

*Ollscoil na hÉireann, Gaillimh*  
*National University of Ireland, Galway*

GX 1625

**Semester II Examinations, 2002/2003**

Exam Code(s)	<u>4IF1</u>
Exam(s)	<u>Fourth Year Information Technology</u>
Module Code(s)	<u>CT432</u>
Module(s)	<u>Distributed Systems</u>
Paper No.	<u>1</u>
External Examiner(s)	<u>Prof. P. Nixon</u>
Internal Examiner(s)	<u>Prof. G. Lyons</u>
	<u>Dr. D. Chambers</u>

**Instructions:**

Answer any 5 questions.  
All questions will be marked equally.

Duration	<u>3hrs</u>
No. of Answer Books	<u>1</u>
No. of Pages	<u>5</u>
Department(s)	<u>Information Technology</u>

- 1.a: Describe the Delegate-Model Architecture. Using a simple Bank Account as an example show how this architecture could be implemented in a Java environment. Your design should include an AccountController to modify the state of Account objects and at least one view object to display the Account state. Use the built-in Java implementation of the Observer Design Pattern in your design.

10 MARKS

- b: Explain, using a suitable code example (e.g. a Shopping Cart), the operation of the Session Tracking mechanism available in the Java Servlet API. How would you support session tracking for users that access a servlet with a browser that does not support cookies, or that is set up to reject cookies? 10 MARKS

2. Using Java Remote Method Invocation, outline the design for an Internet based automated assessment system for the university. The server allows users to authenticate themselves and download an assessment object. This object implements an interface that provides methods to retrieve and answer a list of multiple-choice questions. The (updated) object can then be submitted back to the server for verification and correction. The following interfaces are required:

- *ExamServer* - this (remote) interface provides methods for user authentication, download of assessments and the submission of completed assessments. Assessments can only be downloaded and submitted during certain specified time intervals.
- *Assessment* - this (serializable) interface should provide methods for the retrieval of information about the assessment, and the retrieval / answering of questions. It should also have a method to output the selected answer to each question - the answer provided to a question can then be changed, if desired, prior to submission of the assessment. The assessment interface also provides a privileged (password protected) method to correct itself i.e. provide a mark and output the list of correct / incorrect answers. Obviously, the password used should not be made available to clients and should be setup on the server when the assessment object is created.

The design of the system should make it possible for new Assessment implementation classes to be easily added to the system in the future, making the system very flexible. The design should use Java RMI and Object Serialisation to download and then submit objects that implement the Assessment interface i.e. these objects are passed by value from the server to the client and then back again to the server. Full implementation classes are not required but the answer should include source code for the Java interfaces described above. Also include the mainline server code to initialise the server and show how a simple client program might use or interact with the server.

20 MARKS

- 3:a: In the context of distributed object technology, discuss briefly what is meant by a "Smart Proxy". What complications can arise when using smart proxies and how might these be resolved? 4 MARKS
- b: Describe how incoming requests are processed in a CORBA Server using the the *Portable Object Adapter*. Show, using a suitable example, how a server application could create a *Child POA* to use a different set of Policies from the default policies used by the Root POA. 8 MARKS
- c: Describe the function and operation of the CORBA Event Service. Include in your description the main interfaces provided by this service. What kind of applications can benefit from using this service? 8 MARKS
4. The following Corba IDL file contains the interface definition for a Currency Converter object. The interface has methods to convert between Euros and other currencies, it can also be loaded with new conversion rates as required.

```
// File CurrencyConverter.idl
struct ConversionRate
{
    String currencyName;
    String euroValue;
};
typedef sequence <ConversionRate> ConversionRates;
interface EuroConverter
{
    readonly attribute ConversionRates currentRates;
    float convertFromEuro(in float amount, in string currencyName);
    float convertToEuro(in float amount, in string currencyName);
    void loadNewRate(in ConversionRates newRates, in string password);
};
```

Using the Java language, write a suitable implementation class for the *EuroConverter* interface. Also write suitable server mainline code to create and register (with the Implementation Repository or Corba Name Server) a single instance of this object. Note that sequences in Corba IDL map to simple arrays (of the appropriate type) in Java.

Assuming that an instance of *EuroConverter* is set-up on node *geminga* with the registered server name *Converter*, write the client code in Java to query the server for some currency conversions. 20 MARKS

- 5.a: Describe briefly the advantages of using the EJB component framework in the context of high volume distributed object applications. What types of beans may be defined using the EJB framework? 4 MARKS
- b: Using the example of an *Employee* within a typical organisation, write the Java code to represent an *Employee* as an Entity EJB that uses Container Managed Persistence. The EJB remote interface for *Employee* should include relevant accessor methods for setting and getting information. The EJB home interface should include create and finder methods. 8 MARKS
- c: Web services represent an evolution and convergence of a number of important areas of technology and business. Describe briefly these technology areas and explain how Web Services builds on previous capabilities. Include in your explanation an overview of the main enabling technologies used to provide Web Services. 8 MARKS
6. You have been asked to develop a commercial online bookstore using J2EE based technologies. The bookstore architecture and design should be able to support different types of client browsers and should use a three-tier application model i.e. a client tier to support different clients, a middle tier that implements the application business logic and an information tier to persist the application state. Based on these requirements:
- a. Describe the top-level application architecture. Identify the technologies that will be used and explain the role each of these technologies plays in the overall system architecture. 8 MARKS
- b. Identify the various Session and Entity EJBs that will be needed. Explain the functionality that will be provided by each EJB (source code for the EJBs is NOT required). 7 MARKS
- c. Explain how you could generate an appropriate web interface for different types of clients e.g. standard HTTP web browsers and WAP enabled phones. 5 MARKS

7.a: Explain briefly how failure recovery mechanisms might cope with the following situations:

- Processor crashes
- Communications failure
- Media failure
- Deliberate *aborts* by users

4 MARKS

b: Discuss the various policies that affect the design of distributed load balancing systems. Consider the example of adding load balancing capabilities to a *Unix Shell*, which algorithm do you think would work best in this case? How would varying the load exchange period affect the results?

8 MARKS

c: Describe the main properties of an *atomic transaction*. Explain how, using the two-phase commit protocol, an atomic commit might be achieved in a distributed system.

8 MARKS