SUNWAY UNIVERSITY

FACULTY OF ENGINEERING AND TECHNOLOGY

FINAL EXAMINATION FOR THE BACHELOR OF SOFTWARE ENGINEERING (HONOURS)

ACADEMIC SESSION : APRIL 2025 SEMESTER

CSC3209: SOFTWARE ARCHITECTURE AND DESIGN PATTERNS

EXAMINATION : AUGUST 2025

TIME ALLOWED : 2 HOURS AND 10 MINUTES READING TIME

INSTRUCTIONS TO CANDIDATES

This exam contains FIVE questions.

Answer **ALL** questions.

All answers must be written in the answer booklets provided using blue or black INK.

IMPORTANT NOTES TO CANDIDATES

Materials Allowed

Standard Items: Pen, Pencil, Eraser or Correction Fluid, Ruler
Special Items : Non-Programmable Calculators, Computer, Tablet,
Notes (Compiled in one Folder)

It is your responsibility to ensure that you do **NOT** have in your possession any unauthorized notes or any other means that would improperly help you in your work. If you have any unauthorized materials with you, hand it to the invigilator BEFORE reading any further.

DO NOT REMOVE THIS QUESTION PAPER FROM THE EXAMINATION HALL

Question 1 **(Total: 17 marks)**

A healthcare insurance provider is developing a management system that must provide a wide range of functionalities including checking the insurance balance, personal details, and panel clinics. Consider yourself as a software architect in this system. You are required to address the following requirements:

First Requirement: Users can check a specific panel clinic map location using, for example, Google Maps, in the system itself directly without the need for browsing the location on Google Maps application separately.

Second Requirement: Allow the users to store and to track their insurance balance and claim details via the Internet.

Third Requirement: Users want to be notified whenever a claim process has been approved, or new promotions are available.

a) Present three suitable architectural patterns to be employed to address those three requirements with a detailed justification. (6 marks)

Architectural Patterns For the specified requirements, a combination of three architectural patterns would be highly effective. To integrate the map functionality without tightly coupling it to the main system, a Microservices Architecture is ideal; the mapping feature can be isolated into its own independent service. For the core requirement of tracking insurance details online, a classic Client-Server Architecture is suitable, where a central server manages all data and logic while clients (user applications) make requests to access it. Finally, to handle notifications for claim approvals and promotions, a Publish-Subscribe (Pub-Sub) Architecture is the best fit, as it allows different parts of the system to publish events without needing to know which users will receive them, ensuring efficient and decoupled communication

b) For each of the selected architectural patterns:

i. Explain an advantage or a quality attribute enhanced by the selected pattern in the scenario. (3 marks)

Microservices (for Map Integration): This pattern enhances modifiability. By isolating the map functionality into its own service, it can be updated or completely replaced with a different map provider in the future without affecting or requiring changes to the rest of the system.

Client-Server (for Data Tracking): This pattern improves scalability. The server component can be scaled up with more resources (like CPU or memory) to handle a growing number of users, independent of the client applications running on users' devices.

Publish-Subscribe (for Notifications): This pattern boosts performance and responsiveness. The publishing service (e.g., the claim approval system) can send a notification and immediately move on to its next task without waiting for the message to be delivered, which prevents bottlenecks.

ii. Explain a limitation or a quality attribute degraded by the selected pattern in this scenario. (3 marks)

Microservices (for Map Integration): This pattern increases complexity. Managing multiple independent services makes the system harder to test and deploy. You must worry about network issues between services, which isn't a problem in a single, unified application.

Client-Server (for Data Tracking): This pattern can degrade availability. The server is a single point of failure. If the server crashes or goes offline for any reason, the entire system becomes unusable for all clients.

Publish-Subscribe (for Notifications): This pattern complicates testability. Because the message sender (publisher) and receiver (subscriber) are completely decoupled, it's difficult to run end-to-end tests to guarantee that a notification was successfully delivered and processed.

c) Consider the following quality attributes: availability, performance, usability, and security. Answer with justification the following:

i. Which of these quality attributes are observable at run time? Explain why? (4 marks)

Availability, because it depends on whether the system is operational and accessible when in use. Performance, because it involves response time, throughput, and resource utilization during execution. Usability and Security are partially observable but mainly assessed through testing or user feedback rather than real-time monitoring.

ii. Provide one example of how to measure each of the quality attributes. (4 marks)

Each quality attribute can be measured using specific metrics. Availability can be measured by calculating the percentage of system uptime over a given period, for example, achieving 99.9% uptime. Performance can be evaluated based on system response time or throughput, such as how quickly a request is processed or how

many transactions are handled per second. Usability can be measured through user satisfaction scores or the task completion rate obtained from usability testing sessions, reflecting how easily users can interact with the system. Lastly, security can be assessed by tracking the number of vulnerabilities or security breaches detected within a certain time frame, indicating the system's resilience against threats.

Question 2 **(Total: 13 marks)**

Assume you are an architect for the same management system stated in Question1. You are required to do the following:

a) Present four quality attribute scenarios and specifications (informally and formally) to address four quality attribute requirements for users. Consider the six parts in quality attribute specification (Source, Stimulus, Environment, Response, Response measure, Artifact). (8 marks)

As the architect for the management system, four quality attribute scenarios are specified to ensure user requirements are met. For performance, a user (source) clicking on a clinic map (stimulus) under normal load (environment) should see the map render fully on the system (artifact, response) in under 2 seconds (response measure). To guarantee availability, a user (source) attempting to view their balance (stimulus) during 24/7 operation (environment) must receive their details successfully from the core services (artifact, response), with the system achieving 99.9% uptime (response measure). In terms of security, an external attacker (source) attempting a web vulnerability like XSS (stimulus) against the live system (environment) will have their request rejected and logged by the web application (artifact, response), with the system preventing 100% of OWASP Top 10 threats (response measure). Lastly, for modifiability, a developer (source) tasked with adding a new notification type (stimulus) to the notification service's source code (artifact) during development (environment) must be able to complete the implementation and deployment (response) in under 8 developerhours (response measure).

b) Present a utility tree that captures the quality attribute scenarios presented in

i) The tree should present all the addressed quality attributes, their scenarios and their impact on business value and architecture. The tree should have nodes for quality attributes, quality attribute refinements, and leaf nodes for the specific quality attribute scenarios. (5 marks) Our design research will focus on two main participant groups: hospitality providers, and adult beginners interested in learning sign language. Hospitality providers such as cashiers, nurses, and NGO workers are also key participants because

they frequently interact with passengers who are deaf or mute, often facing communication barriers during these encounters, as their roles require regular interaction with clients, yet many lack the ability to understand or use sign language effectively.

ii) What is the primary difference between a Microkernel and a Monolithic Kernel. (5 marks)

The significant security advantage of the Microkernel approach is enhanced stability and security isolation. If a bug or security breach occurs in a device driver or file system service, it only crashes or compromises that isolated userspace process, leaving the critical core kernel functioning and preventing a total system failure.