CodeClan

PDA Guidance

Part 2

Professional Development Award in Software Development lvl 8

Focus of Guidance

Part 2 will cover material specifically needed for the PDA evidence for weeks 8 onwards.

- Acceptance Criteria and test plan
- System Interaction Diagram
- Bug Tracking Report
- Inheritance Diagram

Acceptance Criteria & Test Plan

The acceptance criteria are a set of several prerequisites and conditions that must be satisfied for the product to be considered acceptable by clients and end users. They verify the software's development and ensure the product operates as intended, without any flaws or bugs.

The acceptance criteria should always be written before development begins. Writing these criteria should clarify "what" to expect rather than "how" to achieve or implement a specific functionality because the purpose of doing so is to state the aim, not the solution.

Acceptance Criteria & Test Plan

Acceptance Criteria	Expected Result	Pass/Fail
A user is able to…	The…Does…When…	Pass or Fail

System Interaction Diagram

From the term Interaction, it is clear that the diagram is used to describe some type of interactions among the different elements in the model. This interaction is a part of dynamic behavior of the system.

This interactive behavior is represented in UML by two diagrams known as Sequence diagram and Collaboration diagram. The basic purpose of both the diagrams are similar.

Sequence diagram emphasizes on time sequence of messages and collaboration diagram emphasizes on the structural organization of the objects that send and receive messages.

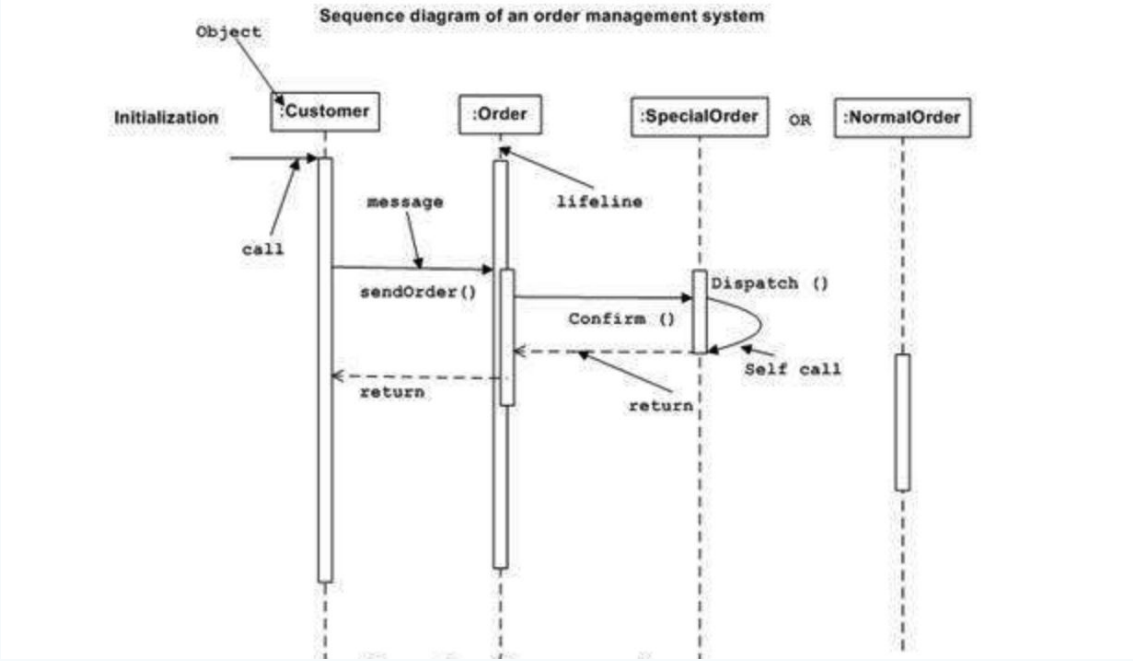
Reading Material for your own understanding -
https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-sequence-diagram/

Sequence Diagram

The following diagram shows the message sequence for SpecialOrder object and the same can be used in case of NormalOrder object. It is important to understand the time sequence of message flows. The message flow is nothing but a method call of an object.

The first call is sendOrder () which is a method of Order object. The next call is confirm () which is a method of SpecialOrder object and the last call is Dispatch () which is a method of SpecialOrder object. The following diagram mainly describes the method calls from one object to another, and this is also the actual scenario when the system is running.

