Module Title	Module Number	JACS Subject Code and % of each subject	ASC Category
Advanced Programming	CIS6003	I320	6

Level (3 - 8)	Credits	ECTS Credit	Module Value	% Taught in Welsh	Module Type
6	20	10	2.0	0%	Taught

Teaching Period	Pre-requisites
Terms 1, 2 & 3	None

Module Leader	School	Campus
Dr Giles Oatley	Cardiff School of Management	Llandaff

Assessment Methods			
Assessment Type	**Duration/Length of Assessment Type**	**Weighting of Assessment**	**Approximate Date of Submission**
WRIT1 – Software Project	4000 words equivalent	100%	May

Aim(s)

This module is designed to provide the student advanced theoretical and with industry standards/practical knowledge of program and software analysis, design, and construction, contextualised by the use of appropriate tools, methodologies and techniques to create useful and usable software in industry.

Learning Outcomes

On successful completion of the module, students should be able to:

- Demonstrate fluency in contemporary programming languages, development tools and environments.
- Evaluate and demonstrate the theory and concepts of contemporary/industry standard programming and design in the software development life cycle.
- Demonstrate awareness of industry standards of professional and ethical software development, software carpentry and codemanship.

Learning and Teaching Delivery Methods

Lectures	24 hours
Seminars and Workshops	24 hours
Independent Study	152 hours

The teaching strategy is based upon lead lectures and a series of supporting workshops.
The purpose of the workshop sessions is to clarify the methodologies outlined in the lead lectures.

Indicative Content

This module takes a student from being an advanced beginner to an 'entry-level' industry programmer, ready to start in graduate level employment.
Advanced principles of programming, programming language theory, evaluation of appropriate programming paradigms.
Refactoring to design patterns, Design by contract, Test-driven development, pair programming
Advanced Object-Oriented Programming: Collections, iterators, domain model programming, Test-driven development, refactoring, exceptions, Design by Contract, Threads, managing memory, serialization, building GUI's, design patterns, Documentation.
Object-Oriented Theory: object-oriented design, four principles of abstraction, inheritance, encapsulation, subtype polymorphism, Open-Closed Principle, Liskov Substitution Principle, Interface Segregation Principle, Dependency Inversion Principle, Single Responsibility Principle, Design by Contract
Object-Oriented Design Practices: The Unified Process, various UML diagrams.

Specific Module Goals for OOP:
Be able to synthesize and document an object-oriented design using object-oriented analysis and design techniques.
Develop an awareness of the development processes that underlie OO development.
Learn the principles of designing and constructing Engineered Interactive Interfaces.
Develop a critical awareness of the concepts of event driven and client server application development.
Choose and implement appropriate OO solutions from problem descriptions, using a toolkit.
Critically appraise, evaluate, and implement designs using Java.
Understand current development practices and environments.
Note that these map naturally to the formal learning outcomes of the module.

Required and Recommended Reading

Required Reading
Kerievsky, J. Refactoring to Patterns (1st ed)

Bloch, J. (2008) Effective Java (2nd ed)

Recommended Reading
Clean Code: A Handbook of Agile Software Craftsmanship 1st Edition
by Robert C. Martin

Lethbridge, T. & Laganiere, R., 2005. *Object-Oriented Software Engineering.* 2nd Edition. McGraw-Hill.
Cooper, J.W., 2000. *Java Design Patterns.* Addison Wesley.

Kernighan, B.W. and Pike, R. (1999) *The Practice of Programming*. Addison Wesley
Mitchell, J. C. (2002) *Concepts in Programming Languages*. Cambridge University Press

Dowek, G. (2009) *Principles of Programming Languages*. Springer

Access to Specialist Requirements

None

CMU B.Sc. (HONS) SE /B.Sc. (Hons) BIS- ASSIGNMENT FEEDBACK SHEET –ICBT CAMPUS

Student Details (Student should fill the content)				
Name				
Student ID				
Scheduled unit details				
Unit code	CIS6003			
Unit title	Advanced Programming			
Unit enrolment details	Year	3		
	Study period			
Lecturer				
Mode of delivery	Full Time			
Assignment Details				
Nature of the Assessment	**Course work**			
Topic of the Case Study	**Mobile accessories selling system**			
Learning Outcomes covered	**1,2,3**			
Word count	4000			
Due date / Time	5th March 2022			
Extension granted?	Yes	No	Extension Date	
Is this a resubmission?	Yes	No	Resubmission Date	

Declaration			
I certify that the attached material is my original work. No other person's work or ideas have been used without acknowledgement. Except where I have clearly stated that I have used some of this material elsewhere, I have not presented it for examination / assessment in any other course or unit at this or any other institution			
Name/Signature		Date	
Submission			
Return to:			

