BCA -08

Software Engineering

UNIT-1: Introduction to Software Engineering

Introduction, The Evolving Role of Software, What is Software Engineering?: Definition. Software Characteristics, Software Crisis, Software myths: Management myths, Customer Myths, Practitioner's myths. Qualities of Software Product, Principals of Software Engineering: Separation of Concerns, Modularity, Abstraction, Anticipation of Change, Generality, Incremental Development, Consistency.

Unit-2: Software Process Models

Introduction: The planning process, The development process, A Software process. Linear Sequential Model or Waterfall Model: Requirements analysis Phase, Specification phase, System and software design phase, Implementation and testing phase, Integration and system testing phase, Maintenance phase. Prototype Model: Version I, Version II. RAD Model: Development Methodology, RAD Model Phases: Business Modeling, Data Modeling, Process Modeling, Application Generation, Testing and Turn over, Incremental Model.

UNIT-3: Software Process Evolutionary Models

The incremental Model, The Spiral Model, The Component Assembly Model, The Concurrent Development Model, The Formal Methods Model, Fourth Generation Techniques models, Comparison of Software Process Models.

UNIT-4: Software Project Management

Structure of unit

Software management activities: Project planning activities, Project Cost Estimation, Problems in Software projects, Project Estimation. Project Management Approaches, Process based unified process, Project development stages or project management life cycle, Project control systems.

UNIT-5: Software Project Metrics and Software Project Planning

Software Measurement: Size-oriented Metrics, Function –oriented Metrics. Project Planning: Objective, Software Scope, Resources like Human Resources, Reusable Resources and environmental Resources.

UNIT-6: Software Project Estimation

Decomposition Techniques: Software Sizing, Problem Based Estimation, Process Based Estimation, Cocomo – An Empirical Estimation Model For Effort, Risk Management.

UNIT-7: Software Requirement Specification

The 'Software Requirement': Functional Requirement, Non-Functional Requirements, Domain Requirement, User Requirement, System Requirement. Feasibility Study, Requirement Engineering

Process: Requirement Elicitation, Requirement Specification, Requirement Validation, Key Challenges In Eliciting Requirement, Software Requirement Specification (srs).

UNIT-8: Structured Analysis

Various Elements of the Analysis Model, Data Modeling: Data Object, Attributes and Relationship, Cardinality and Modality, Entity- Relationship Diagram. Functional Modeling: Data Flow Diagram and Context Diagram, Logical and Physical DFDs, Leveling of DFD s, Various Tools of Structured Analysis: Data Flow Diagram, Data Dictionary, Structured English, Decision Table**.**

UNIT-9: Software Design

Design Concepts and Principles: Design Specification Model, Design Guidelines, Design Principles, Fundamental Software Design Concepts. Tools used for Design and Analysis, Modularity: Cohesion, Coupling, Functional Independence, Function- Oriented design, Object-Oriented design: Function-oriented vs. Object- oriented Design approach, Real Time Design: Design Process of real time system, User Interface Design: User Interface Design Issues, User Interface Design Principles, User Interface Design Process, User Analysis, User Interface Prototyping, Interfaced Evaluation.

UNIT-10: Software Quality Assurance

Introduction, Quality Concepts, Software Reliability, SQA Plan, ISO 9000 Quality Standards.

UNIT-11: Testing Fundamentals

Testing Objectives, Testing Principles, Test Case Design, White-Box testing: Basis Path Testing, Control Structure Testing, Condition testing, Data Flow Testing, Lop Testing. Black Box Testing.

UNIT-12: Software testing strategies

Verification and Validation Activities, Software Inspection, Verification and Formal Methods, Unit Testing, Integration testing Validation testing, System testing: Recovery Testing, Security testing, Stress testing, Performance testing.

UNIT-13: Software Re-Engineering

Why Software Reengineering?, Advantages of Software re-engineering, Disadvantages of Software re-engineering, Software Reengineering Process Model. Software Reverse Engineering: Characteristics of Reverse Engineering. Restructuring: Types of Restructuring, Forward Engineering.

UNIT-14:CASE (Computer Aided Software Engineering)

Nee of CASE, Objective of case: Improve Productivity, Pmprove Information System Quality, Improve Effectiveness, Organizations Reject CASE Because. Disadvantages of Software Re- engineering: Benefits of Case Tools, Levels of Case, Lists of CASE Tools: Taxonomy of CASE Tools. Features of CASE Tools: Categories of CASE Tools: Vertical CASE Tools, Horizontal CASE Tools. Need of CASE Tools. Architecture of CASE Environments: USER Interface, Object- Management System (OMS), Repository.