Online Course Reservation System

Submitted by

Sharmi R (185001142)

Shriya Baskaran(185001149)

Sreedhar V(185001161)

Under Supervision of

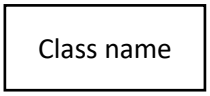
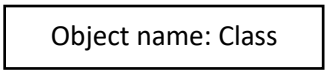
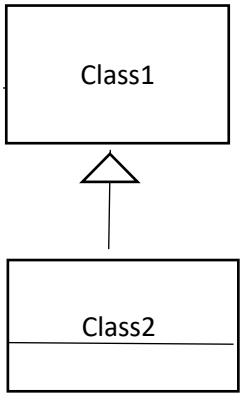
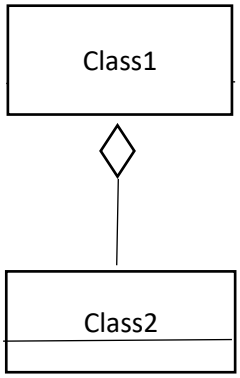
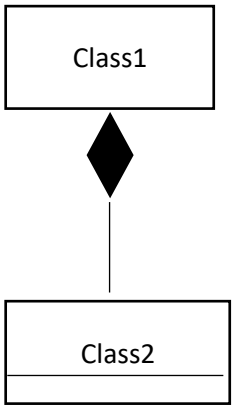
Associate Prof. Dr.V.Balasubramanian.

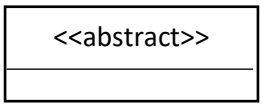
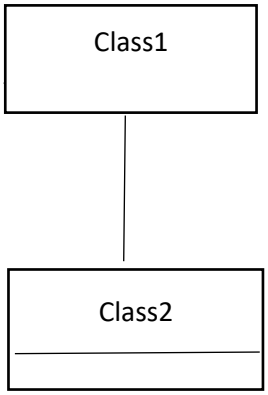
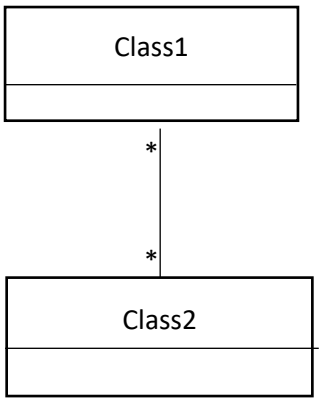
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SSN COLLEGE OF ENGINEERING

JANUARY 2021

UML Notations of Domain Model and Class Diagram:

S.NO	NAME	NOTATION	DESCRIPTION
1	Class		Class is an entity of class diagram. It describes a group of objects with same properties and behavior.
2	Object		An object is an instance of class
3	Generalization		Generalization organizes classes by their super-class and sub-class relationship.
4	Aggregation		Aggregation -one class is a part of another class
5	Composition		Composition is a Form of aggregation. Composition implies ownership of the parts by whole

6	Abstract class		Abstract class has no direct instances
7	Association		An association is a description of a Link (conceptual connection among objects)
8	Multiplicity		Multiplicity specifies the number of instances of one class that may relate to a single instance of an associated class

Aggregation:

Aggregation simply shows a relationship where one thing can exist independently of other thing. It means to create or compose different abstractions together in defining a class. Aggregation is represented as a part of relationship in class diagram. In diagram given below, we can see that aggregation is represented by an edge with a diamond end pointing towards superclass. The “Online Course Reservation System” is superclass that consists of various classes.

These classes are “Dashboard”, ”User”, ”Course” as shown in diagram. Further, for “Account” class, “User” is a superclass. All of these, share a relationship and these relationships are known as aggregate relationships.

Multiplicity:

Multiplicity means that number of elements of a class is associated with another class. These relations can be one-to-one, many-to-many, and many-to-one or one-to-many. For denoting one element we use **1**, for zero elements we use **0**, and for many elements we use *****. We can see in diagram; many students are associated with many courses denoted by ***** and this represents a **many-to-many** type of relationship. One user has only one account that is denoted by 1 and this represents a **one-to-one** type of relationship.
Many user accounts are associated with one Admin and this represents **many-to-one** or **one-to-many** type of relationship. All these relationships are shown in diagram.

Generalization:

A taxonomic relationship between a more general classifier and a more specific classifier. Each instance of the specific classifier is also an indirect instance of the general classifier. Thus, the specific classifier indirectly has features of the more general class. The user class with student and faculty is generalization.

