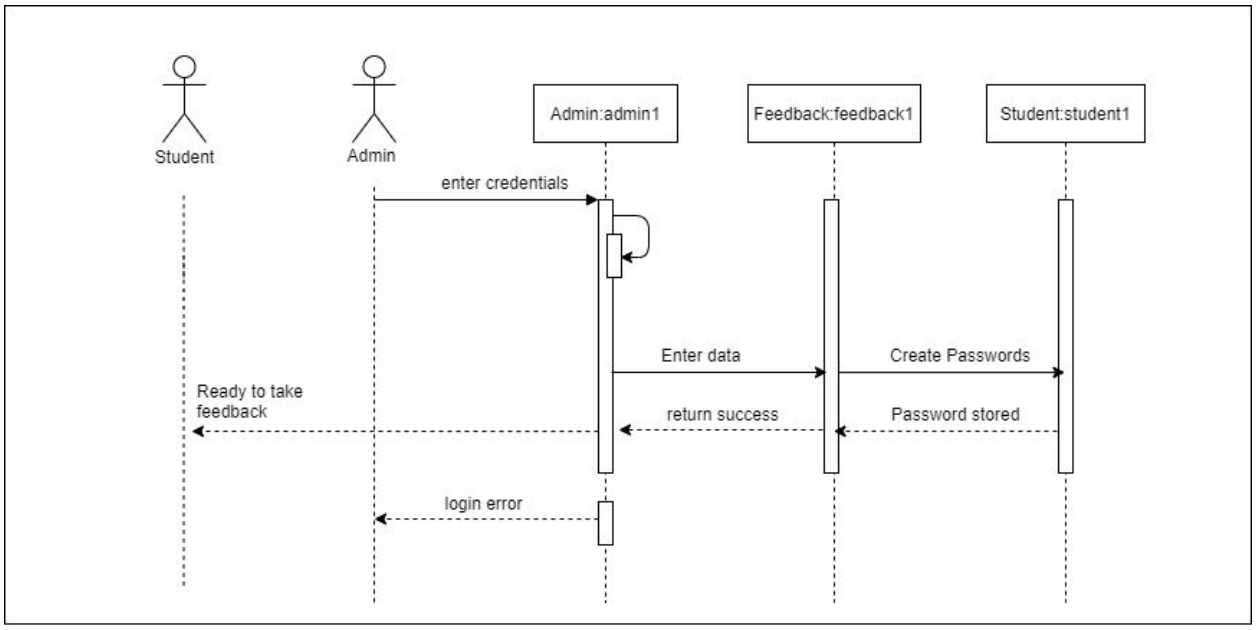


Figure : scenario - Create Feedback/Generate Password

In the Generate feedback scenario, after valid credentials for login are entered by the Admin, the Admin class object admin1 will be created which is inherited from accounts. Which further creates the Feedback object. After updating of feedback, feedback is stored, and a success message is been displayed/returned by the student1 object.After the data has been entered by the admin the password are generated and stored in the feedback class.
In case of invalid credentials entered by the admin, the login error message will be returned by the admin1 object.Now the student is ready to give the feedback.

Student Feedback Scenario

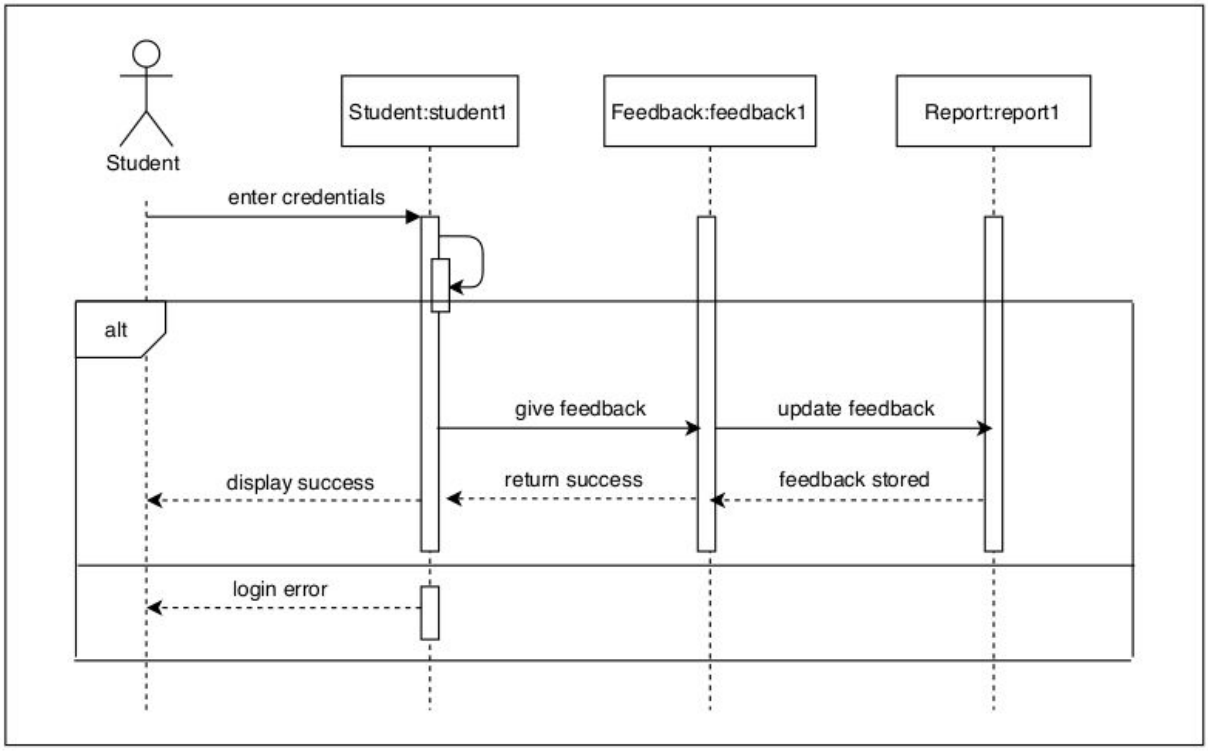


Figure : scenario - Submit Feedback

In the submit feedback scenario, after valid credentials for login are entered by the student, the Student class object student1 will be created. Which further creates the Feedback object which in turn creates the Report object. After updating of feedback, feedback is stored, and a success message is been displayed/returned by the student1 object.
In case of invalid credentials entered by the student, the login error message will be returned by the student1 object.

CONCLUSION -

Thus we have prepared a Sequence Model and studied in detail the working of the system/project. Identified different scenarios and used advanced relationships. We drew a Sequence Model using UML2.0 Notations and implemented the Sequence model with a suitable object-oriented language.