

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

GX 0084

**Semester I Examinations, 2005/2006**

|                      |  |
|----------------------|--|
| Exam Code(s)         | <u>4BP1</u>  |
| Exam(s)              | <u>Fourth Year Electronic and Computer Engineering</u> |
| Module Code(s)       | <u>CT414</u>   |
| Module(s)            | <u>Distributed Systems</u>                             |
| Paper No.            | <u>1</u>   |
| External Examiner(s) | <u>Prof. J. Keane</u>                                  |
| Internal Examiner(s) | <u>Dr. M. Madden</u>                                   |
|                      | <u>Dr. D. Chambers</u>                                 |

**Instructions:**

Answer any 4 questions only.  
Additional questions, if answered, will not be marked.  
All questions will be marked equally.

|                     |                               |
|---------------------|-------------------------------|
| Duration            | <u>2.5 hrs</u>                |
| No. of Answer Books | <u>1</u>                      |
| No. of Pages        | <u>4</u>                      |
| Department(s)       | <u>Information Technology</u> |

1. Using Java Remote Method Invocation, outline the design for an Internet based questionnaire / survey system. The server allows users to get a list of questionnaires available from the system and to download a questionnaire that can then be completed locally, using a suitable client application. The completed questionnaire can then be submitted back to the server for verification and analysis. The following interfaces / classes should be provided:
  - *SurveyServer* - this (remote) interface should provide a number of methods as follows: a method to download an array (of *QuestionnaireSummary* objects) containing a summary of the questionnaires available on the server; a method to download a full questionnaire; and a method to submit a completed questionnaire.
  - *Questionnaire* – this (serializable) interface is used to access a particular questionnaire. A questionnaire is made up of multiple questions and each question can either present a multiple choice option to the user or allow the user to enter a text based answer. The interface should provide methods for the retrieval of information about the questionnaire, 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 questionnaire.

The design of the system should make it possible for new Questionnaire 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 Questionnaire 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 (for the SurveyServer and Questionnaire interfaces) are not required but the answer should include the full 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.

25 MARKS

- 2.a: 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

- b: 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.

15 MARKS

3.a: Describe briefly the typical architecture for a Distributed Operating System that uses the *processor pool model*. What types of servers are normally used to support this model? 5 MARKS

b: Describe the main differences between a *two-tier* and a *three-tier* Client-Server architecture. Include in your description the main limitations of using a *two-tier* approach and how using a *three-tier* approach can overcome some of the potential problems. 10 MARKS

c: Explain the semantics of a typical synchronous *Remote Procedure Call* operation. Describe briefly the purpose of the following components, as used in the Microsoft RPC environment:

- IDL Compiler
- Runtime Library and Header Files
- RPC Name Service
- RPC Endpoint Mapper

10 MARKS

4.a: Explain briefly the functionality of the following CORBA Components:

- Proxy Objects
- Object Adapter
- CORBA Name Service
- Interface Definition Language
- Dynamic Invocation Interface

10 MARKS

b: Using the example of a simple Employee Directory Service, outline the steps required to implement this as a CORBA based application. The following design guidelines will apply:

- It should allow any user to view all other entries.
- The service should allow users to modify their own entries.
- Managers will have extra privileges that allow them modify / delete access to the entries for employees in their department.
- Possible list of interfaces include directory, user, manager, department.
- To keep the task simpler, nesting of departments (orgchart type functionality) is not required!

Include in your answer the *IDL* file definitions and the steps required to complete the application development. Also show how a simple client program might use or interact with the server. Full source code for the implementation classes **is not** required. 15 MARKS

- 5.a: Outline the basic purpose of name resolution services. Using the CORBA standard name service as an example, show briefly how a client program could utilise this service. 5 MARKS
- b: 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. 10 MARKS
- c: 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 *immobility factor* affect the results? 10 MARKS
- 6.a: Describe briefly some of issues that often arise in the design of Distributed File Services. 5 MARKS
- b: 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 could benefit from using this service ? 10 MARKS
- c: Outline the design of the Secure Sockets Layer (SSL) Handshake Protocol. In particular, explain how the following issues are addressed within this protocol:
- Client and Server Authentication.
  - Secure sharing of session keys.
- 10 MARKS