



Faculty of Science

Academy of Computer Science and Software Engineering

Final Summative Assessment
Third Opportunity

Module	IFM03A3 / IFM3A10 Informatics 3A – Introduction to Software Engineering
Campus	APK
Month	August 2020

Date		Time	
Assessor	Mr F F Blauw		
Moderator	Prof AJ van der Merwe (UP)		
Duration	180 minutes	Marks	150

MEMORANDUM

ExtermYGogga

No escape from the fate that you make

Mr Cad Louse enters, looking as mousy as ever, his chihuahua following close in tow.

“Thank you!” Mr Louse squeaks, “You have done an excellent job with our Bug Bibliography. But the time has come to get rid of them! I have started my own extermination business and I need your help!”

“I personally manage our exterminators. We keep strict record of exactly when each of our exterminators are hired and how many critters they have cancelled. Each exterminator has a speciality. Some deal with flying insects while others deal with insects that like to borrow and scamper.

“The insecticide manager – Ms Venin – needs to know exactly what and how many cans of insecticide are currently in storage and how many are currently being used. Depending on how many assignments we get per week, she adjusts her order accordingly.

“Then there is the swatter manager – Mr Clobber – and he must know exactly which swatter is being used. Swatters need to be serviced regularly, otherwise they will fall apart and so will Mr Clobber. Each swat needs to be registered when returning from assignment and a service booked when the swatter has slapped 100 times.

“Exterminators are assigned based on their speciality. When an exterminator goes on assignment, I must know which exterminator is currently on assignment. I also need to know the status of their assignment, as provided by the exterminator. We cannot have too many idle exterminators!

“An exterminator returning from assignment, must log a report, but they also need to return all insecticide cans (full or empty) and swatters to their respective departments. Finally, during the reporting, the exterminator needs to log the bugs that were slain, if any. A record is kept, according to the Bug Bibliography.

“I want to see exactly which bugs can be found in which areas around our service area. If I see an increase, we might need to hire more exterminators.”

“Do you think you can help out, lad?”

Mr Louse squeaks as he leaves. You start questioning your own sanity.

QUESTION 1

- 1.1. Provide a definition for Software Engineering. (3)

The application of a
systematic, disciplined, quantifiable approach [1]
to the
development, operation, and maintenance [1]
of software; that is, the application of engineering to software.
And the **study of approaches [1]** of such.

- 1.2. But, what is software? (3)

Instructions [1] that when executed provide desired features, function, and performance;
Data structures [1] that enable the programs to adequately manipulate information;
And **description information [1]** in both hard copy and virtual forms that describes the operation and use of the programs.

- 1.3. Building software faces new challenges every day. Briefly describe one such a challenge and the best way you believe to overcome this challenge. (4)

Heterogeneity

Increasingly, systems are required to operate as distributed systems across networks that include different types of computer and mobile devices.

Business and social change

Business and society are changing incredibly quickly as emerging economies develop and new technologies become available. They need to be able to change their existing software and to rapidly develop new software.

Security and trust

As software is intertwined with all aspects of our lives, it is essential that we can trust that software.

DESCRIPTION OF CHALLENGE [2]

MEANS TO OVERCOME SUCH CHALLENGE [2]

[10]

QUESTION 2

One of the most difficult tasks software engineer faces, is understanding the requirements. The seven tasks of requirements engineering can be defined as follows:

1. Inception
2. Elicitation
3. Elaboration
4. Negotiation
5. Specification
6. Validation
7. Management

2.1. Which **requirements analysis model** would be the best for the project described by Mr Louse? Motivate your answer by referring to the steps described above. (7)

Scenario-based models
Class-oriented models
Behavioural and Pattern-based models
Data Models
Flow-oriented Models

MOTIVATION SHOULD MATCH PROCESS [3]

MOTIVATION SHOULD INCLUDE CASE STUDY [4]

2.2. Provide a concise **problem statement** for the ExtermyGogga. (4)

Relevance to Case Study [2]

Description of source Problem, not solution [2]

2.3. Provide a brief **proposed solution** for the ExtermyGogga. (4)

Relevance to Case Study [2]

Brevity/Cohesiveness of solution [2]

[15]

QUESTION 3

- 3.1. Name one **non-functional requirement** that specifically applies to the ExtermyGogga. Explain how you believe it can be applied and how it can then be measured. *NOTE: Do not use general non-functional requirements such as: security, reliability, availability, etc.* (5)

Relevance to Case Study [1]

Application to Solution/Functional Requirements [2]

Applicable Measurement [2]

- 3.2. List *all* the **functional requirements** that you can extract from Mr Louse's description of ExtermyGogga. *NOTE: The mark allocation does not indicate the number of functional requirements.* (6)

Phrased as Functional Requirements (Stories) and NOT Use Cases

Each use case should be relevant from the case study.

- 3.3. Draw a **use case diagram** illustrating the use cases derived from the requirements identified in **Question 3.12**. (20)

Use Cases MUST match Functional Requirements from previous question

	Weak	Fair	Accomplished
Applicable Actors	0-1	2-3	4
Applicable Subsystems	0	1	2-3
Applicable Use Cases	0-3	4-6	7-9
Diagram Correctness	0-1	2-3	4

Applicable refers to Case Study

- 3.4. If you could **alter** the functional requirements for ExtermyGogga, what would you add or change, and why? (4)

Student's Answer. Should NOT changes business processes!