Abstract Class:

Abstract classes abstract classes and operations can be shown either with an {abstract} tag (useful when sketching UML) or by italicizing the name. The faculty and student class in the Course Portal is abstract class.

A qualified association:

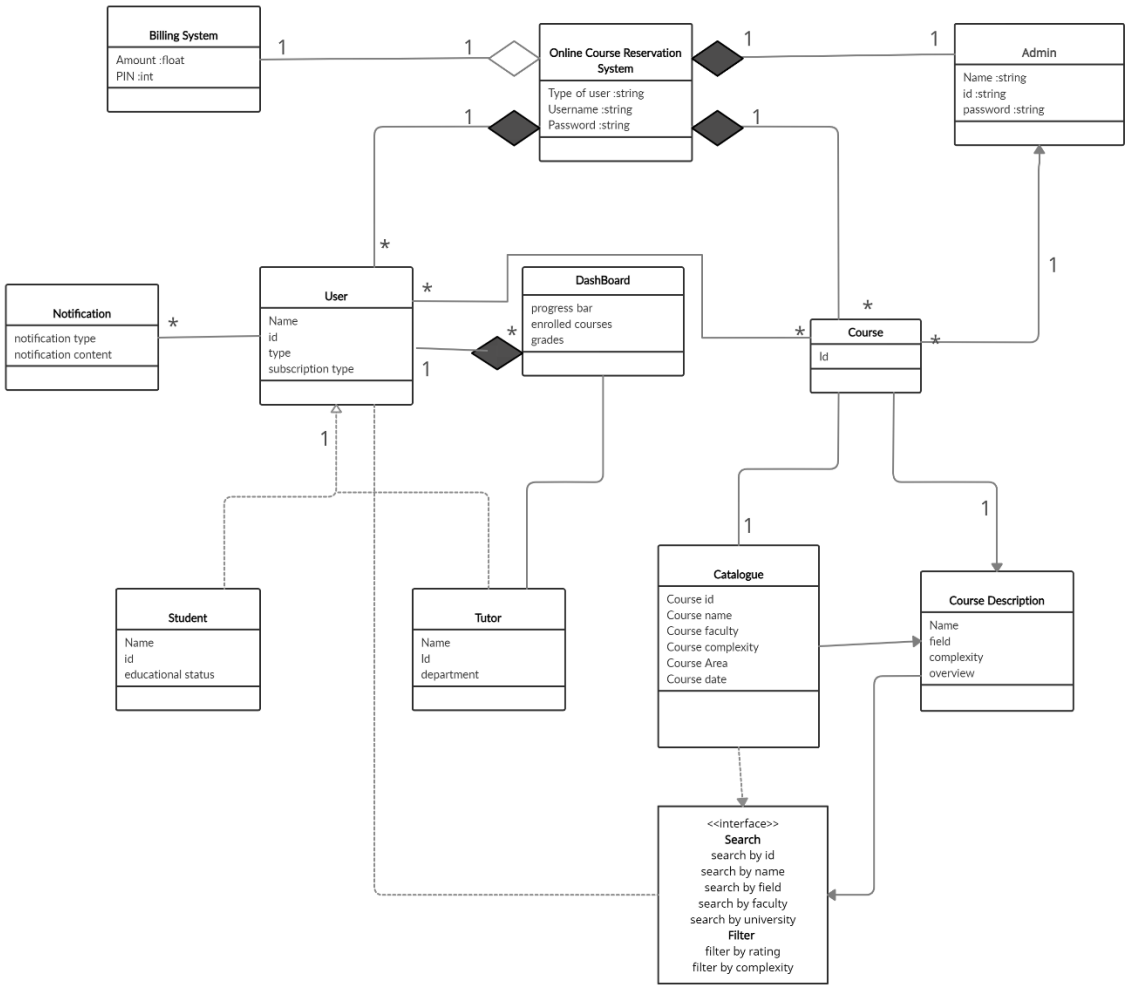
A qualified association has a qualifier that is used to select an object (or objects) from a larger set of related objects, based upon the qualifier key. The Course catalogue and its description has a qualified association.

