

Sample Exam (Text book - Appendix C)

Appendix C Sample final exam

School of Information Technology

College of Science & Technology: Object-Oriented Analysis
Final Exam. 180 minutes + 10 minutes Reading. **CLOSED book.**

Balance your time according to the Marks.

Use of coloured pens is recommended.

Read the problem statement and the Entire Question paper carefully, before answering the Questions.

Question 1: (10 Marks – 2 Marks each)

- (a) What is the difference between a Class and an Object? Explain with examples.

A class is a template that can be used to describe many separate objects with similar features and behaviour. For example, Sam, John and Darrell can be represented by a class 'Name'. Definition of name will then describe the attributes and behaviour of all name objects. An object is a specific individual entity that is instantiated based on class and that can be described by its identity, attributes and operations. Another example is class 'Car' with object being an individual car with a licence plate of ABC123.

- (b) What is the principle of Abstraction in Object-Orientation? Explain with examples.

Abstraction is the process of creating a class 'template' for a number of similarly classified objects, by identifying and extracting common attributes and operations that all objects in that class share. Abstraction also creates higher-level classes from existing classes to facilitate reuse. For example, a car, a truck, and a motorcycle class can be abstracted to a super-class of vehicle, which has attributes such as number of wheels, and behaviour such as Start Engine.

- (c) How is the Model Of Problem Space (MOPS) different from the Model Of Solution Space (MOSS)? Name and briefly describe three UML diagrams of importance in the problem space.

The Model Of Problem Space describes all the work that goes on in understanding the software or system problem that is yet to be developed. MOPS primarily results from analysis work. The Model Of Solution Space documents the system that will be developed to solve the problem and to implement the requirements. MOSS primarily results from design work.

- Use case diagrams – this shows the functionality of the system, from the user's (actor's) perspective
- Activity diagram – this shows the flow of tasks, and interactions between the actors and system, within a use case
- Business class diagram – this shows the relationships and hierarchies between entities within the system

- (d) Demonstrate your understanding of iterative and incremental software development process by creating an example mini-project plan based on the requirements modelling process-component.

1. Identify packages and create package diagram
2. Develop initial use cases
3. Develop initial use case diagrams
4. Develop initial activity diagrams
5. Identify first cut of classes and develop them

6. Create initial class diagram
7. Refine use cases and use case diagrams
8. Refine activity diagrams
9. Refine classes and update class diagram
10. Develop initial sequence diagrams
11. Develop initial state chart diagrams
12. Refine class diagram further
13. Refine sequence diagrams
14. Refine state chart diagrams
15. Finalise class diagram

- (e) What is the importance of Non-Functional requirements in good analysis? Explain with examples.

Non-functional requirements document necessary aspects of a system that cannot be captured using UML diagrams, including performance, volume, scalability, and security; i.e. the maximum response time for a transaction in an automatic teller machine.

Question 2: (10 marks)

Consider a hospital that has a large medical staff of doctors, nurses and administrators, and which offers medical services to various categories of patients. Name and document in detail TWO use cases as follows:

- (a) first use case deals with registering patient enquiries for surgical procedures and (b) second use case deals with scheduling of doctor's duties. (Reminder: use case descriptions contain at least the following: Use Case: Actors: Description: Pre-Condition: Post-Condition: Type: Normal Course of Events: Alternate Course of Events:

