

Spring Examinations, 2002/2003

Exam Code(s)	3IF121
Exam(s)	3 rd University Examination in Information Technology
Module Code(s)	CT321
Module(s)	Software Engineering II
Paper No.	1
Repeat Paper	
External Examiner(s)	Prof. P. Nixon
Internal Examiner(s)	Professor G. Lyons
	Dr. O. Molloy

Instructions: Answer Question 1 and any other 2 questions.
All questions will be marked equally.

Duration	3hrs
No. of Answer Books	

Requirements:

Handout	
MCQ	
Statistical Tables	
Graph Paper	
Log Graph Paper	
Other Material	

No. of Pages	4
Department(s)	English

1. SlipDisc Records is a music retailer. They have asked you to create a system to help put their business online. Draw a class diagram for the business objects required in the system. Include associations, inheritance, attributes and operations. Remember to include any appropriate attribute types and initial values, as well as operation signatures (parameter lists with types, and return values). *Note any assumptions you make in interpreting the requirements.*

SlipDisc sell cassettes, CDs, videotapes, and DVD's. The name of the recording artist and the title of the album must be stored for each album. Music albums are available in either cassette or CD format – a separate serial number is assigned for each format. Films come in either videotape or DVD format. The title of the film must be stored as well as the serial number (again, a different serial number is needed to distinguish between the two formats). For every item (unique serial number) in the system, we need to maintain the wholesale price, retail price and current number in stock.

Customers will be asked to register, giving their name, e-mail address and a password. When they return to the system or want to buy online they will be asked for their e-mail address and password. No two customers can have the same e-mail address.

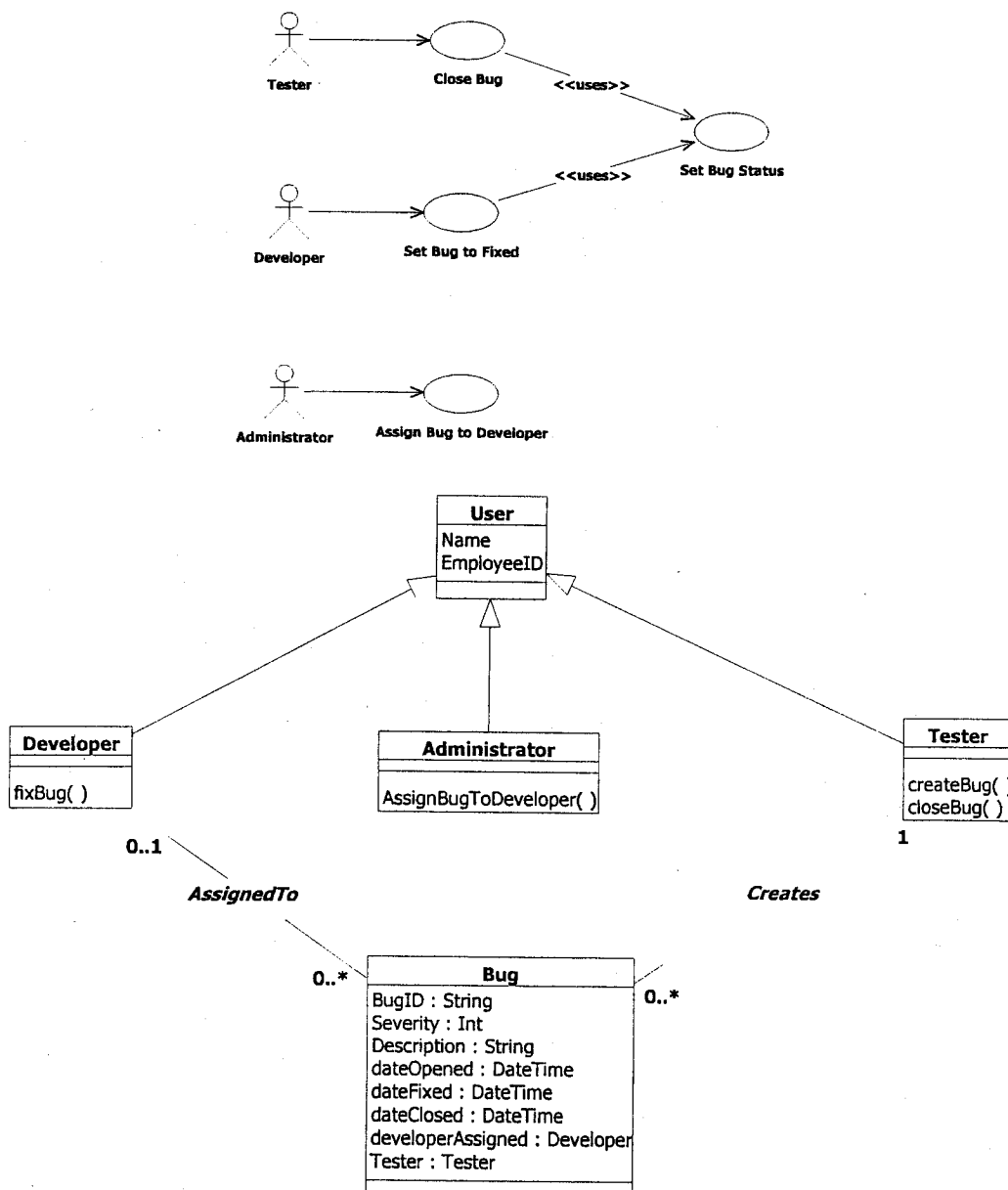
When customers are shopping online they will fill up a virtual shopping basket with as copies of as many cassettes, CDs, videotapes, and DVD's as they like. A running total of the price is kept with the shopping basket. Each customer can be associated with only one shopping basket at a time.

