CONFIDENTIAL

UTM UNIVERSITI TEKNOLOGI MALAYSIA | Faculty of Computing

FINAL EXAMINATION
SEMESTER I, 2024/2025

SUBJECT CODE	:	**SECJ2203**
SUBJECT NAME	:	**SOFTWARE ENGINEERING**
SECTION	:	
TIME / DURATION	:	**3 HOURS**
DATE/DAY	:	
VENUE	:	

INSTRUCTIONS:

This test book consists of **TWO (2)** sections:

Section A: Multiple Choice Questions	50 Marks
Section B: Case Study	50 Marks
TOTAL	**100 Marks**

ANSWER ALL QUESTIONS IN THE ANSWER BOOKLET.

The question test book must be returned at the end of the test.

*This question paper consists of **11** printed pages excluding this page.*

SECTION A: MULTIPLE CHOICE QUESTIONS **[50 Marks]**

Instruction: *Read each question below carefully and choose the letter (A, B, C or D) that* ***BEST*** *describes the answer. Write your answer in the answer booklet. Each question carries* ***2 marks****.*

1. What is the primary purpose of system modeling?

   A. To create user interfaces.
   B. To develop abstract models of a system.
   C. To build hardware prototypes.
   D. To code software functionality.

2. What is a key feature of structural models?

   A. They depict dynamic system behavior.
   B. They represent the organization of system components.
   C. They focus on interaction between actors and systems.
   D. They show transitions between system states.

3. Which UML diagram would be the best to represent the interactions between users and the system?

   A. Class Diagram
   B. Activity Diagram
   C. Use Case Diagram
   D. Sequence Diagram

4. In a hospital management system, a "Patient" can have multiple appointments, and each appointment is linked to a single doctor. What multiplicity would you use to show the relationship between patients and appointments?

   A. 1:1
   B. 1: *
   C. *:1
   D. *:*

5. A traffic management system controls signals and monitors vehicles.

   Why a state diagram is useful to show the state for traffic signals?

   A. It shows interactions with vehicles.
   B. It models transitions between signal states like "Red" to "Green."
   C. It defines the relationship between signals and roads.
   D. It captures the process flow for signal operation.

6. Why modeling user interaction is crucial in software engineering?

   A. It reduces software cost significantly.
   B. It helps identify user requirements.
   C. It ensures faster software deployment.
   D. It eliminates the need for testing.

7. Which of the following is false about the architectural design?

   A. It is concerned with understanding how a software system should be organized and designing the overall structure of that system.
   B. It identifies the main structural components in a system and the relationships between them.
   C. It is the critical link between data collection and requirements engineering.
   D. The output of the architectural design process is an architectural model that describes how the system is organized as a set of communicating components.

8. Which of the following is not an advantage of explicit architecture?

   A. It can be used as a focus of discussion by system stakeholders.
   B. It can be used to describe the details of the system components.
   C. It can be used in system analysis to determine whether the system can meet its non-functional requirements.
   D. It may be used in large scale reuse to determine whether the architecture will be reusable across a range of systems.

9. Which of the following best describes the process view in the 4 + 1 view model of software architecture?

   A. Key abstractions in the system as objects or object classes.
   B. The decomposition of software for development.
   C. The system hardware and distribution of software components across processors.
   D. The composition of interacting processes at run-time.

10. Which of the following UML Diagram best represents the implementation view in the 4 + 1 view model of software architecture?

    A. Class diagram and object diagram.
    B. Deployment diagram.
    C. Component diagram.
    D. Use case diagram.

11. Which of the following is the best architectural patterns to be used in data processing applications where inputs are processed in separate stages to generate related outputs?

    A. Client-Server Pattern.
    B. Repository Pattern.
    C. Pipe and Filter Pattern.
    D. Layered Architecture.

12. Which of the following best describes the Model-View-Controller (MVC) Pattern?

    A. It organizes the system into layers with related functionality associated with each layer.
    B. It separates presentation and interaction from the system data. The system is structured into three logical components that interact with each other.
    C. All data in a system is managed in a central repository that is accessible to all system components.
    D. The functionality of the system is organized into services, with each service delivered from a separate server. Clients are users of these services and access servers to make use of them.

13. Which of the following best describes high cohesion in a subsystem?

A. The classes perform similar tasks and are related to each other via associations.
B. The classes perform a wide range of unrelated tasks.
C. The subsystem depends on many other subsystems.
D. The subsystem has very few classes with no meaningful relationships.

14. What does "coupling" measure in the context of system design?

A. The number of classes in a subsystem.
B. The dependency between different subsystems.
C. The number of associations within a class.
D. The time it takes for a system to process data.

