Programme Code: TU856
Shared with: TU858
Module Code: CMPU 2019
CRN: 22390, 26464

TECHNOLOGICAL UNIVERSITY DUBLIN

Grangegorman

TU856 - BSc. Degree in Computer Science

TU858 - BSc. Degree in Computer Science (International)

Year 2

SEMESTER 1 EXAMINATIONS 2023/2024

CMPU2019 Software Engineering 1

Internal Examiner(s):
Mr. Richard Lawlor
Dr. Paul Doyle

External Examiner:
Ms. Pamela O'Brien

Exam Duration: 2 hours

Instructions To Candidates:

Attempt 4 out of 5 questions. All questions carry equal marks.

Special Instructions /Handouts/ Materials Required: None

1. (a) Provide a usecase description for each of the following 3 usecases for a Library system:

- return book
- borrow book
- borrow book and pay fine

and draw a corresponding usecase diagram.

(10 marks)

(b) Describe briefly two other roles usecases may have besides requirements description.

(5 marks)

(c) What is the difference between **generalisation** and **<<extends>>** in linking two related usecases?
In the Unified Process, outline two situations where you might use the **<<extends>>** stereotype to split a usecase.

(10 marks)

2. (a) Explain what is meant by the terms *modularity* and why it is important in programming and software design.

What is meant by encapsulation in object-oriented design and how does it relate to coupling and cohesion? How well is it supported in an object-oriented language you are familiar with?

(10 marks)

(b) How are coupling and cohesion affected by inheritance when using object-oriented programming principles?

(7 marks)

(c) Explain what an interface is and provide two ways of showing one in UML.
Then comment on the meaning of the UML diagram shown in Figure Q2(c). How does this design affect coupling and cohesion?

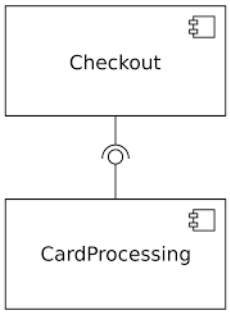


Figure Q2(c)

(8 marks)

3. (a) What is meant by *Requirements Engineering*? Describe the activities in requirements engineering and show them and their outputs in a process diagram.

(10 marks)

(b) Briefly outline two approaches to Prototyping and describe their main purpose, paying particular attention to the one which relates more to requirements.

(10 marks)

(c) Comment on the claim that Agile Methodologies are adaptive rather than predictive.

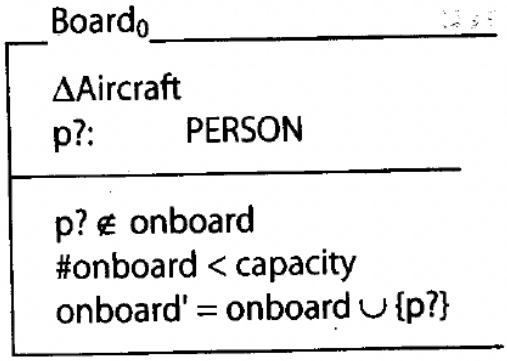
(5 marks)

4. (a) Explain what is meant by Formal Specification and Verification and describe a significant advantage of each. Mention a disadvantage that applies to both.

Describe briefly a situations where this approach to software development would be suitable.

(13 marks)

(b) Given the following diagram which is a specification in Z of the boarding operation for aircraft boarding software, briefly comment on what each line in the specification means. Also express each of the conditions in Object Constraint language (OCL).



(12 marks)

5. (a) Given the class diagram in Figure 1 below, write USE class definitions for the classes ***Order*** and ***OrderLine***. Include a SOIL implementation for the operation ***addLineItem()*** but not for ***calculatePrice()***. Also provide a SOIL implementation for ***processOrderLine(q, p)*** but not for ***getPrice()***.

(15 marks)

(b) Write a class constraint or invariant in OCL for class ***Order*** which says: if an order's customer has a poor credit rating, then the order's ***isPrepaid*** attribute must be ***True***.