Result				
Marks by 1st Assessor		Signature of the 1st Assessor		**Agreed Mark**
Marks by2nd Assessor		Signature of the 2nd Assessor		

Comments on the Agreed Mark.			
For Office use only (hard copy assignments)			
Receipt date		Received by	

STUDENT NAME:	**STUDENT NUMBER:**
Module Number & Title:	**Semester:**
Assignment Type & Title:	
For student use: ***Critical feedback on the individual progression towards achieving the assignment outcomes***	
<u>For the Assessors' feedback</u> **Indicate the Task number strength and Weaknesses and the marks for each task**	
Task No/Question No	**Strengths**

Task No / Question No	Weaknesses

Areas for future improvement

Marks			
Task /Question No	Allocated Marks	Awarded Marks	Remarks
Total Marks			

Name and the Signature of the Assessor	
Date	

Upon successful completion of this module, you will have demonstrated:

- Demonstrate fluency in contemporary programming languages, development tools and environments.
- Evaluate and demonstrate the theory and concepts of contemporary/industry standard programming and design in the software development life cycle.
- Demonstrate awareness of industry standards of professional and ethical software development, software carpentry and codemanship.

Coursework –**Mobile accessories selling system**

"TechMart" is a company which sells a variety of mobile phones and accessories. It has several branches (Galle, Kandy, Nugegoda, Gampaha, Kurunegala, Jaffna) connected with the main branch located in Colombo.

The customers can get registered first to the system. Then a registered customer can order any item available via online. But when the orders are made, the **delivery location will be restricted only within cities where branches are located**.

The sales agents from all the branches are already registered in the system. They can view and would be able update the state of the orders in the queue (confirmed, processing, out of delivery, delivered) **only in which the delivery location provided belongs to their branch**. The orders will be processed and released for delivery with the delivery service available in the company by the sales agents. The sales agents will also check for the inventory of their respective branches and will send the inventory restock requests to the respective suppliers, when needed.

The suppliers, who are already registered can view the inventory refilling requests from their side and can take actions accordingly.

Customer

Customers can register to the system themselves, search for a respective product in the system, place an order, view the orders they made along with the details, view the status of the order and can leave a feedback once the ordered items are delivered.

Sales agent

Sales agent should belong to a specific branch. They can view the pending customer orders which are not confirmed yet and belong to their branch, view customer orders confirmed by themselves, process the orders (update the status), check for the all inventory status of their branch (product name, description, available quantity, supplier of the product etc), check for the inventory status

(available quantity) which are under a specific number, make inventory restock requests for the respective suppliers of their branch, view inventory restock requests made by themselves, update the inventory, view the total sales made by themselves and view both vehicle and driver details available for delivery only within their respective branch etc.

Suppliers

Suppliers can view all the inventory restock requests made by sales agents.

Admin

Admin can add new sales agents to the systems, update, remove and view the details of the sales agents of all the branches s, view the customer orders (along with customer details, ordered items details and sales agent details who made it confirmed) of all the branches, view inventory details, add new product items to the product list(inventory) and view the total sales made by the sales agents individually and as branch wise etc.

Customer order

Once the customer order is confirmed, processed and reach for the state of "Out of delivery", the respective delivery information (vehicle number, driver id etc) should also be updated by the sales agent who confirmed and processed it. This delivery information should be available in customer, respective sales agent and admin views. Once a sales agent confirmed an order, it should be removed from the pending order list of all sales agents in the respective delivery region.

Delivery vehicles and Drivers

Both entities should belong to a specific branch.

Products in the inventory

Entire company has a specific set of products and brands they sell in common.

Note:

- *Admins of the system are the officials, specifically appointed by the main branch.*
- *The payments for the customer orders can only be made both as a bank transfer or as cash on delivery.*

Provide a well-designed, user friendly system addressing the following features:

- System should have differential access rights to the system users.
- Interactive user friendly interfaces
- Clear implementation of the business flow via the system.
- Design & implement suitable sets of reports, which you think will add more value to the entire business
- Use test driven development and include test classes to test your application

Students are free to make necessary assumption on system design & granting access permissions other than mentioned within the scenario, but all suggestions must be well explained with the Valid reasons.
Students can add any functionality which will enhance the system and make the proposed Solution more comprehensive.

Use Harvard referencing to properly acknowledge all the external sources you use.
Your tasks

Tasks A:
Provide a requirement specification for the proposed system. (06 marks)

Tasks B:
Provide the UML diagrams for the given problem with clear explanations on the design decisions. Derive detailed Use Case diagram, Class diagram & a sequence diagram. Whenever necessary document the relevant assumptions you made.

(09 marks)

Tasks C:
There are many system design patterns available in system development. Critically evaluate singleton, factory and abstract factory design patterns and apply the most suitable design pattern for your system development.

