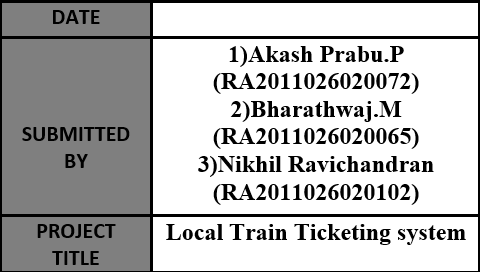
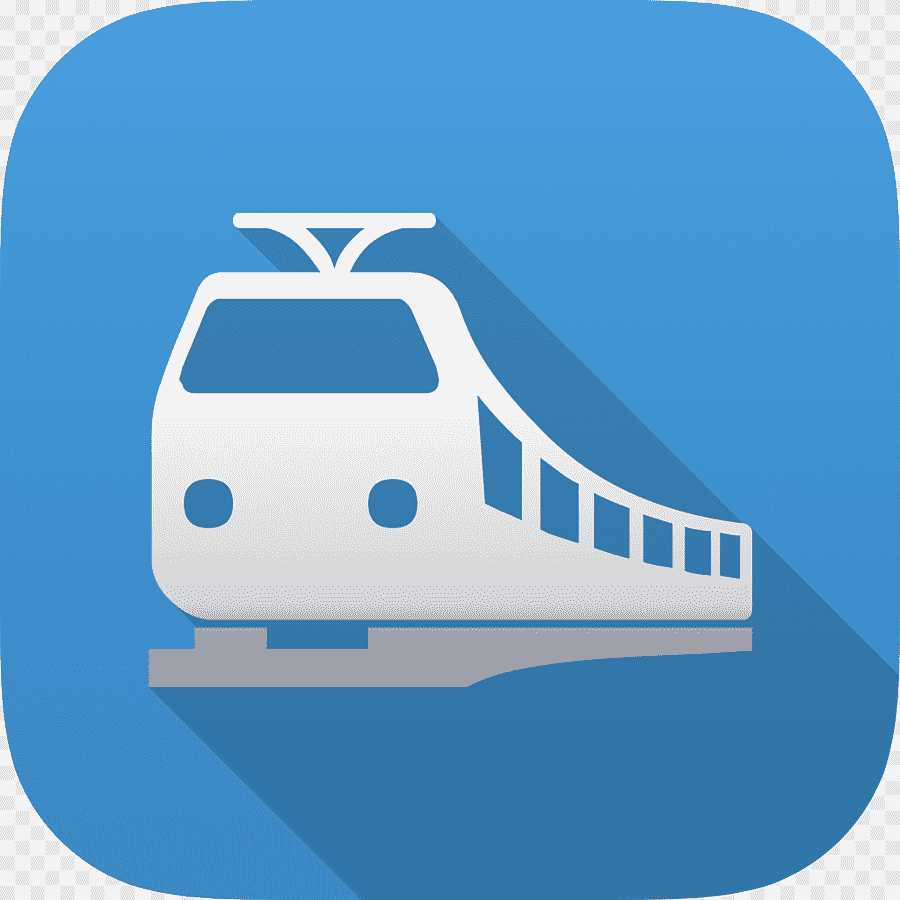
**12 MODELLING UML CLASS AND SEQUENCE DIAGRAM**

**12.1 CLASS DIAGRAM DESCRIPTION**

**CLASS DIAGRAM:**

Class diagrams are one of the most useful types of diagrams in UML as they clearly map out the structure of a particular system by modelling its classes, attributes, operations, and relationships between objects.

**COMPONENTS OF CLASS DIAGRAM:**

The standard class diagram is composed of three sections:

* **UPPER SECTION:**

Contains the name of the class. This section is always required, whether you are talking about the classifier or an object.

* **MIDDLE SECTION:**

Contains the attributes of the class. Use this section to describe the qualities of the class. This is only required when describing a specific instance of a class.

