**UML Class Notation.**

Lines or arrows between classes indicate relationships

Association

A relationship between instances of two classes, where one class must know about the other to do its work, e.g. client communicates to server

indicated by a straight line or arrow

Aggregation

An association where one class belongs to a collection, e.g. instructor part of Faculty

Indicated by an empty diamond on the side of the collection

Composition

Strong form of Aggregation

Lifetime control; components cannot exist without the aggregate

Indicated by a solid diamond on the side of the collection

Inheritance

An inheritance link indicating one class a superclass relationship, e.g. bird is part of mammal

Indicated by triangle pointing to superclass

A <<includes>> B

Somewhere through use case A, use case B is called. When B finishes, A carries on from where it left off.

A <<extends>> B

All the steps from use case A are performed during the execution of use case B, at the extension point which is specified within B.

A <<precedes>> B

Use case A must take place in its entirety before use case B even begins.

A <<invokes>> B

Use case B happens during the lifespan of use case A.

Name: Purchase ticket

Participating actor: Passenger

Entry condition:

Passenger standing in front of ticket distributor.

Passenger has sufficient money to purchase ticket.

Exit condition:

Passenger has ticket.

Event flow:

1. Passenger selects the number of zones to be traveled.

2. Distributor displays the amount due.

3. Passenger inserts money, of at least the amount due.

4. Distributor returns change.

5. Distributor issues ticket.

A use case represents a class of functionality provided by the system as an event flow.

A use case consists of:

Unique name

Participating actors

Entry conditions

Flow of events

Exit conditions

Special requirements

An actor models an external entity which communicates with the system:

User

External system

Physical environment

An actor has a unique name and an optional description.

Examples:

Passenger: A person in the train

GPS satellite: Provides the system with GPS coordinates

Used during requirements elicitation to represent external behavior

Actors represent roles, that is, a type of user of the system

Use cases represent a sequence of interaction for a type of functionality; summary of scenarios

The use case model is the set of all use cases. It is a complete description of the functionality of the system and its environment

STATIC

Use case diagram

Class Diagram

Dynamic

Object Diagram

State Diagram

Activity Diagram

Sequence Diagram

collaboration Diagram

IMPLEMENTATION

Component Diagram

Deployment Diagram

Use Case Diagrams

Class Diagrams

Package Diagrams

Interaction Diagrams

Sequence

Collaboration

Activity Diagrams

State Transition Diagrams

Deployment Diagrams

A model is an abstraction describing a subset of a system

A view depicts selected aspects of a model

A notation is a set of graphical or textual rules for depicting views

Views and models of a single system may overlap each other

Examples:

System: Aircraft

Models: Flight simulator, scale model

Views: All blueprints, electrical wiring, fuel system

Inundated with methodologies in early 90’s

Booch, Jacobson, Yourden, Rumbaugh

Booch, Jacobson merged methods 1994

Rumbaugh joined 1995

1997 UML 1.1 from OMG includes input from others, e.g. Yourden

UML v2.0 current version

Open Standard, Graphical notation for

Specifying, visualizing, constructing, and documenting software systems

Language can be used from general initial design to very specific detailed design across the entire software development lifecycle

Increase understanding/communication of product to customers and developers

Support for diverse application areas

Support for UML in many software packages today (e.g. Rational, plugins for popular IDE’s like NetBeans, Eclipse)

Based upon experience and needs of the user community

Unified Modeling Language

OMG Standard, Object Management Group

Based on work from Booch, Rumbaugh, Jacobson

UML is a modeling language to express and design documents, software

Particularly useful for OO design

Not a process, but some have been proposed using UML

Independent of implementation language

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 26/9/2023 UML \*\*\*\*\*\*\*\*