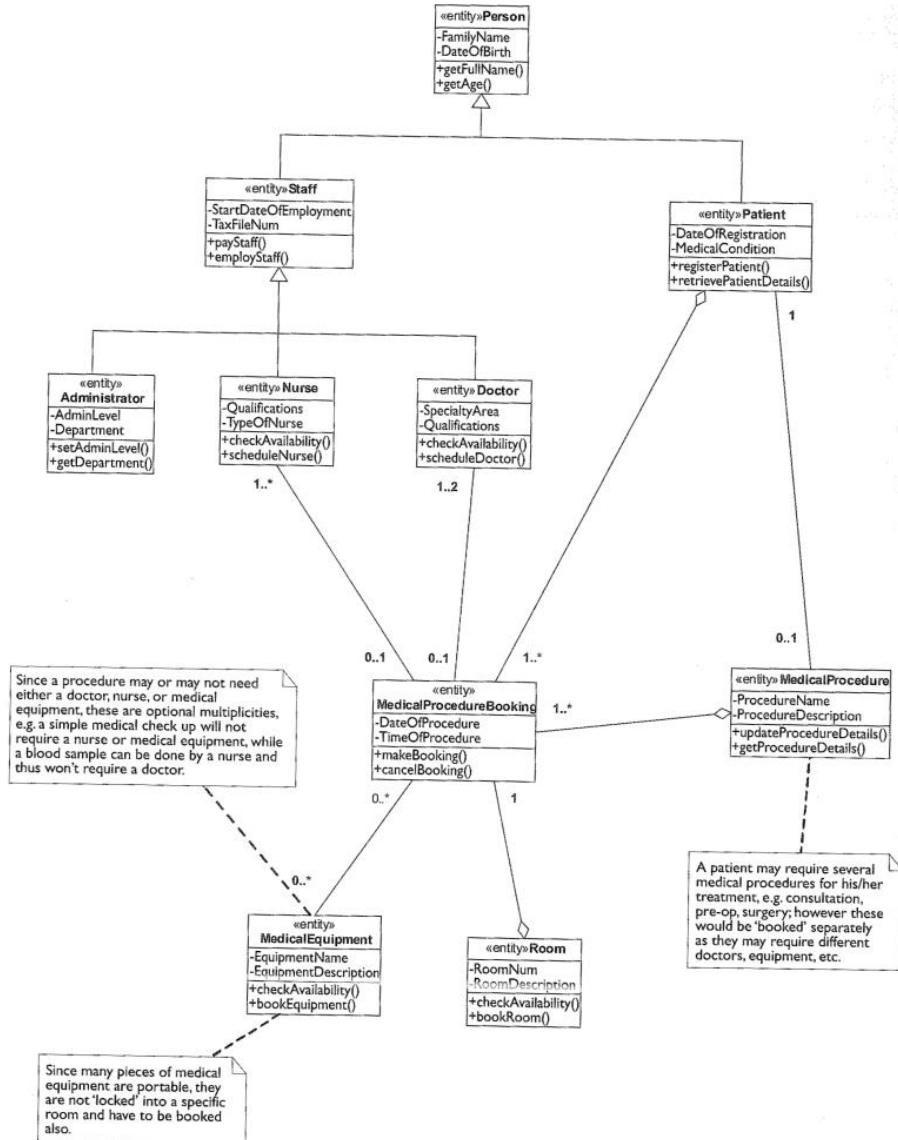
Use Case:	RegistersPatient
Actors:	Patient, Administrator
Description:	A patient needs to register in the hospital's system
Pre-Condition:	None
Post-Condition:	Patient is registered in the system
Type:	Simple
Normal Course of Events:	<ol style="list-style-type: none"> 1. Patient informs administrator that he needs to be registered in the hospital's system. 2. Administrator asks for patient's name details (First, Last, Middle, Aliases) 3. Patient provides administrator with his/her name details. 4. Administrator enters patient's name into system & submits. 5. System searches for matching patient records (A1). 6. System displays no match message and requests full details. 7. Administrator requests additional details from the patient. 8. Patient provides their full details as requested. 9. Administrator enters full patient details into the system & submits. 10. System saves patient details and displays confirmation message.
Alternate Course of Events:	A1. Patient is already registered and the system displays the patient's full details to the administrator.

Use Case:	SchedulesSurgery
Actors:	Surgeon, Administrator, Nurse, Patient
Description:	This use case describes how a surgeon books a surgical procedure with the help of administrator and depending on the availability of nurse and patient.
Pre-Condition:	Patient must be registered with the system
Post-Condition:	Patient is given a date and time, and a booking referral number for the procedure.
Type:	Medium
Normal Course of Events:	<ol style="list-style-type: none"> 1. Surgeon describes the procedure and provides the patient details to the administrator. 2. Administrator verifies with patient on the procedure. 3. Patient confirms procedure. 4. Administrator enters procedure details into the system. 5. System determines availability of the surgeon, nurses, operating theatres, rooms, and medical equipment required to perform the procedure (A1). 6. Administrator offers patient the available dates & times. 7. Patient chooses a date and time (A2). 8. Administrator books surgeon (and nurses, room, and medical equipment) for the chosen date and time. 9. System returns a booking referral number for the patient.
Alternate Course of Events:	<ol style="list-style-type: none"> A1. The specific surgeon or the rooms etc. are not available on the required date. Either the hospital asks the patient to wait, or the patient is referred to another hospital. A2. None of the offered dates & times is acceptable to the patient, so the patient cancels his/her request.

NOTE: Since both processes are Web-enabled then both the patient and the doctor can perform these tasks online. The patient can register his/her details and the doctor can schedule a surgery for a patient without the need for an administrator. The course of events will vary accordingly.

