THE UNIVERSITY OF THE WEST INDIES

Semester I ☒ Semester II ☐ Supplemental/Summer School ☐

Examinations of December ☒ /April/May ☐ /July ☐ 2010

Originating Campus: **Cave Hill** ☐ **Mona** ☒ **St. Augustine** ☐

Mode: **On Campus** ☐ **By Distance** ☐

Course Code and Title: **COMP 2140 Introduction to Software Engineering**

Date: **Wednesday, December 15, 2010** Time: **4:00 PM – 6:00 PM**

Duration: **2 Hours.** **Paper No:**

Materials required:

Answer booklet: **Normal** ☐ **Special** ☒ **Not required** ☐

Calculator: **Programmable** ☐ **Non Programmable** ☐
(where applicable)

Multiple Choice answer sheets: **numerical** ☐ **alphabetical** ☐ **1-20** ☐ **1-100** ☐

Auxiliary/Other material(s) – Please specify:

Candidates are permitted to bring the following items to their desks:

Instructions to Candidates: This paper has 3 pages & 4 questions.

Candidates are reminded that the examiners shall take into account the proper use of the English Language in determining the mark for each response.

Do Question 1 and Any Other Two

Question 1 *General Question [30 marks] [Compulsory Question]*

a) *Software Engineering Ethics*: The IEEE/ACM Software Engineering Code of Ethics and Professional Practice identifies eight principles that software engineers must adhere to. List the eight principles. **[2 marks]**

b) *Software Processes*: Whether we are developing software following a plan-driven or an agile approach, there are certain software process activities that must be carried out.

i. What are the central activities of agile approaches? Why are they central? **[3 marks]**

ii. Rank all the activities in their order of importance for a plan-driven approach. Give reasons for your ordering. **[5 marks]**

c) *Requirements*: Why are different levels of requirements useful? What are these levels? Give an example of a requirement written at the different levels you identify. **[5 marks]**

d) *System Modelling:* Using the UML we are able to construct several models as part of our system design exercise. The following models are common: context model, interaction model, structural model, behavioural model. What UML diagrams are used under each model category? **[4 marks]**

e) *Architectural Design*: Why is it important to explicitly design and document software architecture? **[2 marks]**

f) *Testing*: "The testing process has two distinct goals." What are they? **[2 marks]**

g) *Software Evolution*: Why is it important to pay special attention to software evolution? **[2 marks]**

h) *Software Reuse:* "Reuse-based software engineering is an approach to development that tries to maximise the reuse of existing software artefacts." What opportunities for reuse existed for your coursework project? How did you exploit these opportunities? If you did not exploit them, why not? **[5 marks]**

Question 2 ***Requirements [15 marks]***

Refer to the project that you did for your coursework.

a) List the functional and non-functional requirements for the system **[5 marks]**
b) Draw a use case diagram for the same system **[3 marks]**
c) What is the relationship between the functional requirements and the use case diagram, if any? **[1 mark]**
d) Briefly describe the process for gathering requirements. **[2 marks]**
e) Why is it important to validate requirements? Define three validity checks that a requirements document should undergo. **[4 marks]**

Question 3 ***System Modelling [14 marks]***

Refer to the project that you did for your coursework.

a) List the UML diagrams that you used in the design phase. **[2 marks]**
b) Draw a UML diagram showing the relationships between the important entities in your project. **[5 marks]**
c) Draw a UML diagram showing some entities exchanging messages to effect a desired functionality. **[5 marks]**
d) Explain how you would use these diagrams as the basis for your implementation, i.e how do they relate to the code? **[3 marks]**

Question 4 ***Software Evolution [15 marks]***

Lehman's eight laws of system evolution dynamics are a result of empirical studies carried out on software systems.

a) What are the limitations of Lehman's laws? **[1 mark]**

b) State four of Lehman's laws **[4 marks]**

c) What is a legacy system? **[1 mark]**

d) An organisation has a legacy system and they need to manage it carefully. They need to decide what to do with it. What options do they have? When does each of the options best apply? **[8 marks]**

e) In what sense is software maintenance more than software evolution? **[1 mark]**

END OF QUESTION PAPER