* **BOTTOM SECTION:**

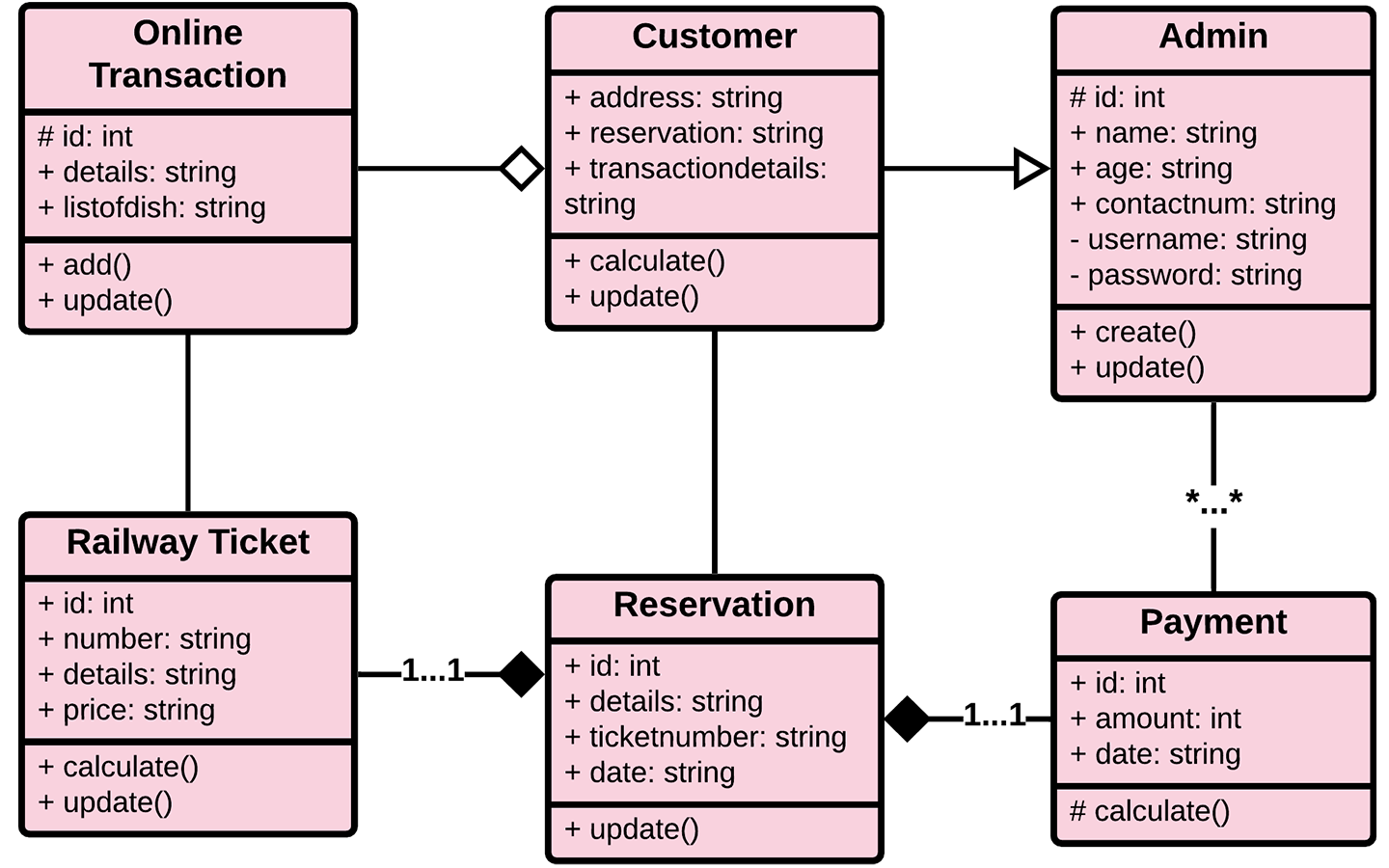
Includes class operations (methods). Displayed in list format, each operation takes up its own line. The operations describe how a class interacts with data.

**MEMBER ACCESS SPECIFIERS:**

All classes have different access levels depending on the access modifier (visibility). Here are the access levels with their corresponding symbols.

* Public (+)
* Private (-)
* Protected (#)
* Package (~)

**12.2 CLASS DIAGRAM**



**12.3 SEQUENCE DIAGRAM DESCRIPTION**

**SEQUENCE DIAGRAM:**

Sequence diagram are a popular dynamic modelling solution in UML because they specifically focus on lifelines, or the processes and objects that live simultaneously, and the messages exchanged between them to perform a function before the lifeline ends.

**COMPONENTS IN SEQUENCE DIAGRAM:**

**1] ACTOR:**

Stick figure represents the actor. Shows entities that interact the external objects of the system.

**2] OBJECTS:**

Rectangular boxes represent the object, demonstrates how an object will behave in the context of the system.

**3] ACTIVATION BOXES:**

Represents the time needed for an object to complete a task. The longer the task will take, the longer the activation box becomes.

**4] MESSAGE SYMBOLS:**

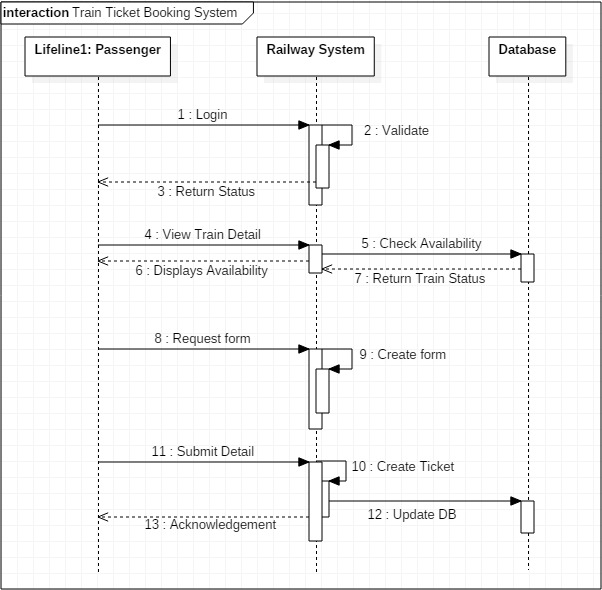
We use the following arrows and message symbols to show how information is transmitted between objects. These symbols may reflect the start and execution of an operation or the sending and reception of a signal.

* **SYNCHRONOUS MESSAGE:** Represented by a solid line with a solid arrowhead. This symbol is used when a sender must wait for a response to a message before it continues. The diagram should show both the call and the reply

.

* **ASYNCHRONOUS MESSAGE:** Represented by a solid line with a lined arrowhead. Asynchronous messages don't require a response before the sender continues. Only the call should be included in the diagram.
* **REPLY MESSAGE:** Represented by a dashed line with a lined arrowhead, these messages are replies to calls.
* **DELETE MESSAGE:** Represented by a solid line with a solid arrowhead, followed by an X. This message destroys an object.

**12.4 SEQUENCE DIAGRAM**

****