**Software Development Processes and Methodologies**

Week 1

**Practice Quiz:** What software development looks like

### Question 1 Which of the following software development models can best respond to requirements changes?

1. The V-model
2. The Waterfall model
3. Agile models

Correct (Agile models)

Question 2 In which of the following software development models are the software development activities performed sequentially rather than in iterations?

1. The Waterfall model
2. Agile models

Correct(The Waterfall model)

Question 3 Which of the following are limitations of the waterfall model? Select three.

1. It is not suitable for big projects.
2. Misinterpretations of requirements or design can remain undetected until the later development phases.
3. Integration issues may remain undetected until the last phase.
4. It is difficult to respond to requirements changes.

Correct

1. Misinterpretations of requirements or design can remain undetected until the later development phases.
2. Integration issues may remain undetected until the last phase.
3. It is difficult to respond to requirements changes

# **Practice Quiz:** Why do we need requirements?

Question 1 Which is the right description about the requirement specification process?

1. Requirement specification is a process to identify and specify the problem to solve.
2. Requirement specification is a process that identifies and specifies the problem and possible solutions to the problem.
3. Requirement specification is a process of solving the requirements.
4. Requirement specification is a process to design the solution to the problem.

Correct (Requirement specification is a process to identify and specify the problem to solve.)

Question 2 Specifying requirements is difficult because (select three):

1. The client or end users might not be clear about what they want.
2. Terminology can be interpreted in multiple ways depending on the person or the context in which it was used.
3. There is no guideline for writing a software requirement specification document.
4. Software is intangible, which makes it difficult to comprehend and communicate.

Correct

1. The client or end users might not be clear about what they want.
2. Terminology can be interpreted in multiple ways depending on the person or the context in which it was used.
3. Software is intangible, which makes it difficult to comprehend and communicate

Question 3 Why is requirement specification important? Select two.

1. Spending time upfront in requirement specification can save time in the later phases of the software development lifecycle.
2. Changing the requirement is not possible once the system is fully developed.
3. Repairing a bug in the requirements can cost thousands of times less than in the later phases of the software development lifecycle.

Correct

1. Spending time upfront in requirement specification can save time in the later phases of the software development lifecycle.
2. Repairing a bug in the requirements can cost thousands of times less than in the later phases of the software development lifecycle.

# **Practice Quiz: WRSPM Quiz**

Question 1 WRSPM stands for:

1. Wide, Requirement, Software, Planning and Model.
2. World, Requirement, Specification, Program and Machine.
3. None of the above.

Correct (World, Requirement, Specification, Program and Machine.)

Question 2 Looking at the difference between user requirements and system specifications in the ATM example, we know that swiping the card and prompting for a PIN are requirements, while reading the card details and a 4-digit PIN are specifications.

1. True.
2. False.

Correct (True! )

Question 3 The purpose of the WRSPM model is to ensure that:

1. Specifications meet the requirements.
2. Requirements meet specifications.
3. None of the above.

Correct (Specifications meet the requirements.)

# **Practice Quiz:**

# Software Architecture: Definition

Question 1 A subsystem in an architecture must:

1. be created separately and can operate individually.
2. have business value.
3. be integrated with one another or with existing subsystems.
4. All of the above.

Correct (All of the above. )

Question 2 Partitioning of a large system into smaller subsystems helps the buy-or-build decision because we can examine each subsystem and reason about possible buy-or-build options for each.

1. True.
2. False.

Correct (true)

A good software architecture is important because:

1. It helps organize the workforce and resources.
2. It allows for parallelization in development.
3. It helps build-or-buy decisions.
4. It helps with funding decisions.

Correct

1. It helps organize the workforce and resources.
2. It allows for parallelization in development.
3. It helps build-or-buy decisions.
4. It helps with funding decisions.

**Practice Quiz:**

# Software Architecture Models

Question 1) A UNIX program where the output of one program is the input of another, is an example of which of the software architecture models below:

1. Blackboard Model.
2. Pipe-and-Filter Model.
3. Client-Server Model.
4. Event-based Model.
5. Layered Model.

Correct (Pipe-and-Filter Model.)

Question 2) An online banking system is best modeled by:

1. Pipe-and-Filter Model.
2. Blackboard Model.
3. Client-Server Model.
4. Event-based Model.

Correct (Client-Server Model.)

Question 3) A vehicle identification and tracking system, where each moving vehicle is tracked and monitored through a shared program, is best modeled by:

1. Pipe-and-Filter Model.
2. Blackboard Model.
3. Layer Model
4. Client-Server Model
5. Event-based Model

Correct (Blackboard Model.)

**Software Architecture: Process**

Question 1) The difference between subsystems and modules are:

1. Subsystems can independently comprise the business logic by itself while modules can't.
2. Subsystems can communicate with other subsystems while modules cannot communicate with other modules.
3. All of the above.

Correct (Subsystems can independently comprise the business logic by itself while modules can't)

Question 2 Software quality attributes that we care about during software architecture are:

1. Performance.
2. Reliability.
3. Testability.
4. Security.
5. Usability.

Correct

1. Performance.
2. Reliability.
3. Testability.
4. Security.
5. Usability.

Question 3 Software architecture concerns itself with both estimation and quality but not partitioning.

1. True.
2. False.

Correct (False.)

**Requirements and Architecture**

Question 1) Which of the following is considered the most difficult in the software development process?

1. Determining the accurate market price for the finished product
2. Developing a Java program to meet the designed solution
3. Testing the solution with users
4. Accurately identifying user needs

Correct (Accurately identifying user needs)

Question 2) What is a Requirements Specification?

