NOTTINGHAM TRENT UNIVERSITY

School of Science and Technology

COURSEWORK ASSESSMENT ELEMENT

MODULE CODE	:	COMP20081
MODULE TITLE	:	Systems Software
MODULE LEADER	:	Dr Evtim Peytchev
TUTOR(S)	:	Dr Chris Windmill, Dr Jon Robinson
COMPONENT	:	1 of 3
TITLE	:	Coursework for Part 2 of the Systems Software Module (Internetworking with Java)
LEARNING OUTCOMES ASSESSED	:	3 to 11
WEIGHTING	:	50% of the total coursework mark 25% of the overall module mark
DISTRIBUTION DATE	:	9th March 2012 (Teaching Week 19)
SUBMISSION DATE	:	**23rd April 2012** (Teaching Week 24) for report and code; Demonstration of Task 1 in **week beginning 19th March**; Demonstration of Task 2 in **week beginning 23th April 2011**.
SUBMISSION METHOD	:	Electronic version of the **report** along with **all code** to be submitted to dropbox in addition to hard copy of the report handed in to the Student Information Desk; Demonstrations in timetabled sessions (see above).
NOTE	:	The usual University penalties apply for late submission and plagiarism. Please consult your student handbook for further details.

I. Assessment Requirements

To pass the second coursework component you must demonstrate a solution to **BOTH** of the tasks described in section II. You may use the example programs as a source but you must be in a position to be able to explain in detail the workings of your code. Failure to demonstrate and explain the first task will lead to a penalty of 10 percentage marks being deducted from your score on the second task. The solution to the first task must be demonstrated in the practicals to your lab tutor during the week of **19th March – 23rd March 2012**. The overall coursework (task 2) must be demonstrated and explained to your lab tutor in the labs during the week of **23rd April – 27th April 2012**. If you fail to explain certain feature of the coursework, the marks for the feature will not be awarded to you. You are required to submit short (min 3 pages) explanation of your coursework solution. Hand in the write-up for the coursework by the **23th April 20012** (note – you may have to hand in the documentation before you demonstrate your solution). **You must also submit an electronic version of your report along with code to dropbox.** The code that you demonstrate **has** to be the same code that is submitted electronically.

II. Assessment Scenario/Problem

Task 1: Simple string sending/receiving program.

This first piece of work you should do on your own.

Network Programming using Datagrams. This application comprises of two programs running on different machines such that one program will transmit a String of text to the other which will reply with another String. The choice of text is up to you but should include both your names and "N" number. No documentation is required for this first exercise - just show that you can communicate using two different computers. Demonstrating this task will help you prepare for the full coursework.

Task 2: Full coursework demonstration: - Java program for tickets booking, selling and reservations

This second piece of work you must do alone or in a pair with a colleague, not necessarily from the same seminar or lab group.

If you elect to work as a pair then equal marks will normally be given to both parties but the module leader reserves the right to reduce the marks for a contributor if it is clear there has been an imbalance of effort.

The assignment is about delivering an on-line service booking system. The topic of the bookings is up to you. You can choose booking tickets for a cinema, air travel, restaurant seats, or car hire – any system that features two or more clients and a server responsible for the bookings. You must implement the client server communication and manage the bookings yourself – without the need for relational database programming. The customers should be able to: request a display of the available positions for a given date (no more than 1 month of data ahead is OK); make reservation(s) for a given date; cancel a reservation and download currently active messages from the booking company (special offers) etc.

The system will make use of datagram Sockets technology (you can use Stream Sockets if you prefer) and will consist of a server and multiple clients. You are to implement a simple information server and several client objects. You need not worry about producing a graphical user interface (but if you do it will be worth 10 extra marks).

The functionality of the Client is that it should be able to:-

- connect to the server and register its IP address as a valid user along with the name of the user in order to join the service (i.e. login).

- remove the user from the service (i.e. when disconnecting to execute log-off command).

- send a message to the server specifying the information required: request the number of bookings available for the day for a given event; make reservation; cancel reservation; request information about special offers.

- asynchronously receive and display urgent messages from the information server that the number of bookings for a given /date have all been reserved and there are no more free bookings available for this date. The second urgent message is that a fully reserved date has just had a cancellation and now has at least 1 booking available (when somebody else cancels the booking). This requirement can only be properly achieved through the use of multi-threading.

The required functionality of the Server is that it should be able to:-

- keep a record (in some form) of all on-line registered participants and their IP addresses.

- accept new participant registrations and add them to the active participants list.

- accept requests from the client to remove itself from its records (log-off).

- read/save the reservations from/to a text file consisting of multiple lines with the format below.

- receive messages and perform the actions required (report free bookings for a given date, reserve a ticket, cancel reservation, send special offers available messages.

- handle multiple client connections concurrently.

- maintain correctly the list of tickets for all dates available in the booking system.

The format of the file for reading/saving reservations from/to the text file to store one or more lines describing the performances and the bookings available should be as follows:

< event ><date><total Number of Bookings>< reserved bookings number><Customer Name>

Full marks will be awarded for implementing ALL of the required functionality. **The mark awarded will be proportionate to the functionality implemented** – you can pick and choose which functionality you want implemented. The design of the system is up to you. Demonstration and explanation of the prototype is required. Please also refer to the marking scheme defined separately in Java Coursework Marking Scheme file.

As well as demonstrating your Information Server and Client you must submit a short report to include:-

1. List of the students presenting this particular implementation.

2. Commented listings of the client and server

3. An explanation of the design of your system which details items such as :

    - the nature of the network connection used. That is if the server holds an open connection to each client or not and if connections are pooled or created on the fly as temporaries.

    - the data structure used by the server to record active participant information

    - how the client handles the asynchronous communication from the server.

    - any other enhancements you have made for extra points – you will get additional marks for every feature even if the feature is part of a higher band marking requirements and you have not

fulfilled some of the lower bands marking requirements.

III. Assessment Criteria

The assessment will be carried out in two parts – task 1 – no documentation required (**penalty 10 marks if not demonstrated and adequately explained**).

For detailed information on the assessment of the second part see the additional file – Java Coursework assessment sheet.

IV. Feedback Opportunities

Formative (Whilst you're working on the coursework)

You will be given the opportunity to receive informal verbal feedback from your lab tutor regarding your coursework development during the practical sessions.

Summative (After you've submitted the coursework)

You will receive specific feedback regarding your coursework submission together with your awarded mark when it is returned to you. Clearly, feedback provided with your coursework is only for developmental purposes so that you can improve for the next assessment or subject-related module.