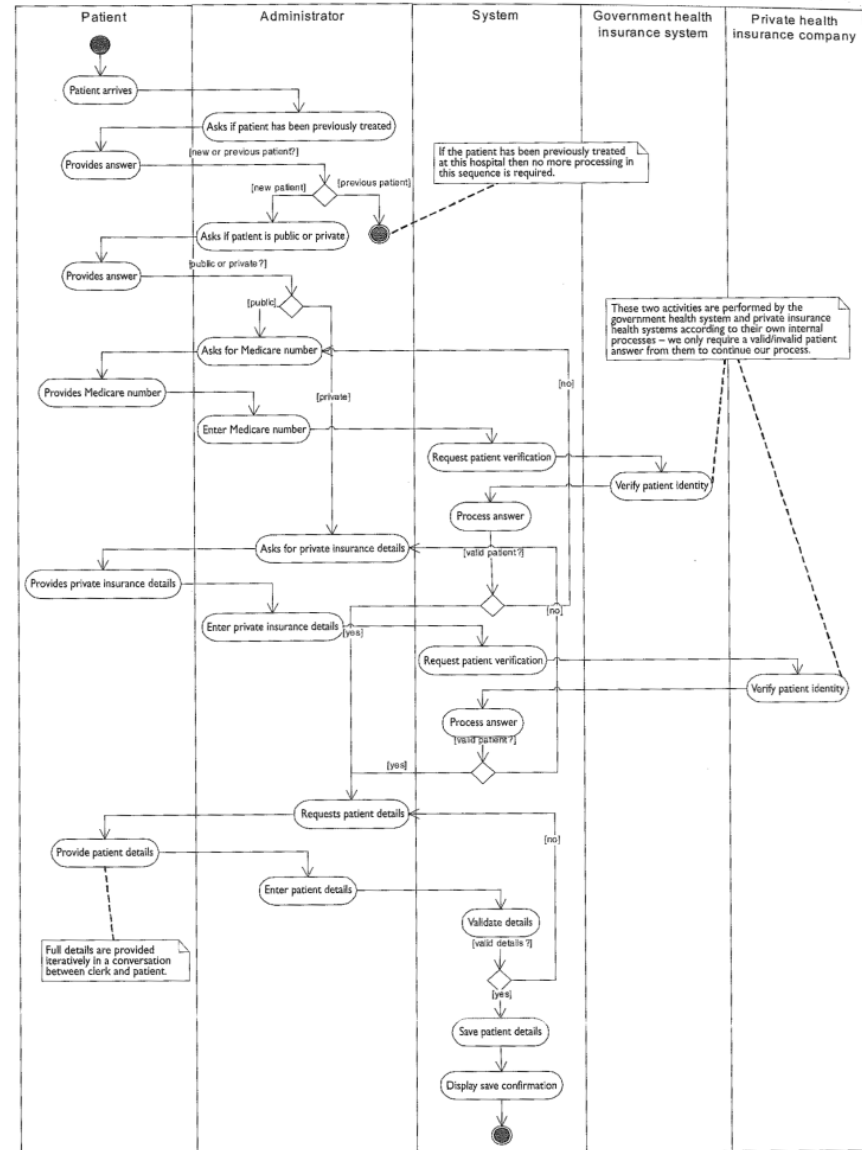
Question 3: (10 marks)

Identify TEN <<entity>> classes based on the two use cases you described in Question 2 and place them in ONE DETAILED class diagram. Ensure you have shown class **relationships**, class **stereotypes** and **multiplicities** on your model. It is essential to show at least TWO attributes and TWO operations for each class.



Question 4 (10 marks: 5 + 5)

- (a) Draw a detailed activity diagram that includes partitions, decision points, multi-threads and notes, corresponding to the text you entered for any ONE use case of your choice in Question 2. Additionally show verification of the patient's details by the government insurance regulatory system and by the selected private insurance company for private patients.



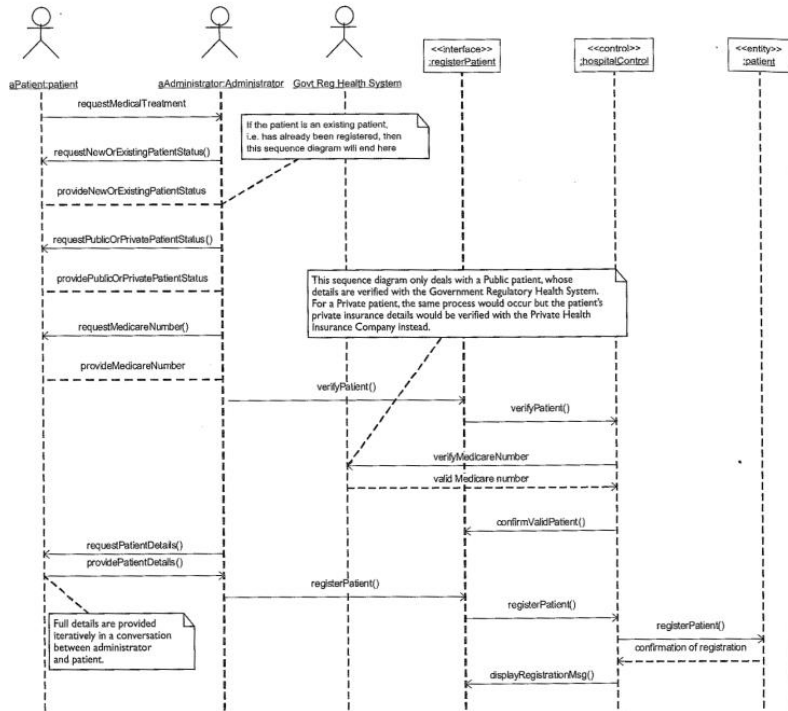
Identification: Test case for Use Case 'SchedulesSurgery'.

