

Hall Ticket No

Question Paper Code: AITB26



INSTITUTE OF AERONAUTICAL ENGINEERING

(Autonomous)

Dundigal, Hyderabad - 500 043

MODEL QUESTION PAPER

B.Tech IV Semester End Examinations (Regular), April / May – 2020

Regulations: IARE-R18

SOFTWARE ENGINEERING
(INFORMATION TECHNOLOGY)

Time: 3 hours

Max. Marks: 70

Answer ONE Question from each Unit

All Questions Carry Equal Marks

All parts of the question must be answered in one place only

UNIT – I

1. a) Discuss the need for the process models. Explain in detail the risk management phase in spiral with the help of a neat diagram. [7M]
- b) Differentiate between perspective and specialized process models. Discuss in detail about function point based estimation with a suitable example. [7M]
2. a) Elaborate the concept of earned value analysis with effect to project scheduling. Discuss the importance of project scheduling with its merits. [7M]
- b) What is project management? List and explain the principles related to software project management related to a project. [7M]

UNIT – II

3. a) Define software requirement. Explain the importance of software requirements document in a project with suitable example. [7M]
- b) Explain software requirements engineering process. Outline the importance of feasibility study in generating the feasibility report for an assigned project. [7M]
4. a) Discuss various non-functional requirements with suitable examples. Differentiate between functional and non functional requirements. [7M]
- b) Define cohesion and coupling and discuss different levels of coupling. Explain the importance of petri nets in software development. [7M]

UNIT – III

5. a) Discuss various design concepts related to software design. Explain in detail about designing class based components. [7M]
- b) Explain about user interface design. Discuss various architectural styles and architectural patterns with suitable example. [7M]
6. a) Define software architecture? Discuss about architectural design and architectural mapping using data flow in design process. [7M]
- b) Explain the importance of traditional components and discuss data centered architectural style with suitable example. [7M]

UNIT – IV

7. a) Define testing and give the importance of testing. List the fundamentals of a software testing process. [7M]
- b) Distinguish between testing and debugging process. Explain basis path testing and control structure testing with the help of an example. [7M]

8. a) Discuss different software implementation techniques like coding practices and refactoring of software testing. [7M]
- b) Differentiate between internal and external views of testing. Discuss the role of system testing and debugging for software. [7M]

UNIT – V

9. a) Define risk management. Discuss about various types of Software Risks identified during software development. [7M]
- b) Compare and contrast between reactive risks and proactive risks with suitable example. Discuss the need for risk identification. [7M]
10. a) Define cost estimation. Discuss the importance of constructive cost estimation model II under project estimation. [7M]
- b) Discuss the relationship between people and effort. Explain the need for considering various project metrics with an example. [7M]



INSTITUTE OF AERONAUTICAL ENGINEERING

(Autonomous)

COURSE OBJECTIVES:

The course should enable the students to:

I.	Learn how to elicitate requirements and develop software life cycles.
II.	Understand the design considerations for enterprise integration and deployment.
III.	Analyze quality assurance techniques and testing methodologies.
IV.	Understand implementation issues such as modularity and coding standards.
V.	Prepare a project plan for a software project that includes estimates of size and effort, a schedule, resource allocation, configuration control, and project risk.

COURSE OUTCOMES:

CO 1	Understand the concept of process models associated under requirements development process.
CO 2	Explore the importance of requirements analysis and preparation of a system requirements document, for any software development.
CO 3	Understand the basic design principles and mapping data flow into architecture.
CO 4	Explore the benefits of different testing strategies.
CO 5	Understand the concept of risk management and the relation between people and effort.

COURSE LEARNING OUTCOMES:

Students, who complete the course, will have demonstrated the asking to do the following:

