CMR INSTITUTE OF TECHNOLOGY	USN	

Internal Assesment Test – I, November 2019

Sub:	SOFTWARE ENGINEERING							Code:	18MCA14
Date:	18.11.2019	Duration:	90 mins	Max Marks:	50	Sem:	I	Branch:	MCA

		Answer **ONE FULL QUESTION** from each part	Marks	OBE	
				CO	RBT
		Part-I			
1	(a)	What is software process model? Explain the essential attributes of good software	[6]	CO1	L2
	(b)	Describe the difference between software engineering and system engineering product	[4]	CO1	L3
		(OR)			
2		Discuss the various issues in software engineering professional responsibility and code of ethics	[10]	CO1	L2
		Part-II			
3	(a)	Discuss the key challenges facing software engineering professionals that are common to all software processes	[4]	CO2	L2
	(b)	Explain the phases of RUP with a neat diagram.	[6]	CO2	L2
		(OR)			
4		Discuss the process involved in waterfall and incremental development model with advantages and disadvantages.	[10]	CO2	L2

CMR INSTITUTE OF TECHNOLOGY	USN	

Internal Assesment Test – I, November 2019

Sub:	SOFTWARE ENGINEERING							Code:	18MCA14
Date:	18.11.2019	Duration:	90 mins	Max Marks:	50	Sem:	I	Branch:	MCA

		Answer **ONE FULL QUESTION** from each part	Marks	OBE	
				CO	RBT
		Part-I			
1	(a)	What is software process model? Explain the essential attributes of good software	[6]	CO1	L2
	(b)	Describe the difference between software engineering and system engineering product	[4]	CO1	L3
		(OR)			
2		Discuss the various issues in software engineering professional responsibility and code of ethics	[10]	CO1	L2
		Part-II			
3	(a)	Discuss the key challenges facing software engineering professionals that are common to all software processes	[4]	CO2	L2
	(b)	Explain the phases of RUP with a neat diagram.	[6]	CO2	L2
		(OR)			
4		Discuss the process involved in waterfall and incremental development model with advantages and disadvantages.	[10]	CO2	L2

Part-III

5	(a)	What is meant by pair programming? Mention the advantages of pair programming	[5]	CO2	L2
	(b)	What is scrum? Discuss the various phases involved in scrum process.	[5]	CO2	L2
		(OR)			
6	(a)	Explain the principles of agile methods.	[6]	CO2	L2
	(b)	Extreme programming expresses user requirements as stories, with each story written on a card. Discuss the advantages and disadvantages of this approach to requirements description.	[4]	CO2	L3

Part-IV

7	Explain in detail about extreme programming.	[10]	CO2	L2
	(OR)			
8	Explain with neat diagram, the various activities in requirement engineering process.	[10]	CO1	L2

Part-V

9	(a)	Define: (i) Functional requirements and (ii) Non-functional requirements	[4]	CO1	L1
	(b)	Explain the different metrics of non-functional requirements.	[6]	CO1	L2
		(OR)			
10	(a)	What is requirement specification? Explain various ways of writing system requirements	[6]	CO3	L2
	(b)	Justify which process model will be suitable to design insulin pump system	[4]	CO3	L2

Part-III

5	(a)	What is meant by pair programming? Mention the advantages of pair programming	[5]	CO2	L2
	(b)	What is scrum? Discuss the various phases involved in scrum process.	[5]	CO2	L2
		(OR)			
6	(a)	Explain the principles of agile methods.	[6]	CO2	L2
	(b)	Extreme programming expresses user requirements as stories, with each story written on a card. Discuss the advantages and disadvantages of this approach to requirements description.	[4]	CO2	L3

Part-IV

7	Explain in detail about extreme programming.	[10]	CO2	L2
	(OR)			
8	Explain with neat diagram, the various activities in requirement engineering process.	[10]	CO1	L2

Part-V

9	(a)	Define: (i) Functional requirements and (ii) Non-functional requirements	[4]	CO1	L1
	(b)	Explain the different metrics of non-functional requirements.	[6]	CO1	L2
		(OR)			
10	(a)	What is requirement specification? Explain various ways of writing system requirements	[6]	CO3	L2
	(b)	Justify which process model will be suitable to design insulin pump system	[4]	CO3	L2

