

# Software Systems Development - ICS 2312: Answer Guide

## Answer Guide for ICS 2312 - Software Systems Development

### QUESTION 1

#### a) i) Definitions:

- Object: An object is an instance of a class that encapsulates data and behavior. It represents a real-world entity with attributes (data) and methods (functions).
- Use Case: A use case describes a system's functional requirement by identifying a user's interaction with the system to achieve a specific goal.

#### ii) Non-functional requirements categories:

1. Performance - Example: System response time must be less than 2 seconds.
2. Security - Example: Users must log in with a username and password.
3. Usability - Example: The interface must be intuitive and require minimal training.

#### b) Purpose of use case modeling:

Use case modeling helps capture functional requirements of a system from the user's perspective. It is typically used during the requirements analysis phase. Outputs include use case diagrams and textual use case descriptions.

#### ii) Abstract vs Concrete classes:

- Abstract classes: Used when a base class is needed that should not be instantiated directly. Suitable when common behavior is shared among multiple derived classes.
- Concrete classes: Fully implemented and can be instantiated. Suitable when the class can stand alone and

## Software Systems Development - ICS 2312: Answer Guide

perform specific operations.

c) Responsibilities and freedoms:

- Specify: Identify what the system must do (requirements). Freedom: Define scope.
- Prototype: Create a working model. Freedom: Explore designs quickly.
- Design: Plan the architecture and components. Freedom: Choose implementation strategies.
- Deploy: Release the system. Responsibility: Ensure stability and operability.

d) State transition diagram (text description):

States: Waiting -> Ready -> Dispense -> Out of Stock

Transitions:

- Insert coins -> Ready
- OK button -> Dispense
- If item dispensed and stock available -> Waiting
- If out of stock -> Out of Stock

e) Classes and class diagram (text description):

- Student: studentNumber, name
- Course: courseName, duration, type
- Unit: unitName
- Lecturer: name (coordinates one course)

Relationships:

- Student selects/registers for units.
- Each student enrolled in one course.
- Lecturers assigned to teach units.

## Software Systems Development - ICS 2312: Answer Guide

### QUESTION 2

- a) i) Software artifact: Any product produced during software development (e.g., code, design documents).
- ii) Step post condition: The expected outcome/state after a use case step is executed. Example: "User is logged in".
- b) Operation in class: An operation is a function or service provided by a class. It is made up of name, parameters, and return type.
- c) i) Abstract use case: A use case that represents common behavior among several use cases. Cannot be executed alone.
- ii) Super state: A state in a state machine that encompasses multiple substates. Allows reuse of common transitions.
- d) Class diagram (Air transportation system - text description):
- Classes:
- Flight: flightNumber, date
  - Airline: name
  - Plane: serialNumber, model
  - Pilot: name
  - Passenger: name
- Relationships:
- Flight belongs to an Airline
  - Flight uses a Plane

## Software Systems Development - ICS 2312: Answer Guide

- Pilot assigned to Flight
- Passenger reserves Flight

### QUESTION 3

a) Requirements traceability: Ability to link requirements throughout development lifecycle.

Reasons:

1. Ensures all requirements are implemented.
2. Facilitates impact analysis of changes.
3. Improves project tracking and documentation.

b) Pervasive step: A process that occurs throughout the lifecycle.

Examples: Configuration management, Quality assurance.

c) Class diagram (Retail Establishment - text description):

Classes:

- Product: name, price, quantity
- Order: orderID, date
- Customer: name, contact
- Supplier: name, contact

Relationships:

- Customer places Order
- Order contains Products
- Products supplied by Supplier