AITB26.01	Understand the key concerns that are common to all software development processes.
AITB26.02	Identify the appropriate process models, approaches and techniques to manage a given software development process.
AITB26.03	Identify the approach to risks management through risk identification, risk measurement and risk mitigation.
AITB26.04	Use the concept of Earned Value Analysis (EVA) to measure the projects progress at any given point in time, forecasting its completion date and final cost, and analyzing variances in the schedule and budget as the project proceeds.
AITB26.05	Memorize project planning activities that accurately help in selection and initiation of individual projects and of portfolios of projects in the enterprise.
AITB26.06	Identify dependability and security issues that affect a given software product.
AITB26.07	Use the concept of classical analysis to determine the acceptance criteria as part of specification.
AITB26.08	Memorize the importance of eliciting the requirements for a software product and translate these into a documented design.
AITB26.09	Understand the concept of data dictionary in order to manage the details in large-scale systems, to locate errors and omissions in the system.
AITB26.10	Understand the concept of petri nets that exhibit concurrency, synchronization and used as a visual communication aid to model the system behavior.
AITB26.11	Memorize the design of object oriented software using with the aid of a formal system modelling notation.
AITB26.12	Learn to model the structure and behavior of a software system.
AITB26.13	Memorize different architectural styles, patterns and architectural mapping using data flow.
AITB26.14	Understand the principles of graphical user interface design.
AITB26.15	Understand the concept of component-level design used to define interface characteristics and communication mechanisms for each software component identified in the architectural design.
AITB26.16	Understand the importance of testing with the performance of root cause analysis.
AITB26.17	Memorize the concepts of software testing approaches such as unit testing and integration testing.
AITB26.18	Understand the approaches to verification and validation including static analysis and reviews.
AITB26.19	Identify the major differences between white box testing and black box testing.

AITB26.20	Understand the importance of refactoring which improves the performance of non functional attributes of the software.
AITB26.21	Learn to manage time, processes and resources effectively by prioritizing competing demands to achieve personal and team goals.
AITB26.22	Understand the concept of risk management through risk identification, risk measurement and mitigation.
AITB26.23	Memorize the relationship between people and effort.
AITB26.24	Identify the importance of earned value analysis related to project scheduling.
AITB26.25	Use a proactive, structured risk assessment and analysis activity to identify and analyze root causes.

MAPPING OF SEMESTER END EXAMINATION TO COURSE LEARNING OUTCOMES:

SEE Question Number		COURSE LEARNING OUTCOME	Course Outcomes	Blooms Taxonomy Level	
1	a	AITB26.01	Understand the key concerns that are common to all software development processes.	CO 1	Understand
	b	AITB26.02	Identify the appropriate process models, approaches and techniques to manage a given software development process.	CO 1	Remember
2	a	AITB26.03	Identify the approach to risks management through risk identification, risk measurement and risk mitigation.	CO 1	Remember
	b	AITB26.04	Use the concept of Earned Value Analysis (EVA) to measure the projects progress at any given point in time, forecasting its	CO 1	Understand
3	a	AITB26.08	Memorize project planning activities that accurately help in selection and initiation of individual projects and of portfolios of	CO 2	Remember
	b	AITB26.07	Use the concept of classical analysis to determine the acceptance criteria as part of specification.	CO 2	Remember
4	a	AITB26.08	Memorize the importance of eliciting the requirements for a software product and translate these into a documented design.	CO 2	Understand
	b	AITB26.10	Understand the concept of petri nets that exhibit concurrency, synchronization and used as a visual communication aid to	CO 2	Remember
5	a	AITB26.11	Memorize the design of object oriented software using with the aid of a formal system modelling notation.	CO 3	Remember
	b	AITB26.14	Understand the principles of graphical user interface design.	CO 3	Understand
6	a	AITB26.15	Understand the concept of component-level design used to define interface characteristics and communication mechanisms for each software component identified in the architectural	CO 3	Remember
	b	AITB26.13	Memorize different architectural styles, patterns and architectural mapping using data flow.	CO 3	Understand
7	a	AITB26.16	Understand the importance of testing with the performance of root cause analysis.	CO 4	Remember
	b	AITB26.17	Memorize the concepts of software testing approaches such as unit testing and integration testing.	CO 4	Remember
8	a	AITB26.20	Understand the importance of refactoring which improves the performance of non functional attributes of the software.	CO 4	Understand
	b	AITB26.18	Understand the approaches to verification and validation including static analysis and reviews.	CO 4	Understand
9	a	AITB26.22	Understand the concept of risk management through risk identification, risk measurement and mitigation.	CO 5	Understand
	b	AITB26.25	Use a proactive, structured risk assessment and analysis activity to identify and analyze root causes.	CO 5	Remember
10	a	AITB26.21	Learn to manage time, processes and resources effectively by prioritizing competing demands to	CO 5	Understand
	b	AITB26.23	Memorize the relationship between people and effort.	CO 5	Remember

Signature of Course Coordinator

HOD, IT