Purpose: To verify that the use case *SchedulesSurgery* correctly books a doctor for a medical procedure in the hospital system.

Prerequisites: Patient must be registered in the system.

Input	Actions	Expected Output	Actual Output
Valid data: Attempt to schedule a doctor for a procedure with the following details Family name: Smith Procedure name: Appendectomy Date of procedure: 1/12/2004 Time of procedure: 10:00 am	System will validate the data entered and create a booking for the procedure. A message confirming the scheduled procedure and a booking referral number is displayed.	A booking will be created for the patient for the date/time entered, with a booking referral number. Doctor is scheduled to perform the procedure. System will display the following message: Procedure scheduled for 10:00 am on 1/12/2004. Referral number: 2004120100351	A booking is created for the patient for the date/time entered, with a booking referral number. Doctor is scheduled to perform the procedure. System displays the following message: Procedure scheduled for 10:00 am on 1/12/2004. Referral number: 2004120100351
Invalid data: Attempt to schedule a doctor for a procedure with the following details Family name: Smith Procedure name: Appendectomy Date of procedure: 1/12/1990 Time of procedure: 26:00 am	System will validate the data entered and display an appropriate error message for incorrect fields. User can then correct the data entered.	No booking will be created until the invalid data is corrected. System will display the following message: Invalid data entered for the following field(s): Date of procedure – cannot be earlier than today's date Time of procedure – must be valid time	No booking is created. System displays the following message: Invalid data entered for the following field(s): Date of procedure – cannot be earlier than today's date Time of procedure – must be valid time

- (b) Draw a detailed sequence diagram based on a sequence of your choice (you are welcome to consider the activity diagram drawn by you in Answer 4a). Ensure that entity objects and messages you show in the sequence diagram have a corresponding match in the class diagram drawn in Answer 3.



Question 5 (10 marks: 4 + 6)

- (a) What is the difference between these three terms: Quality Management, Quality Assurance and Quality Control? Explain in detail with suitable examples.

Quality management is the overall strategic approach to quality in the development project. It includes the creation of a quality environment, by putting together a team and a set of procedures that are committed to creating a quality system.

Quality assurance is the application of a software process in order to ensure that the development process results in a quality software product. Quality assurance includes use of the tools, activities and tasks, templates for deliverables, software standards and role descriptions resulting in a matured and repeatable process. Quality assurance's objective is the *prevention* of errors.

Quality control is the testing of the software system to ensure that it functions correctly and as designed. Testing is performed on 'units' of software (e.g. classes) to verify that they operate as expected, and also on the interaction of the units together to verify that the system as a whole works as expected. Quality control deals with *detection* of errors.

- (b) Write TWO detailed acceptance test cases that correspond to the two use cases you have documented in Question 2. Include sample 'Valid' and 'Invalid' test data. (Note: One test case PER one use case is required)

Identification: Test case for Use Case 'RegistersPatient'.

Purpose: To verify that the use case *RegistersPatient* correctly registers a patient in the hospital system.

Prerequisites: None.

Input	Actions	Expected Output	Actual Output
Valid data: Attempt to register a new patient with the following details Family name: Smith Date of birth: 2/3/1970 Condition: Appendicitis	System will validate the data entered and create a new patient in the system. A message confirming the registration will be displayed.	A patient will be registered in the system for the details entered. System will display the following message: Patient 'Smith' has been successfully registered	A patient is registered in the system for the details entered. System displays the following message: Patient 'Smith' has been successfully registered
Invalid data: Attempt to register a new patient with the following details Family name: 123 Date of birth: 2/2/2012 Condition: 12345	System will validate the data entered and display an appropriate error message for incorrect fields. User can then correct the data entered.	No patient will be registered until the invalid data is corrected. System will display the following message: Invalid data entered for the following field(s): Family name – must be alphanumeric Date of birth – cannot be later than today's date Condition – must be alphanumeric	No patient is registered. System displays the following message: Invalid data entered for the following field(s): Family name – must be alphanumeric Date of birth – cannot be later than today's date Condition – must be alphanumeric