LIVERPOOL JOHN MOORES UNIVERSITY

Faculty *of* Engineering & Technology

Department *of* Computer Science

Coursework Title: **eAuction System OOP**

Module Name: **Object Oriented Systems**

Module Code: **5104COMP**

Level: **5**

Credit Rating: **20**

Weighting: **60%**

Maximum Mark Available: **100**

Lecturer: **Mr. Glyn Hughes**

Contact: *If you have any issues with this coursework you may contact your lecturer whose contact details are:*

eMail: **g.d.hughes@ljmu.ac.uk**

Room: **604B**

Hand-Out Date: **15th Mar 2019**

Hand-In Date: **8th Apr 2019**

Hand-In Method: **Canvas**

FeedBack Date: **29th Apr 2019**

FeedBack Method: **eMail**

Programme(s): **CS, CSc, SE, CF**

Introduction:

Using the model **OOAD** (**O**bject **O**riented **A**nalysis & **D**esign) solution for the eAuction system as a basis, employ an **OOP** (**O**bject **O**riented **P**rogramming) process to develop a working prototype.

Working in **groups** *of* **two** or **three**, your prototype should be developed as a **Console Application**, using either **Eclipse** (Oxygen) and written in **Java** or **Visual Studio 2017** and written in **C#.**

You will be required to demonstrate your working prototype.

Learning Outcome(s) Being Assessed:

1. *<not assessed in this coursework>*
2. *<not assessed in this coursework>*
3. Implement object oriented designs using object oriented program code.
4. Employ an application programming interface using an integrated development environment.

Details *of* Task:

Every UML diagram in the model solution is important as each one models some aspect *of* the eAuction system. However, the most important diagram is the Class diagram as it forms not only the starting point *of* your prototype's implementation but also models the majority *of* the eAuction system's structure.

Ensure that your prototype is able to execute the tasks that are isolated in the Use Case diagram. The specific steps involved in executing these tasks are described either in the requirements document or the model solution. Where steps are poorly described you must use common sense / assumptions to implement those steps.

The eAuction system will require a menu system that should be implemented in the *System* class. You are not required to store data (i.e. a Database or Comma Separated Value .csv file). You can as an alternative, hard code test data for *Users*, *Auctions* etc.

What you should hand in:

The working prototype in a ZIP file. Specifically, a single Eclipse (Oxygen) project directory complete with all Java (.java) source code or a single Microsoft Visual Studio 2017 project directory complete with all C# (.cs) source code.

Include in the ZIP file, a statement *of* group membership and the relative contribution (out *of* 100%) *of* each group member to the coursework. This statement must be signed by each student.

Demo sessions will be run during labs sessions on the **9th Apr 2019**.

Marking Scheme/Assessment Criteria:

Assessment	Assessment Criteria	% weighting *for* part
1	Accurate Implementation *of* **Class Structure**.	30
2	**Functionality** *of* Class Members (especially Methods).	30
3	**Demo** *of* Working Prototype (Console Application).	30
4	Suitable / Descriptive **Commenting** / **Documenting**.	10

Guidelines:

- Correctly reference resources that you use.
- You must annotate your source code with suitable / descriptive comments that describe functionality and any assumptions.
- Periodically and during lecture time, workshop sessions will run to help guide you during your OOP *of* the eAuction system.

Resources Required:

You may use the computing labs on the 6th & 7th floors *of* the Byrom Street Campus as well as the 1st floor *of* the Henry Cotton Campus.

You should make use *of* these specific tools & resources:

- Eclipse (Oxygen). *for students wishing to use Java.*
- Java Language Tutorials.
    - http://docs.oracle.com/javase/tutorial/

- Microsoft Visual Studio 2017. *for students wishing to use C#.*
- C# Programming Guide.
    - http://msdn.microsoft.com/en-us/library/67ef8sbd.aspx

- Lecture Materials.
- The Internet.

Extenuating Circumstances:

If something serious happens that means that you will not be able to complete this assignment, you need to contact the module leader as soon as possible. There are a number *of* things that can be done to help, such as extensions, waivers and alternative assessments, but we can only arrange this if you tell us. To ensure that the system is not abused, you will need to provide some evidence *of* the problem.

More guidance is available at:

https://www.ljmu.ac.uk/about-us/public-information/student-regulations/guidance-policy-and-process

Any coursework submitted late without the prior agreement *of* the module leader will receive 0 marks.

Academic Misconduct:

The University defines Academic Misconduct as 'any case *of* deliberate, premeditated cheating, collusion, plagiarism or falsification *of* information, in an attempt to deceive and gain an unfair advantage in assessment'.

This includes attempting to gain marks as part *of* a team without making a contribution. The Faculty takes Academic Misconduct very seriously and any suspected cases will be investigated through the University's standard policy (*https://www.ljmu.ac.uk/about-us/public-information/student-regulations/appeals-and-complaints*).

If you are found guilty, you may be expelled from the University with no award.

It is your responsibility to ensure that you understand what constitutes Academic Misconduct and to ensure that you do not break the rules. If you are unclear about what is required, please ask.

For more information you are directed to following the University web pages:

- Information regarding academic misconduct:
  https://www.ljmu.ac.uk/about-us/public-information/student-regulations/appeals-and-complaints
- Information on study skills:
  https://www2.ljmu.ac.uk/studysupport/
- Information regarding referencing:
  https://www2.ljmu.ac.uk/studysupport/69049.htm