CMR INSTITUTE OF TECHNOLOGY

Department of Computer Applications-PG Studies

First Internal Assessment Test- Scheme and Solution

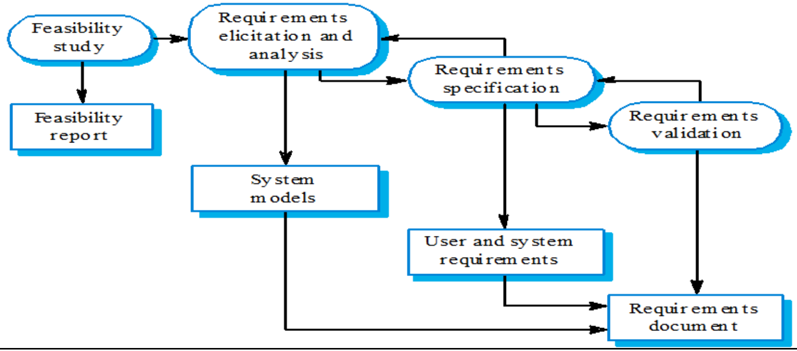
18MCA14-Software Engineering

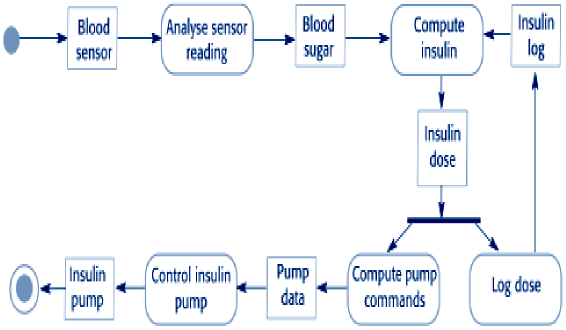
Questions Number	**Expected Answer**	**Marks Break up**	**Maximum Marks**
1 a	What is software process model? Explain the essential attributes of good software A software process model is an ***<u>abstract representation of a process</u>***. It presents a description of a process from some particular perspective Maintainability, Dependability, Efficiency, Acceptability, Usability, Reliability, Scalability, Portability, Reusability	2 any four can be explain 4x1=4	6
1 b	Describe the difference between software engineering and system engineering product The difference between System Engineering and Software Engineering is given below: However, it can be said that the System Engineers focus more on users and domains, while Software Engineering focus more on n implementing quality software. They typically work with QA and hardware engineers to develop testing plans. Any one Example	2 2	4
2	Discuss the various issues in software engineering professional responsibility and code of ethics. *Issues in software engineering professional responsibility* Confidentiality ,Competence, Intellectual property rights Computer misuse *Software engineers shall adhere to the following Eight Principles:* Public , Client And Employer ,Product Judgment ,Management Profession , Colleagues ,Self Learning	Any two points 2 8x1=8	10
3 a	Discuss the key challenges facing software engineering professionals that are common to all software processes. Heterogeneity, delivery and trust with example	4x1=4	4
3b	Explain the phases of RUP with a neat diagram. • Inception • Elaboration • Construction • Transition	4x1=4 2x1=2	6

	Diagrammatic explanation with Example		
4	Discuss the process involved in waterfall and incremental development model with advantages and disadvantages. • There are separate identified phases in the waterfall model: – Requirements analysis and definition – System and software design – Implementation and unit testing – Integration and system testing – Operation and maintenance The incremental build model is a method of software development where the model is designed, implemented and tested incrementally until the product is finished. It involves both development and maintenance. The product is defined as finished when it satisfies all of its requirements. This model combines the elements of the waterfall model with the iterative philosophy of prototyping. Advantages: Easy to understand, easy to use, especially by inexperienced staff Generates working software quickly and early during the software life cycle. Disadvantages: All requirements must be known upfront – in most projects requirement change occurs after project start Each phase of an iteration is rigid and do not overlap each other.	3 3 2 2	10
5a	What is meant by pair programming? Mention the advantages of pair programming. Pair programming is an agile software development technique in which two programmers work together at one workstation. One, the driver, writes code while the other, the observer or navigator, reviews each line of code as it is typed in. The two programmers switch roles frequently. Pair Programming Advantages • Two heads are better than one. • More efficient. • Fewer coding mistakes. • An effective way to share knowledge.	1 4x1=4	5

