



UNIVERSITY OF COLOMBO, SRI LANKA

UNIVERSITY OF COLOMBO SCHOOL OF COMPUTING

DEGREE OF BACHELOR OF INFORMATION TECHNOLOGY (EXTERNAL)

Academic Year 2012/2013 – 2nd Year Examination – Semester 4

**IT4304: Rapid Software Development
PART 2 - Structured Question Paper**

21st July, 2013
(ONE HOUR)

To be completed by the candidate

BIT Examination Index No:

Important Instructions:

- The duration of the paper is **1 (one) hour**.
- The medium of instruction and questions is English.
- This paper has **3 questions** and **8 pages**.
- **Answer all questions.** All questions **do not** carry similar marks.
- **Write your answers** in English using the space provided **in this question paper**.
- Do not tear off any part of this answer book.
- Under no circumstances may this book, used or unused, be removed from the Examination Hall by a candidate.
- Note that questions appear on both sides of the paper.
If a page is not printed, please inform the supervisor immediately.

Questions Answered

Indicate by a cross (x), (e.g.) the numbers of the questions answered.

| To be completed by the candidate by marking a cross (x). | Question numbers | | |
|--|------------------|---|---|
| | 1 | 2 | 3 |
| To be completed by the examiners: | | | |
| | | | |
| | | | |

1) (a) Briefly describe what “Agile Software Development” is? (10 marks)

ANSWER IN THIS BOX

Answer

Agile software development is a group of software methodologies which apply

- Time boxed iterative and evolutionary development
- Adaptive planning
- Promote evolutionary delivery
- Values and practices that encourage agility (rapid and flexible response to change)

(b) Provide 5 examples for early implementations of “Agile Software Development”? (5 marks)

Answer

- Scrum
- Adaptive Software Development
- Feature Driven Development
- Dynamic Systems Development Method (DSDM)
- Crystal Clear
- Extreme Programming (XP)

- c) List and briefly (max. of two lines) explain 6 characteristics of the Agile Process. (12 marks)

ANSWER IN THIS BOX**Answer****Modularity:**

Modularity allows a process to be broken down into activities.
In agile software process, activities are used like a good tool.

Iterative:

Agile software processes focus on short cycles
Within each cycle, a certain set of activities is completed
The short cycle is repeated many times to refine the deliverables

Time-Bound:

Each iteration is set to time limits (between one and six weeks)
If the iteration cannot be completed within the allocated time period,
functionality may be reduced or activities may be rescheduled

Parsimony:

Requiring a minimal number of activities necessary to mitigate risks and achieve their goals
This allows software developers to deliver systems against an aggressive schedule, while maintaining some semblance of a normal life

Adaptive:

If the goal cannot be achieved using the planned activities during the iteration,
new activities can be added

Incremental:

Partition the nontrivial system into increments which may be developed in parallel, at different times, and at different rates

Convergent:

Doing everything within the power to ensure success in the most rapid fashion

People-Oriented:

Agile process supports people over process and technology

Collaborative:

Agile processes encourage communication among team members.
When a system is developed in incremental manner, collaboration are requested to understand how to integrate them

- (d) What are the four main facts stated in the “Agile Manifesto”? (8 marks)

ANSWER IN THIS BOX

Answer

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

2.

- (a) What is meant by “Software Prototyping” in the context of software engineering? (3 marks)

ANSWER IN THIS BOX

Software prototyping: An approach to software development that uses prototypes to help the developers and customers to visualize the proposed system early in the Software process.