[35]

QUESTION 4

Draw an **activity diagram** for a use case entitled: “*Complete Assignment*”.

[15]

If it does not contain INFORMATION SYSTEM subsystems/swimlanes = 0

	Weak	Fair	Accomplished
Basic Activity Elements (Start, End, Arrows)	0	1	2
Swimlanes	0	1	2-3
Appropriate Actions	0-1	2	3-5
Logical Flow	0-1	2	3-5

QUESTION 5

Consider the following C# code. Draw an **Interaction Sequence Diagram** to model the code. You may assume that Mrs Bush clicked on **ClickCheckOwner** on the GUI.

```

01: class Program
02: {
03:     static void Main()
04:     {
05:         bool ground = // value from GUI
06:         Exterminator me = new Exterminator();
07:         Assignment myAssignment = new Assignment(me);
08:         myAssignment.GetTransport(ground);
09:         myAssignment.Execute();
10:     }
11: }
12:
13: class Assignment
14: {
15:     private Exterminator _mainExterminator;
16:     private Vehicle missionTransport;
17:     public Assignment (Exterminator mainExterminator)
18:     {
19:         _mainExterminator = mainExterminator;
20:     }
21:
22:     public void GetTransport(bool groundMission)
23:     {
24:         missionTransport = new Vehicle(groundMission);
25:     }
26:
27:     public void Execute() {
28:         mainExterminator.Dispatch(missionTransport);
29:     }
30: }
31:
32: class Exterminator
33: {
34:     public void Dispatch(Vehicle myTransport) {
35:         // Off on an assignment
36:         myTransport.Go();
37:     }
38: }
39:
40: class Vehicle
41: {
42:     private bool _needsDigger;
43:     public Vehicle(bool groundMission) {
44:         _needsDigger = groundMission;
45:     }
46:     public void Go() {
47:         if (_needsDigger)
48:             System.Console.WriteLine("Dig dig dig.");
49:         else
50:             System.Console.WriteLine("Swat swat swat.");
51:     }
52: }

```

[20]

	Weak	Fair	Accomplished
Classes from Code	0-1	2	3-4
Appropriate Messages/ Method Calls	0-1	2-4	5-6
Logical Flow	0-2	3-5	5-8
Diagram Correctness	0	1	2

QUESTION 6

- 6.1. Which **software development process** do you think will be best for the development of ExtermyGogga? Motivate your decision based on ExtermyGogga. (3)

Waterfall

Evolutionary

Iterative

Any Agile-based process

MOTIVATION SHOULD MATCH PROCESS [3]

- 6.2. Provide a brief description of the **operation** of the software development process you selected in Question 6.1. You may use a diagram to aid you. (7)

Description of give an overview

	Weak	Fair	Accomplished
Diagram	0	1	2-3
Description	0-1	2-3	3-4
Description*	0-1	2-4	5-7

* If no diagram, full [7] marks can be awarded to description

[10]

QUESTION 7

7.1. What is a **software architecture**? (3)

The software architecture of a program or computing system is the **structure or structures of the system [1]**, which comprise **software components [1]**, the **externally visible properties of those components [1]**, and the **relationships [1]** among them

7.2. Why is it important to decide on a software architecture for a particular project? (3)

Analyze effectiveness
 Consider architectural alternatives
 Reduce the risks
 Communication among all stakeholders
 Early design decisions
 Model of how the system is structured

7.3. When starting an architectural design, you are faced with many possible architectural styles. What are the two factors you need to consider when choosing an appropriate style? Briefly describe each. (2)

CONTROL

How is control managed? How do components transfer control?

DATA

How is data communicated? Is the flow continuous? What is the mode of transfer?
 How do functional components interact with data components?

7.4. Considering ExtermyGogga, which **generic architectural style** will you base your design on? Motivate your decision. (2)

Layered
 Data Centred
 Object Oriented
 NOT: Data flow, Main/Subsystem
 Motivation must make sense in terms of the Case Study

7.5. Use a diagram to describe the architectural style you selected in **Question 7.4** while referring to ExtermyGogga in the diagram. (10)

Description of give an overview

	Weak	Fair	Accomplished
Diagram*	0-1	2	3-4
Description*	0-1	2-3	3-6

* If diagram does not contain components from case study, WEAK only, full ACCOMPLISHED can still be awarded to description

[20]

QUESTION 8

- 8.1. Choose any one (1) design pattern that could be applied to ExtermyGogga. Describe this design pattern and the reason you chose it. (5)

	Weak	Fair	Accomplished
Description	0	1	2
Motivation	0	1	2-3

*

- 8.2. Draw a conceptual class diagram of the **business domain** and **process components** for ExtermyGogga. (15)

	Weak	Fair	Accomplished
Applicable Business Domain Components	0-1	2-4	5-6
Applicable Process Components	0-1	2-3	4-5
Logical Connection between Classes	0-1	2-3	4

*

[20]**QUESTION 9**

Considering that ExtermyGogga deals with highly sensitive data, discuss measures you will put in place to secure this data from unauthorised access.

[10]