Sequence Diagram

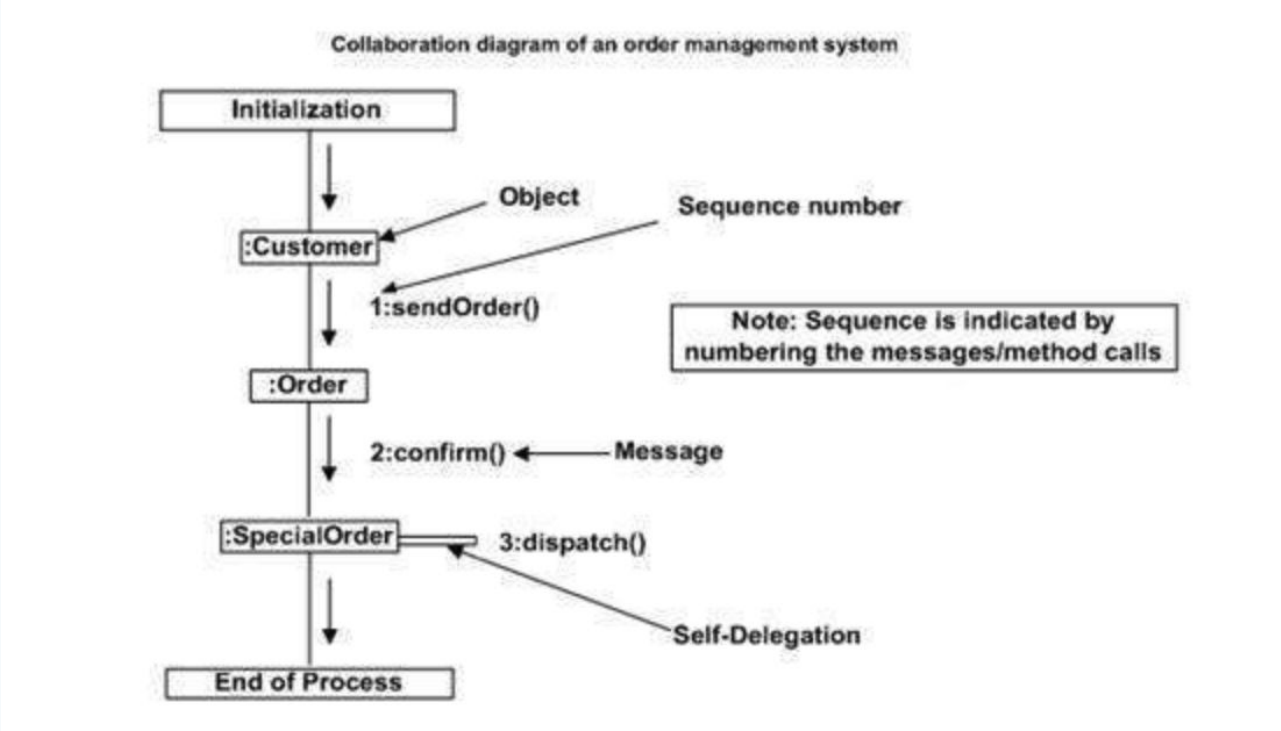


Sequence Diagram

Collaboration Diagram

The second interaction diagram is the collaboration diagram. It shows the object organization as seen in the following diagram. In the collaboration diagram, the method call sequence is indicated by some numbering technique. The number indicates how the methods are called one after another. We have taken the same order management system to describe the collaboration diagram.

Collaboration Diagram



PDA Guidance

Bug Tracking Report

A bug tracking report is a perfect solution to standardize the way your support, developers, and software tester teams manage to stay up to date on the latest reported issues, in order to address them effectively and comprehensively. To put it simpler, a bug report serves as a roadmap for your developers, so that they can acknowledge software issues and get to the bottom of it. These issues can vary from features not working properly to a broken functionality. Having a bug tracking report helps because it provides a procedure for your team, from the moment the issue is reported to the resolution of the problem.
You must include 5-6 examples for the PDA.