- (b) When prototyping is used in rapid application development, it is often said to be iterative until done. What is meant by “iterative until done” in the point of views of the developers, designers? (6marks)

ANSWER IN THIS BOX

Answer

- Developers build/refine prototype based on current requirements
- Designers review the prototype
- Customers try out the prototype and refine their requirements
- This cycle is repeated many times until the full system is built

- (c) Provide 5 advantages of “Software Prototyping” over traditional software development methodologies. (10 marks)

ANSWER IN THIS BOX

Answer

- Good communication and better understandings among developers and customers
- Prototypes can provide early training for users of the system
- Incomplete or inconsistent requirements can be identified
- Prototyping can provide a product that is best fit for customer requirements
- Prototyping may produce some useful deliverables even though project runs out of time or money
- Prototypes may demonstrate progress at an early stage of the development
- Prototyping may less effort than conventional development
- The cost required for redesigning, time required for testing and cost required for initial maintenance are reduced due to the early detection of the problem

- (d) Name an Integrated Development Environment (IDE) that you have used for rapid application development and briefly explain how it facilitated your work. (6 marks)

ANSWER IN THIS BOX

Answer

- Net-Beans, Eclipse, Visual Studio, etc... and it facilitated by providing:
- A database programming language interface
- An interface generator (Drag-and-drop interface creating)
- Links to office applications such as MS Excel for analysis of numeric information
- Facility to create and integrate reporting modules such as “crystal reports”
- Visual programming tool which converts design diagrams to software code

- (e) You are asked to develop an automated system for operating and recording the output of X-Ray machine in a hospital. Your program needs to feed its output to existing hospital management system as well. Is rapid application development (RAD) suitable for the above project? Justify your answer. (5 marks)

ANSWER IN THIS BOX

Answer

No.
 Rapid application development is only good when reliability is not critical and the application is not interacting with the existing systems. An application which operates the X-Ray machine is critical. That kind of critical work should be well analyzed and designed prior to development. Since the system is critical, there is no chance for testing at use and fix any errors. The system should be built with no errors when delivered and prior to delivery, all testing need to be completed. Because of all these reasons, RAD is not suitable for the above requirement

- (3)
 a) What are the characteristics of Scrum Development? Provide 5 of them? (10 marks)

ANSWER IN THIS BOX

Answer

- Self-directed and self-organizing team
- No external addition of work to an iteration, once chosen
- Daily stand-up meeting with special question
- Usually 30-calendar day iteration
- Demo to external stakeholders at end of each iteration
- Each iteration, client-driven adaptive planning

- (b) “Requirements Gold Plating” and “Developer gold plating” are two classic mistakes in software engineering. What do they mean? (6 Marks)

ANSWER IN THIS BOX

Answer

Requirements Gold Plating

Some projects can have more requirements than they needs.

Developer gold plating

Developers are fascinated by new technology and sometimes keen to try out new features of a language or environment, etc... irrespective of the real need.

- (c) What is “Function Point (FP) Analysis / Estimation” in the context of rapid application development? (4 marks)

ANSWER IN THIS BOX

Answer

FP is an algorithmic approach to determine the size of the software project.

Use the following FP Multiplication Table to answer the questions below.

| Program Characteristic | Functional Point | | |
|--------------------------|------------------|-------------------|-----------------|
| | Low Complexity | Medium Complexity | High Complexity |
| Number of Inputs | ×3 | ×4 | ×6 |
| Number of Outputs | ×4 | ×5 | ×7 |
| Inquiries | ×3 | ×4 | ×6 |
| Logical Internal Files | ×7 | ×10 | ×15 |
| External Interface Files | ×5 | ×7 | ×10 |

A medium complexity software project is having 8 inputs, 5 outputs, 2 inquires, 3 logical internal files and 6 external interface files. The influence multiplier of this project is 1.12.

(d)

Calculate the Function Point Total for the above software project

(10 marks)

ANSWER IN THIS BOX

Answer

FP Tot = Inputs (8 x 4) + outputs (5 x 5) + inquiries (2 x 4) + logical internal files (3 x 10) + external interface files (6 x 7) = 32 + 25 + 8 + 30 + 42 = 137

(e)

Calculate the Adjusted Function Point Total for the above software project.

(5 marks)

ANSWER IN THIS BOX

Answer

Adjusted FP Total = 1.12 * 137 = 153.44