1100CST309122102

FINAL SCHEME

Total Pages: 3

Scheme of Valuation/Answer Key

(Scheme of evaluation (marks in brackets) and answers of problems/key)

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

FIFTH SEMESTER B.TECH DEGREE EXAMINATION, MARCH **2022**

Course Code: CST 309

Course Name: MANAGEMENT OF SOFTWARE SYSTEMS

Max. Marks: 100 Duration: 3 Hours

PART A

(Answer all questions; each question carries 3 marks)

		Marks
1	Any three factors -1 mark each	3
2	Well executed Agile software development methodology helps teams significantly improve the quality of their software at each release. Not only that, it allows teams to adapt to change quickly. The Agile process consists of short, time-boxed iterations known as sprints. Each sprint results in a working product.	3
3	For each heading and explanation 1/2 mark each	3
4	For each comparison 1 mark each	3
5	• The main difference between GPL and LGPL is that GPL provides more protection to the software users. It allows them the freedom to make changes to the software, share and receive source code. • When a user distributes the software, one must make sure others can get the same rights. It is important to note that any changes made in the software must also be licensed under GPL. • LPGL, on the other hand, is specially designated for software libraries, wherein one can make changes and give back source codes, but one can link it to a non-free program wherein it is not licensed under GPL. Most programs today are licensed under GPL while most libraries use GPL, some opts to use LGPL so that more people are allowed to use benefit from it. For each point 1 mark each	3
6	For each comparison 1 mark each	3
7	List and explain any three activities – 1 marks each	3
8	Backlogs – 1.5 marks	3

		Sprints- 1.5 marks	
9		Explaining each characteristic 1 mark each	3
10		Definition – 3 marks	3

PART B

(Answer one complete questions from each module)

Module -1

11	a)	Figure- 3 marks, Explanation- 3 marks, Importance- 1 mark.	7
	b)	Differentiating agile software development and traditional software development-4 marks, case study- 3 marks	7
12	a)	Demonstrating waterfall and incremental model with figure 2.5 marks each, justification- 2 marks	7
	b)	Pair programming- 3.5 marks, Refactoring-3.5 marks	7

Module -2

13	a)	Requirement engineering process – 1 mark 3 activities -2 marks each	7
	b)	Concept of traceability matrix – 3.5 marks Requirements management planning -3.5 marks	7
14	a)	Use case- 2 marks Use case diagram for an ATM with explanation- 5 marks	7
	b)	Explaining Personas- 2.5 marks Scenarios- 2.5 marks Feature identification-2 marks	7

Module -3

15	a)	Design patterns-3 marks Essential elements of design patterns-4 marks	7
	b)	Formal and informal review techniques- 3.5 marks each	7
16	a)	Top Down Integration testing method with Diagram- 3.5 marks Bottom Up Integration testing method with Diagram- 3.5 marks	7
	b)	System Testing- 1 mark Explanation of any 4 system testing types 1.5 marks each	7

1100CST309122102

FINAL SCHEME

Module -4

17	a)	Explaining plan driven development -3.5 marks project scheduling-3.5 marks	7
	b)	Explanation of software risk management steps-5 marks Figure-2 marks	7
18	a)	Explaining algorithmic cost modeling-4 marks Problems-3 marks	7
	b)	Critical path definition-3 marks Significance with sample project-4 marks	7

Module -5

19	a)	CMMI- 3.5 marks ISO 9001:2000- 3.5 marks	7
	b)	Software Quality- 1 mark 1. Software engineering methods 2. Project management techniques 3. Quality control actions 4. Software quality assurance For the above activities- 1.5 marks each	7
20	a)	Explaining elements of Software Quality Assurance- 3.5 marks SQA Tasks- 3.5 marks	7
	b)	Illustrating SPI process in detail- 7 marks	7

****