Bug Tracking Report

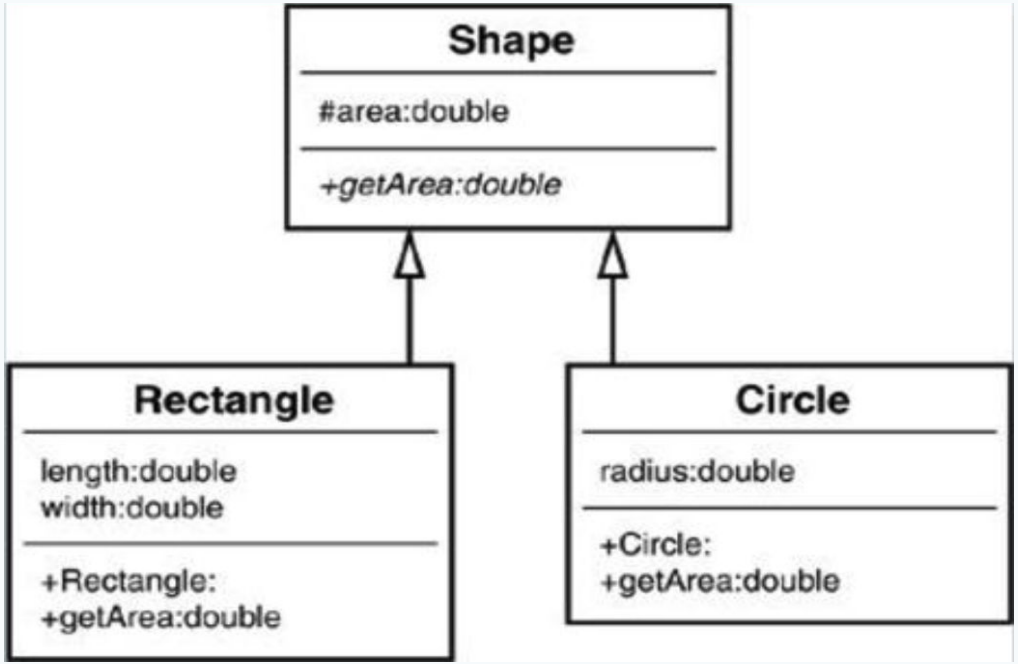
Date	Bug/Error	Solution

Inheritance Diagram

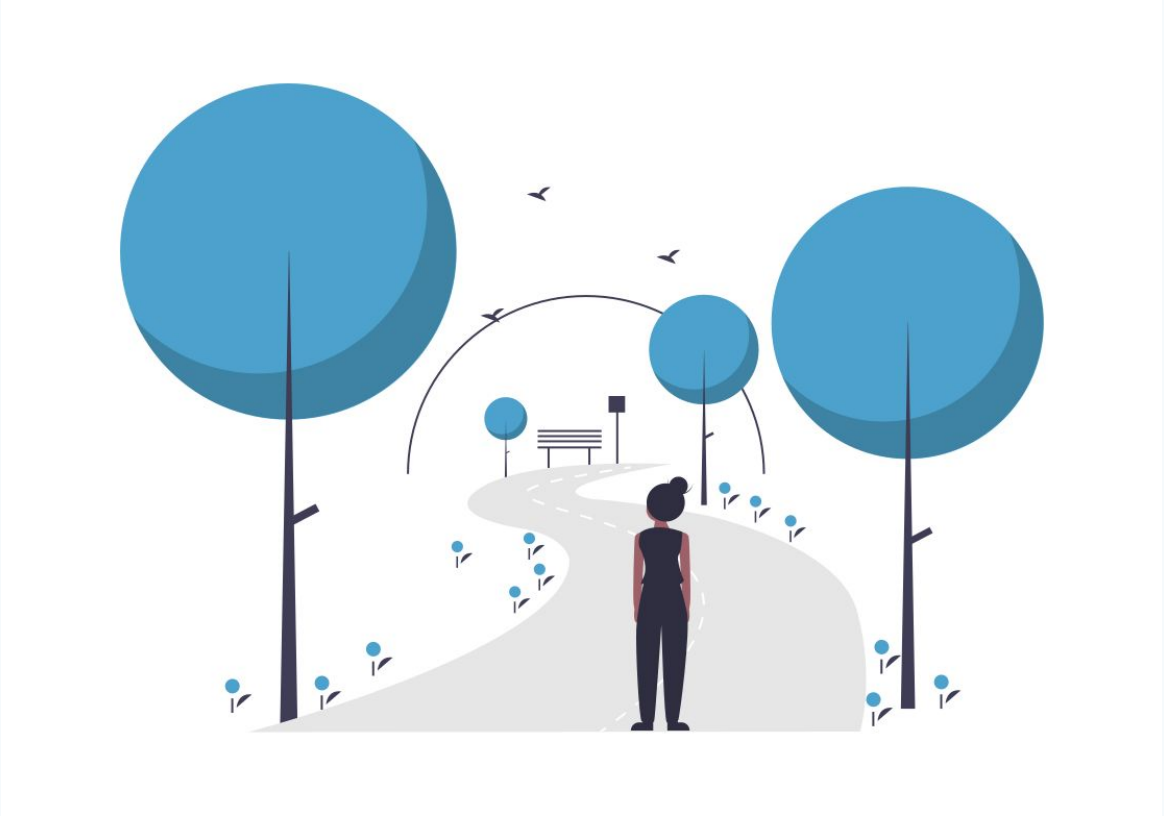
Inheritance is a central concept of the Unified Modeling Language (UML) and is used in the design of classes and the modeling of class diagrams. A class consists of a collection of attributes and methods that determine the state and behavior of its instances. Using inheritance, attributes and/or methods of one class are passed on to another class. The inheriting class is called the base class – alternatively also as super class or parent class – and the inheriting class as derived class – or as child class or subclass.

The relationship between the base class and the derived class is permanent.

Inheritance Diagram



Inheritance Diagram



CodeClan

End