15. Which of the following is false in refining steps for use case realization?

A. Step 1: Add <<boundary>> classes in sequence diagram for View Layer
B. Step 2: Add <<control>> class in sequence diagram for Business Layer
C. Step 1: Add <<dataAccess>> class in sequence diagram for Data Access Layer
D. Step 3: Add <<dataAccess>> class in sequence diagram for Data Access Layer

16. What is the purpose of information hiding in object-oriented design?

A. To make the object’s data accessible to all other objects.
B. To restrict access to the object’s internal data.
C. To hide the object's methods.
D. To allow objects to directly modify each other’s data.

17. Why object reuse is important in software development?

A. It reduces the complexity of new systems by using previously developed objects.
B. It increases the overall cost of development.
C. It limits the scope of the project.
D. It eliminates the need for encapsulation.

18. Which of the following describes low coupling in a well-formed design class?

    A. The class is highly dependent on other classes, causing changes in one class to affect others.
    B. The class has minimal interaction with other classes, meaning changes to one class will not impact others.
    C. The class has high cohesion, but is tightly coupled with external components.
    D. The class only interacts with classes that perform similar tasks.

19. Which of the following correctly defines Verification and Validation in software engineering?

    A. Verification ensures that the software product meets the user's expectations, while Validation ensures the software is free of bugs.
    B. Verification ensures the software is built according to its specifications, while Validation ensures the software fulfills its intended use.
    C. Verification is the process of testing the software manually, while Validation is the process of testing the software automatically.
    D. Verification and Validation are interchangeable terms referring to testing methods in software engineering.

20. Which of the following is an example of static testing in software engineering?

    A. Executing test cases to identify runtime errors.
    B. Reviewing the code to identify potential defects without executing it.
    C. Simulating user interactions to validate system functionality.
    D. Running automated scripts to test software performance.

21. What is the purpose of using a prototype during verification testing in software development?

    A. To validate that the final software meets the needs of the end-users.
    B. To check whether the software design aligns with specified requirements.
    C. To execute functional tests and identify runtime defects.
    D. To assess the performance of the software under load conditions.

22. Which of the following describes the purpose of model analysis when cross-checking UML models?

    A. To ensure that the UML diagrams accurately represent the software's runtime behavior.
    B. To verify consistency and correctness between different UML diagrams, such as class and sequence diagrams.
    C. To simulate the UML diagrams for testing the software's performance under various conditions.
    D. To validate user requirements against the implemented UML models.

23. Which of the following are the four stages of testing typically used in software development?

    A. Unit Testing, System Testing, Performance Testing, Acceptance Testing
    B. Component Testing, Integration Testing, System Testing, Acceptance Testing
    C. Integration Testing, Regression Testing, System Testing, Security Testing
    D. Component Testing, User Testing, Stress Testing, Acceptance Testing

24. During Component Testing in software development, which of the following are examples of components that may be tested?

    A. Individual functions, classes, or modules within the software.
    B. The entire software system as a whole.
    C. The interaction between different software applications.
    D. The user interface tested by end-users.

25. Which of the following is not an example of system testing used to test non-functional attributes of a software system?

    A. Stress Testing
    B. Performance Testing
    C. Usability Testing
    D. Unit Testing

SECTION B: CASE STUDY **[50 Marks]**

ABC is a new company looking to create a smart and efficient food delivery platform. In today's fast-paced world, people want quick and reliable ways to order food online. ABC's goal is to build a system that makes it easy for customers to order food, restaurants to manage their menus and orders, and delivery drivers to complete their deliveries smoothly. To succeed, this platform must also be easy to use, reliable, and secure.

The Food Delivery System (FDS) will include four main modules. The customer module will let users create accounts, browse restaurant menus, place orders, and track deliveries in real time. The restaurant partner module will allow restaurants to update their menus, manage food availability, and handle incoming orders. The delivery personnel module will help drivers see their delivery tasks, update order statuses, and track their earnings. Lastly, the admin module will be used by the company's administrators to monitor the entire system and manage user accounts and performance.

Apart from the main features, the system also needs to handle some technical challenges to work well. It must be fast enough to manage thousands of orders during busy times, always available with minimal downtime, and secure to protect customer and payment information. It should also be simple to use so that anyone, no matter their technical skill, can navigate the platform easily. Additionally, it needs to be designed in a way that makes it easy to update and expand in the future.

There are some unique challenges that ABC must solve. For example, it needs to provide live updates to customers about their orders without slowing down the system. It also must allow the system to grow over time, supporting more users and features as ABC expands to new cities. These goals mean the system's design must be flexible and well-organized from the start.

Question 1 **[13 marks]**

Table 1 presents the normal flow for UC001 Customer Place Order.

Table 1: Normal Flow for UC001 Customer Place Order

Normal Flow (NF)	1. Customer view menus. 2. The customer adds desired food items to the cart. 3. The customer views the cart to see the selected food items. 4. The customer submits the order. 5. The system calculates the total price, including applicable taxes 6. The use case ends.