(4 marks)

(c) Write an appropriate precondition and postcondition for ***updateStock(q : Integer)*** in class ***Product***.

(6 marks)

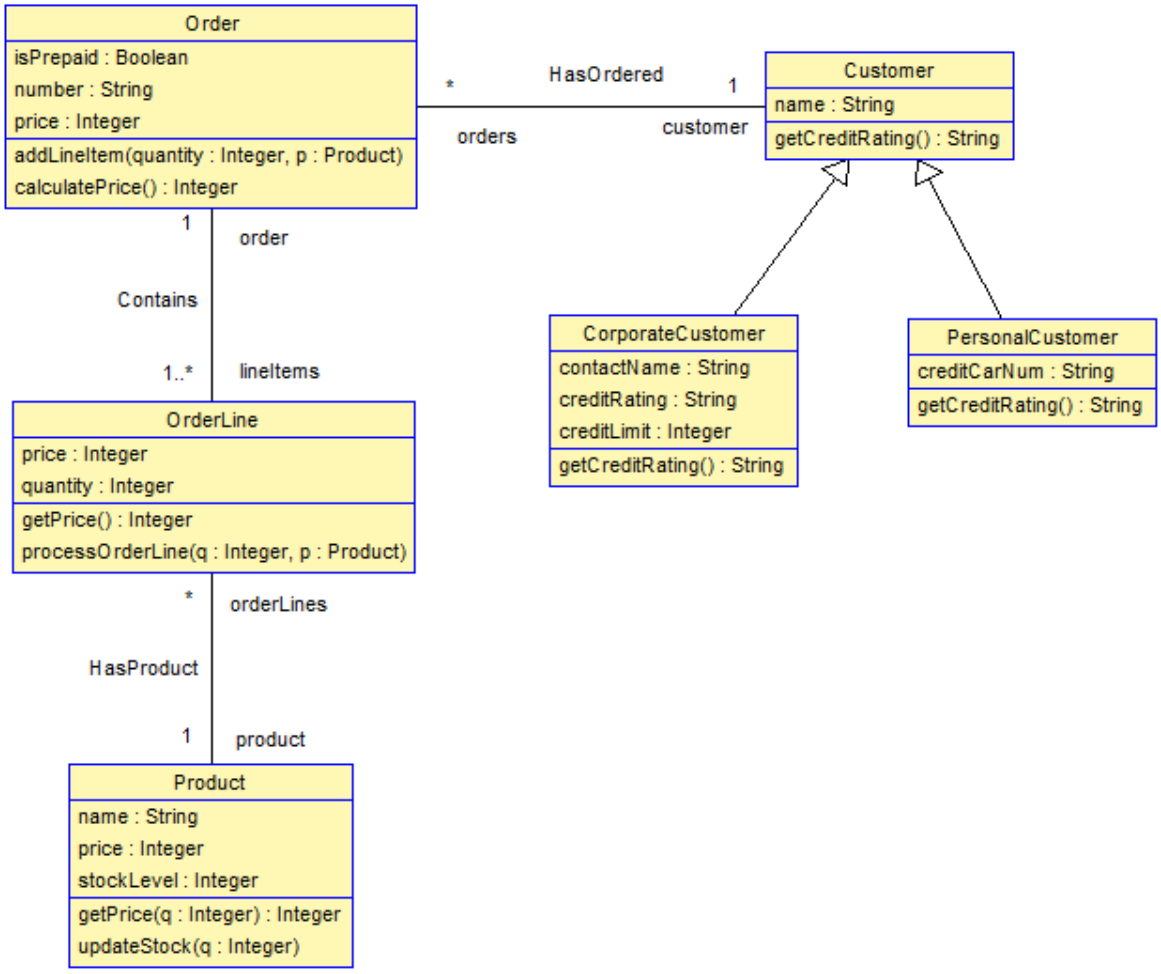


Figure 1