Singleton class:

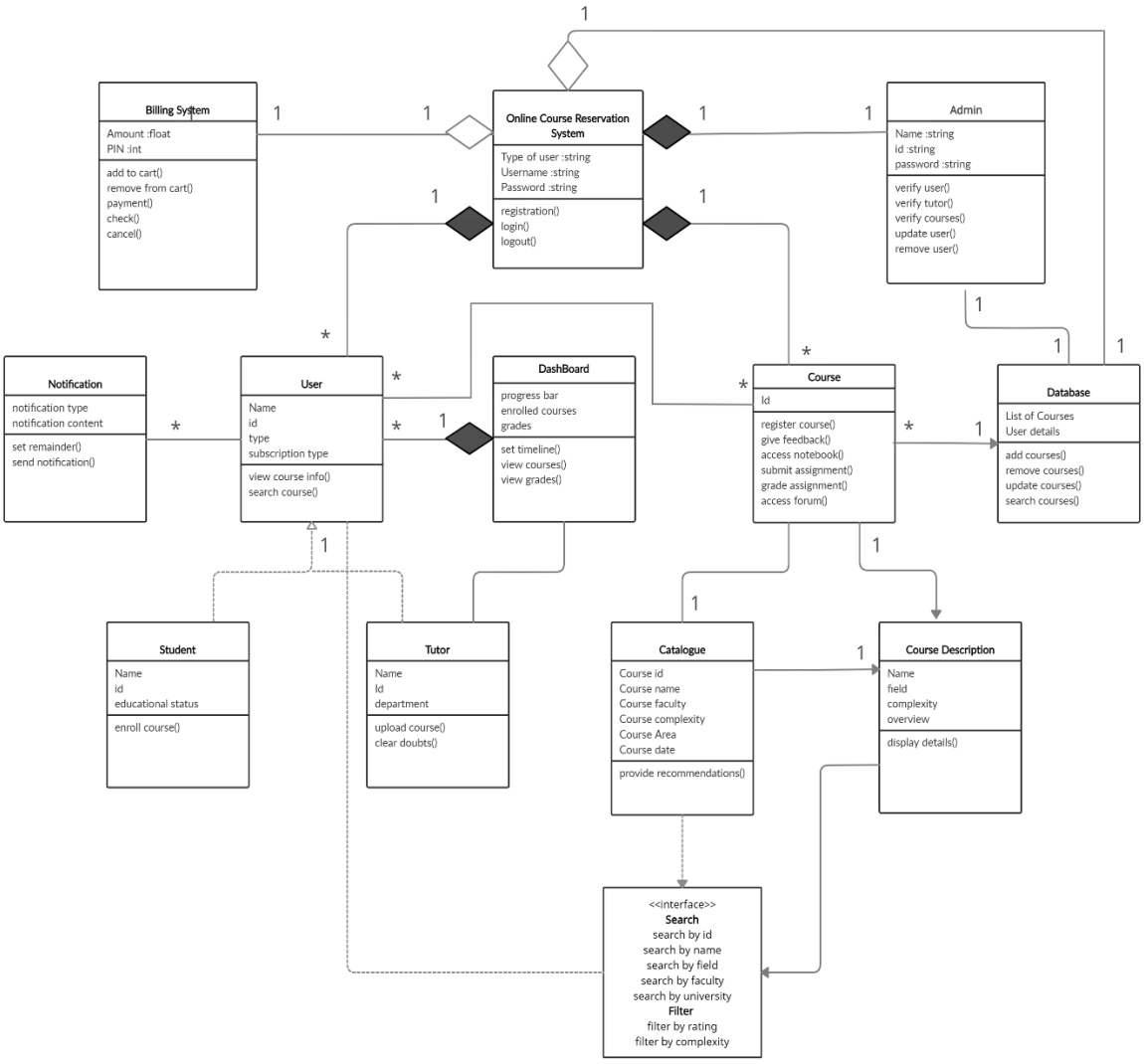
There is only one instance of a class instantiated—never two. In other words, it is a “singleton” instance. In a UML diagram, such a class can be marked with a ‘1’ in the upper right corner of the name compartment. The Admin is the singleton class.

Domain Model and Class diagram

Domain Model Diagram:



Class Diagram



Observation:

Domain Model and Use Case Diagrams decompose the problem domain into significant concepts or objects. Here we've broken down the Online Course Reservation into various classes and associated them with each other which provides quick overview of what's happening among the different system elements as well as their properties and relationships.