When the customer is happy with the contents of his/her shopping basket, they can order by providing credit card details (card number, cardholder name, and expiry date). They also provide a name and address to which the goods must be shipped. Once the order is confirmed, the contents of the shopping basket are stored on the system as an order. The order object will be passed on to a separate shipping application.

2. (a) Design the GUI classes needed (in addition to the business classes *user*, *developer*, *administrator*, *tester*, *bug* below) to implement the use cases given in the UML diagram for the following system description. Assume you are using dialog boxes to implement the GUI.
 - (b) Create the sequence diagrams, including the GUI objects, for the three main use cases ("close bug", "set bug to fixed", "assign bug to developer").
 - (c) Note any necessary changes to the business classes (classes *user*, *developer*, *administrator*, *tester*, *bug*) given below, and any additional operations or attributes they will need to support the GUI classes in implementing the use cases.

System Description

You are asked design a system to computerise the operations of the test and maintenance division of a software company. The test division has a certain number of software testers. As the tester finds a bug it must be logged. Each bug is tracked by a unique identifier. Other information must also be logged for each bug, such as: product, severity, date and time found, developer assigned to fix the problem, textual description of the problem, date fixed, date retested and date closed. The system must also keep track of which bugs are assigned to which developer at any given time. The system is used by testers, developers and an administrator. Bugs are normally found and created on the system by testers. Each bug is assigned to a developer by the administrator. Developers fix the problem and set the bug status to "fixed". A tester will then set the bug status to closed, provided the problem does not reoccur during retest.



3. (a) Write notes on any four of the following, using examples to illustrate your answer:

Inheritance
 Approaches to Integration Testing
 Multiplicity
 User Interface Design Metaphors
 Stereotypes
 CRC Cards

(b) A system is to be developed to support the management of exercises completed by students taking a course. Students first meet with the course tutor to register for the course, and then during the course they submit a number of pieces of work. At any point, a student can find out from the system the marks they have received for any exercises already completed. The course tutor can enter a mark for a piece of work, and print out a summary of the marks gained by all students on the course.

- i. Describe in detail suitable use cases for the system, and
- ii. Draw a use case diagram for this system.

4. (a) Draw a State Diagram summarising the information given in the following description of some of the events that can arise in the life cycle of a thread in Java.

When a thread is created, it does not start running immediately, but is left in the *New Thread* state. When the thread is in this state, it can only be started or stopped.

The *start* method causes system resources to be allocated to the thread and calls the threads *run* method. At this point the thread is in the *Running* state.

A thread becomes not runnable if either its *sleep* or *suspend* methods are called. The *sleep* method has a parameter specifying the length of time the thread should sleep for; when this time has elapsed the thread starts to run again. If the *suspend* method has been called, the thread only runs again when its *resume* method is called.

A thread can die in two ways. It dies naturally when its *run* method exits normally. A thread can also be killed at any time by calling its *stop* method.

- (b) Write notes on the following, *using examples to illustrate your answers*:

- ☐ Guard Conditions
- ☐ Event triggers with parameters
- ☐ Entry and exit actions

5. (a) Describe main differences between White Box Testing and Black Box Testing

(b) For the program described by the following flow chart, derive each of the following:

- I. Cyclomatic Complexity
- II. Flow Graph
- III. Set of Basis Test Paths needed to test the program