1. A process, how the requirements are written (specified)
2. A product, a written specification of the requirements
3. Both
4. Neither

Correct (Both)

Question 3) Based on the NASA statistics on budget and schedule overrun vs. time spent on requirements process, what is the recommended amount of time to spend on the requirements stage?

1. 0% of the total time spent on the project
2. 5-10% of the total time spent on the project
3. 20% of the total time spent on the project
4. The same amount as you expect to spend on testing.

Correct (5-10% of the total time spent on the project)

### Question 4) Which of the following matches the terms to the correct definition?

1. Requirements - solution properties designed to solve problem; Specification - user needs in user language
2. Requirements - user needs in user language; Specification - solution properties designed to solve problem

Correct (Requirements - user needs in user language; Specification - solution properties designed to solve problem)

Question 5) Which of these is a user requirement?

1. Allow an authorized user to post a message of no more than 136 characters to the Facebook POST API
2. Use the MariaDB database for internal data persistance
3. Retry posting the message up to 3 times every 5 minutes if the post is rejected by the server
4. Allow the user to post a message to Facebook

Correct (Allow the user to post a message to Facebook)

Question 6) Which of the following is NOT an example of a non-functional requirement.

1. The messages between the client and server shall be in YAML 1.2 format.
2. The developing team shall use the spiral model of software development lifecycle.
3. Communication between the client and server will be expressed in EBCDIC endoding.
4. The user shall be able to select the course number.

Correct (The user shall be able to select the course number.)

Question 7)What element(s) of the WRSPM model belong in the interface (i.e. the overlap between environment and system)?

1. M
2. R,S
3. P,M
4. S
5. S,P
6. R,S,P
7. W,R,S
8. W,R
9. S,P,M
10. W

Correct (S)

Question 8) Which of these is proper definition of software architecture?

1. integrating small systems with no individual business value into larger ones
2. partitioning large systems into smaller ones that can be created separately, have individual business value, and can be easily integrated
3. planning and pricing the resources involved with developing a large-scale software system, including presenting such plans and budgets to senior level executives for funding
4. all of the above

Correct (partitioning large systems into smaller ones that can be created separately, have individual business value, and can be easily integrated)

Question 9) Which of the following models is best suited for a system with significant shared data that needs to be shared across a variety of components or sub-systems, somewhat like global variables, but with better data integrity?

1. layered
2. event-based
3. client-server
4. pipe-and-filter
5. blackboard

Correct (blackboard)

Question 10) Which of the following models is best suited for a system that includes several subsets of functionality that are used in more than one area of the system?

1. client-server
2. event-based
3. blackboard
4. pipe-and-filter
5. layered

Correct (pipe-and-filter)

Question 11) Which of the following models is best suited for a system that benefits from the separation of complexity and processing between work which can be done locally and that which should be completed on a shared, remote service?

1. layered
2. client-server
3. event-based
4. pipe-and-filter
5. blackboard

Correct (client-server)

Question 12) Which of the following would be considered a sub-system, rather than a module?

1. A set of classes which convert various values based on environment or parameter information
2. A class which provides a service (e.g. AccountCreationService)
3. A class which represents the paying customer
4. A set of classes which generates reports

Correct (A set of classes which generates reports)

# Software Design: Introduction

Question 1) Software design is the process of transforming the stated problem into a ready-to-use implementation.

1. False
2. True
3. Correct (False)

Question 2) Abstract solutions do not require extensive domain knowledge and effectively reduce the costs during the software design phase.

1. False
2. True

Correct (False)

Question 3) It is often advised that abstract solutions do not provide optimization details regarding the implementation.

1. True
2. False

Correct (True)

Question 4) When it comes to software design, it is always best to follow a solution that is widely popular in the industry

1. False
2. True

Correct (False)

Question 5) While a solution coming from software design does not include implementation details, there are still common cases where pseudocode may be provided to correctly capture the sense of a complex algorithm.

1. True
2. False

Correct (True)

# Software Design: Modularity

Question 1) Check all that apply: The four aspects of modularity are...

1. Coupling
2. Components
3. Sort Functions
4. Data Encapsulation
5. Cohesion
6. Information Hiding
7. Data Hiding

Correct

1. Coupling
2. Data Encapsulation
3. Cohesion
4. Information Hiding

Question 2) Which of the four aspects of modularity is defined as: How well modules work together.

1. Coupling
2. Data Encapsulation
3. Cohesion
4. Information Hiding

Correct (Coupling)

Question 3) Which of the four aspects of modularity can be described as: Abstracting away implementation details.

1. Information Hiding
2. Cohesion
3. Data Encapsulation
4. Coupling

Correct (Information Hiding)

### Question 4) Which of the four aspects of modularity can be described as: How well a module meets a single well-defined goal.

1. Data Encapsulation
2. Coupling
3. Information Hiding
4. Cohesion

Correct (Cohesion)

Question 5) Which of the four aspects of modularity can be described as: Containment of constructs and concepts within a module.

1. Coupling
2. Cohesion
3. Information Hiding
4. Data Encapsulation

Correct (Data Encapsulation)

Question 6) Three goals of \_\_\_\_\_\_ can be described as (1) Decomposability, (2) Composability, and (3) Ease of Understanding.

1. Cohesion
2. Modularity
3. Coupling

Correct(Modularity)

Question 7)You have a sort function that provides no details on which sorting algorithm is used. This is an example of which aspect of modularity?

1. Cohesion
2. Information Hiding
3. Coupling
4. Data Encapsulation

Correct (Information Hiding)

### Question 8) A benefit to using \_\_\_\_\_\_ is that you know if your data is corrupted, then it must have been corrupted by the module.

1. Cohesion
2. Information Hiding
3. Coupling
4. Data Encapsulation

Correct (Data Encapsulation)