	• Develops your staff's interpersonal skills.		
5b	What is scrum? Discuss the various phases involved in scrum process. Scrum is a framework within which people can address complex adaptive problems, while productively and creatively delivering products of the highest possible value. Scrum processes address the specific activities and flow of a Scrum project. In total there are 19 processes in SBOK™ Guide which are grouped into following five phases: **Initiate** - This phase includes the processes related to initiation of a project: Create Project Vision, Identify Scrum Master and Stakeholder(s), Form Scrum Team, Develop Epic(s), Create Prioritized Product Backlog, and Conduct Release Planning. **Plan and Estimate** -This phase consists of processes related to planning and estimating tasks, which include Create User Stories, Approve, Estimate, and Commit User Stories, Create Tasks, Estimate Tasks, and Create Sprint Backlog. **Implement** - This phase is related to the execution of the tasks and activities to create a project's product. These activities include creating the various deliverables, conducting Daily Standup Meetings, and grooming (i.e., reviewing, fine-tuning, and regularly updating) the Product Backlog at regular intervals. **Review and Retrospect** - This phase is concerned with reviewing the deliverables and the work that has been done and determining ways to improve the practices and methods used to do project work. **Release** - This phase emphasizes on delivering the Accepted Deliverables to the customer and identifying, documenting, and internalizing the lessons learned during the project.	1 Any four 4x1=4	5
6 a	Explain the principles of agile methods. The key principles, and how Agile Development fundamentally differs from a more traditional Waterfall approach to software development, are as follows: • Active user involvement is imperative • The team must be empowered to make decisions • Requirements evolve but the timescale is fixed • Capture requirements at a high level; lightweight & visual • Develop small, incremental releases and iterate • Focus on frequent delivery of products • Complete each feature before moving on to the next • Apply the 80/20 rule	6	6

6b	• Testing is integrated throughout the project lifecycle – test early and often • A collaborative & cooperative approach between all stakeholders is essential Extreme programming expresses user requirements as stories, with each story written on a card. Discuss the advantages and disadvantages of this approach to requirements description. The primary advantage of a user story is that it is short and concise and it expresses the requirement from the perspective of what the user business need is rather than attempting to tell a developer how to satisfy that need. The disadvantage (if it is a disadvantage) is that it relies on direct communications to further elaborate the details of the requirement and some developers may not be very skilled in doing that.	2 2	4
7	Explain in detail about extreme programming XP is a lightweight, efficient, low-risk, flexible, predictable, scientific, and fun way to develop a software. e**X**treme **P**rogramming (XP) was conceived and developed to address the specific needs of software development by small teams in the face of vague and changing requirements. Extreme Programming is one of the Agile software development methodologies. It provides values and principles to guide the team behavior. The team is expected to self-organize. Extreme Programming provides specific core practices where − • Each practice is simple and self-complete. • Combination of practices produces more complex and emergent behavior. Embrace Change A key assumption of Extreme Programming is that the cost of changing a program can be held mostly constant over time. This can be achieved with − • Emphasis on continuous feedback from the customer • Short iterations • Design and redesign • Coding and testing frequently • Eliminating defects early, thus reducing costs • Keeping the customer involved throughout the development • Delivering working product to the customer	10	10

8	Explain with neat diagram, the various activities in requirement engineering process  Feasibility study Requirements elicitation and analysis Requirements specification Requirements validation Feasibility report System models User and system requirements Requirements document	10	10
9 a 9b	Define: (i) Functional requirements and (ii) Non-functional requirements Functional requirements are those which are related to the technical functionality of the system. Non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system in particular conditions, rather than specific behaviors. • Time ○ Transactions / sec ○ Response time ○ Time to complete an operation • Space ○ Main memory ○ Auxiliary memory ○ (Cache) • Usability ○ Training time ○ Number of choices ○ Mouse clicks • Reliability ○ Mean time to failure ○ Downtime probability ○ Failure rate ○ Availability • Robustness ○ Time to recovery ○ % of incidents leading to catastrophic failures ○ Data corruption probability after a failure • Portability ○ % of non-portable code	2+2 6	10

	○ Number of systems where software can run		
10 a	What is requirement specification? Explain various ways of writing system requirements **Requirements Specification** It's the process of writing down the user and system requirements into a document. The requirements should be clear, easy to understand, complete and consistent. Several standards organizations (including the IEEE) have identified nine topics that must be addressed when designing and writing an SRS: 1. Interfaces 2. Functional Capabilities 3. Performance Levels 4. Data Structures/Elements 5. Safety 6. Reliability 7. Security/Privacy 8. Quality 9. Constraints and Limitations	6	6
10b	Justify which process model will be suitable to design insulin pump system Insulin pumps are small, computerized devices that mimic the way the human pancreas works by delivering small doses of short acting insulin continuously (basal rate). ... You program the amount of insulin for your mealtime bolus directly on the pump.  Blood sensor Analyse sensor reading Blood sugar Compute insulin Insulin log Insulin dose Log dose Compute pump commands Pump data Control insulin pump Insulin pump	4	4