(15 marks)

Tasks D:
Develop an interactive set of interfaces to get the necessary user inputs. Make sure to implement proper validation mechanisms in order to restrict invalid entries to the system. Come up with suitable set of reports, which you will think add more value to your system

i. Your program must be a distributed application with web services

ii. Your program should make use of a proper database to store information

(30 marks)

Tasks E:
Document the test plan and explain how you used test driven development in this scenario and do a test automation to achieve that. This includes test rationale, test plan, test data and proper application of the test plan (LO II)

(20 marks)

Tasks F:
Create user and technical documentation for the developed solution. (10 marks)

Tasks G:
Critically evaluate your ethical considerations in developing the solution as well as justify the tools and function you have used to achieve software carpentry and codemanship.

(10 marks)

Guidelines for the report format

Paper A4 | Margins 1.5" left, 1" right, top and bottom
Page numbers – bottom, right | Line spacing 1.5
Font size
Headings 14pt, Bold | Normal 12pt
Font face- Times New Roman
Referencing and in-text citation should be done strictly using **Harvard Referencing System.**

Marking Scheme

Task (A) contains 6 marks

Criteria	Marks Out of 6
Functional requirement specification	3
Nonfunctional requirement specification	3

Task (B) contains 9 marks
Diagrams should be evaluated according to the following criteria.

Criteria	Marks Out of 9
Proper use of Object Oriented Design Methodology Use case Diagram · Identification of correct use cases · Identification of correct Actors and associations Sequence Diagram · Implementing identified set of use cases (about 3) as sequence diagrams. Class Diagram · Identification of associated methods, with correct signatures and attributes in each class · Correct identification of relationships	1-3
Average Design · Clear identification of private, public access modifiers & it is visible in the class diagram · Accurate use of <<include> <<extend>> stereo types in use case diagram · Appropriate use of lifelines, messages and objects in proposed sequence diagrams · Correct use of UML notations with minor mistakes Evaluation · Student has given basic description about the design and given a reasonable justification · Effective judgements have been made	4-6

Excellent Design · Highly detailed diagram · Use of OO concepts clearly visible · Backed by relevant assumptions · Multiplicity, navigability aggregation & compositions visible in class diagrams · Excellent use of UML notation Evaluation · Good justification of the design · Judge validity of results · Use critical reflection to evaluate the work and justify with valid explanations Fluency (Of design) · Evidence of critical analysis on different perspectives covering how, use case, class & sequence diagrams support in designing	7-9

Task (C) contains 15 marks

Criteria	Marks Out of 15
Identify the different types of design patterns and there advantages	1-5
Apply the suitable design patterns for system development	6-10
Critically evaluate the impact of design patterns	11-15

Task (D) contains 30 marks

Criteria	Marks Out of 30
Pass · Basic data management system features . · Use a database (simple design) · Have simple Web user interface	0-8
Good · Make a good attempt to follow the three tier architecture. · More sophisticated database design and queries · More sophisticated data representation (e.g. several classes at business logic level) · Separate UI windows for entering results and viewing overall scores.	9-20

Excellent · More sophisticated UI, · Complex functionality (Email alerts/SMS/Innovative aspects) · 3- tier architecture should exit · Appropriate use of more sophisticated database features (e.g. use of stored procedures / functions / triggers to implement business rules) · Reports being proposed to facilitate decision making. · Effective use of sessions / cookies	21-30

Task (E) contains **(20 marks)**

Test Rationale **(5 marks)**

- Provide a concise rationale for the approach adopted. Discuss how you are going to use test driven development.

Devise your test data **(5 marks)**

- Derive test data for the system.

Produce and apply a test plan **(10 marks)**

- Create test classes for your system
- You are to carry out relevant tests and provide documentation detailing the tests used to verify your system**.**
- Demonstrate that the code passes all the tests (use screen-grabbing software and insert images into your submission).
- Use of test automation
- Evaluation of overall success or failure and lessons learned.
- Traceability showing how each requirement is met by the design.

Task (F) contains 10 marks

Criteria	Marks Out of 10
Errors in the documentation	0-3
Acceptable standard of documentation with poor explanations	3-5
High standard of documentation with screen shots & average explanations	5-7
Professional standard of documentation with screen shots & good explanation	5-10

Task (G) contains 10 marks

Criteria	Marks Out of 10
· Identify ethical issues relevant to the case study. · Analyze the ethical considerations · Justify the chosen considerations.	3

Justify the tools and functions used to achieve software carpentry in your development.	3
Justify the tools and functions used to achieve software codemanship in your development.	4

Final Grading criteria for the coursework:

Marks	Final Grade
>=70	1
69-60	2:1
59-50	2:2
49-40	3
<40	Fail

****