(a) Based on Table 1: UCS of UC001 Customer Place Order, assign ONE (1) class as <<boundary>>, assign ONE (1) class as <<controller>> and assign THREE (3) classes as <<entity>> or <<data access>>. Follow Table 2 to write your answer in the answer booklet. (5 marks)

Table 2: Class Name

Stereotype	Class Name
<<boundary>>	
<<controller>>	
<<entity>>	
…	
…	

(b) Complete the sequence diagram in Figure 1, to show the interaction for UC001 Customer Place Order, based on the identified classes in (a). (Note: Redraw the use case diagram in the answer booklet) (3 marks)

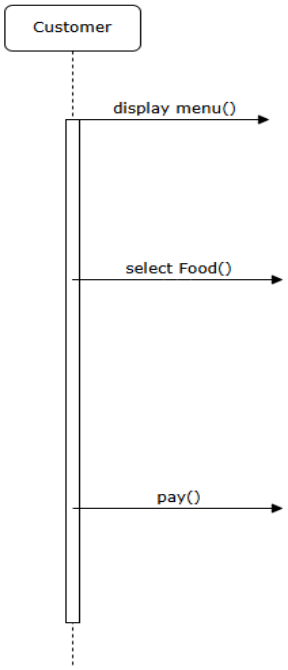

Figure 1: Sequence Diagram for Customer Place Order

(c) Based on the transition shown in Table 3, complete the state diagram for Order Class in Figure 2 with a suitable transition. (Note: Redraw the state diagram in the answer booklet).
(3 marks)

Table 3: Transition for Order Class

Transition		
Event	**Condition**	**Action**
saveOrder()	ID generated	save the order ID
orderIsConfirm()	-	calculate payment
selectMenu()	when(food>0)	add food to the order

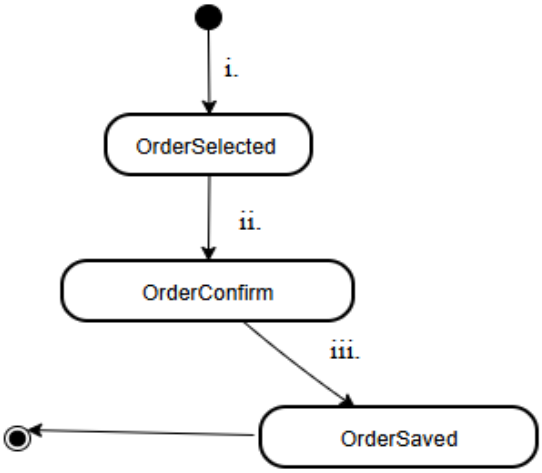

Figure 2: State Diagram for Order Class

(d) Identify TWO (2) non-functional requirements suitable for FDS.
(2 marks)

Question 2 **[12 marks]**

(a) Suggest a suitable architectural pattern to design the structure of the FDS. (1 mark)

(b) Explain TWO (2) reasons why your choice is the best fit architectural pattern for FDS.
(4 marks)

(c) Describe the design of the FDS components based on the suggestion in (b).

(3 marks)

(d) Draw the package diagram to visualize the design of FDS subsystems based on (c).

(4 marks)

Question 3 **[13 marks]**

(a) Identify FOUR (4) classes involved in the "Customer Module" of the FDS. (2 marks)

(b) Based on the principles of cohesion and coupling, explain how these classes identified in (a) can be organized to ensure a well-formed design. (2 marks)

(c) Draw a design class diagram for the "Delivery Personnel Module" in the FDS. The design should include at least THREE (3) classes and demonstrate appropriate relationships and navigability. (5 marks)

(d) Propose TWO (2) object-oriented design principles that can be used to enhance the security of customer and payment information in the FDS. (4 marks)

Question 4 **[12 marks]**

(a) Generate ONE (1) test requirement for the following user modules:

i. Customer modules for place order. (1 mark)
ii. Restaurant partner module for manage food availability. (1 mark)
iii. Delivery personnel module for update order statuses. (1 mark)

(b) Based on the test requirement generated in (a) for restaurant partner module for manage food availability, design ONE (1) test case that covers normal scenario and ONE (1) test case that covers error scenario. Normal scenario is that user input positive integer number for the food amount and the food amount availability are updated. For error scenario, user input other than integer number, negative integer number and value exceeded daily food limit, and system notify an error when updating the available food amount. Assume by default, the daily limit of food amount is 50. Use the following template to design your test cases.

(3 marks)

Test Case ID	(Data Value) entered	Expected result	Actual Results	Pass/Fail

(Note: Redraw the table in answer booklet)

(c) Based on generated test cases in (b), implement black-box testing using equivalence partitioning (EP) technique. Design ONE (1) valid equivalence class and TWO (2) invalid equivalence classes. Use the following template to implement the EP technique.

(6 marks)

Equivalence class for food amount	**Status**	**Representative**	**Expected result**

(Note: Redraw the table in answer booklet)

-End Of Question-