Follow Us

@dbatu_btech_maharashtra

https://www.instagram.com/invites/contact/?i=14ecv66nyj1tv&utm_content=mohp4lb

DBATU B.Tech,Maharashtra

https://t.me/+HusDh0a5RqVlMGE1

DBATU B.Tech,Maharashtra

Computer Science & Engineering

Software Engineering

Question Bank

Unit I

1. What is software engineering? Briefly discuss the need for software engineering.

2. What are attributes of good software? Explain the key challenges facing software engineering

3. What is software engineering? Explain software engineering code of ethics.

4. List and explain ant four software engineering code of ethics.

5. What is ethics doing in a course for software engineers? Explain with the help of case study.

6. How do you solve a case study business ethics? Explain with example.

7. Explain Ethical case study on conflict of interest.

8. With a neat diagram explain the activity model of insulin pump control system.

9. Explain model for Mental Health Care-Patient Management System

10. Design a model to help monitor climate change and to i…

19. With the help of diagram illustrate spiral model and state two benefits and the problems in spiral development.

20. Explain in detail Concurrent Development Model

21. Give some benefits of using V model?

22. What is the iterative model of the SDLC? What are its advantages and disadvantages?

23. Why Reuse-oriented software engineering is in demand? Explain in detail.

24. What are four process activities which are common to all software process? Explain it in detail.

25. How can software projects manage change?

26. What is prototyping?

27. What is incremental delivery/development?

28. What is coping with change in software engineering?

29. With a neat diagram explain Boehm's spiral model

30. With neat diagram explain The Rational Unified Process.

Unit II

1. State Agile manifesto.

2. Explain agility and cost of change.

3. What are the Principles of Agile developments?

4. Explain plan driven development with a neat block diagram.

5. Explain plan-based development and agile development

6. State XP practice principles.

7. What are XP values?

8. How XP process helps in project development?

9. Write short note on

A. pair programming

B. Refactoring

C. Testing in XP

10. What are industrial XP practices? Explain.

11. What is Agile Project Management?

12. Explain Scrum Process Flow with neat diagram. 13. Explain Scrum life cycle with neat diagram.

14. What is Scrum Sprint?

15. Explain product backlog in Scrum

16. What is the role of scrum master in Scrum?

17. What is scrum burn down chart?

18. Explain the activities performed in daily scrum meeting

19. Explain: Sprint Review and Retrospective

20. How the agile methods are scaled? State the coping of agile methods for large system engineering.

21. What is requirement engineering? Distinguish functional and non-functional requirements.

22. Explain the structure of requirement document.

23. Explain different ways of writing system requirements specification.

24. With a neat diagram explain the different stages of requirement engineering process. What are its benefits?

25. Explain Requirement elicitation and analysis process.

26. Write short note on interviewing.

27. Explain Ethnography in detail.

28. How do you validate requirements?

29. What is requirements management? Why it is important?

30. What are the steps in requirements change management process?

Unit III

1. Explain Context models,

2. Explain interaction model.

3. Explain Context models.

4. Draw Use case diagram for ATM system.

5. Draw Sequence diagrams for ATM system.

6. Explain Model-Driven Engineering (MDE).

7. Draw state transiton diagram for ATM system.

8. Draw the software architecture of an ATM system.

9. Explain Layered information system architecture.

10. Explain Client-Server Architecture.

11. Explain in detail.

(a) Layered Architecture.

(b) Repository Architecture

(c) Pipe and Filter Architecture

(d) Client-Server Architecture

Unit IV

1. Assume that the Library management system is being developed using an object-oriented approach. Draw a use case diagram showing at least six possible use cases for this system.

2. Using the UML graphical notation for object classes, design the following object classes, identifying attributes and operations. Use your own experience to decide on the attributes and operations that should be associated with these objects.

A messaging system on a mobile (cell) phone or tablet.

A printer for a personal computer.

A personal music system.

A bank account.

A library catalogue.

3. Develop the design of the Hospital Management system to show the interaction between the data collection subsystem and the instruments that collect patient data. Use sequence diagrams to show this Interaction.

4. Identify possible objects in the following systems and develop an object-oriented design for them. You may make any reasonable assumptions about the systems when deriving the design. A group diary and time management system is intended to support the timetabling of meetings and appointments across a group of co-workers. When an appointment is to be

made that involves a number of people, the system finds a common slot in each of their diaries and arranges the appointment for that time. If no common slots are available, it interacts with the user to rearrange his or her personal diary to make room for the appointment

5. A filling station (gas station) is to be set up for fully automated operation. Drivers swipe their credit card through a reader connected to the pump; the card is verified by communication with a credit company computer, and a fuel limit is established. The driver may then take the fuel required. When fuel delivery is complete and the pump hose is returned to its holster, the driver's credit card account is debited with the cost of the fuel taken. The credit card is returned after debiting. If the card is invalid, the pump returns it before fuel is dispensed.

6. Draw a sequence diagram showing the interactions of objects in a group diary system when a group of people are arranging a meeting.

7. Draw a UML state diagram showing the possible state changes in either the group diary or the filling station system.

8. When code is integrated into a larger system, problems may surface. Explain how Configuration management can be useful when handling such problems.

9. A small company has developed a specialized software product that it configures specially for each customer. New customers usually have specific requirements to be incorporated into their system, and they pay for these to be developed and integrated with the product. The software company has an opportunity to bid for a new contract, which would more than double its customer base. The new customer wishes to have some involvement in the configuration of the system. Explain why, in these circumstances, it might be a good idea for the company owning the software to make it open source.

Unit V

1. What is software testing? Why it is important?

2. What is the difference between validation and verification?

3. Bring out the differences between testing and inspection.

4. Explain software inspection with its advantages.

5. Explain with neat diagram model of the software testing process

6. Explain the different levels in Development testing.

7. Write short note on

(a) Unit testing

(b) component testing

(c) System Testing

8. Explain unit testing with example.

9. Explain how to choose unit test cases?

10. Explain interface testing with suitable diagram.

11. Explain System testing with example.

12. Explain Test driven development with neat diagram. Also mention its advantages.

13. What is the release testing? What are the differences between release testing and system testing?

14. What are the benefits of involving users in release testing at an early stage in the testing process? Are there disadvantages in user involvement?

15. Explain with example Requirements-based testing. Mention the issues resolved during it.

16. Write a scenario that could be used to help design tests for the wilderness weather station system.

17. What is performance testing? Give examples of performance test cases.

18. Explain different types of user testing.

19. What is acceptance testing? Why it is important?

20. Compare alpha- beta testing?

21. With the help of neat diagram explain six stages of Acceptance testing process.

22. Explain Dependability properties.

23. Explain

(a) Availability

(b) Reliability

(c) Safety

(d) Security

24. Explain dependability properties.

25. Explain Verification and validation processes.

26. Explain in detail.

(a) Fault Avoidance

(b) Fault Detection and Correction

(c) Fault Tolerance.

27. Explain Reliability Measurement.

28. Explain Safety Terminology.

29. Explain Safety Cases.

30. Explain Design Guidelines.