|  |  |
| --- | --- |
| **SESSION** | **SEPT 2024** |
| **PROGRAM** | **BACHLEOR OF COMPUTER APPLICATIONS (BCA)** |
| **SEMESTER** | **V** |
| **COURSE CODE & NAME** | **DCA3103 SOFTWARE ENGINEERING** |
|  |  |
|  |  |

**Set-I**

**1. State the pros & cons of different software development models.**

**Ans 1.**

**Pros and Cons of Different Software Development Models**

Software development models provide structured frameworks for planning, designing, implementing, and managing software projects. Various models cater to specific project requirements, and each has its strengths and weaknesses. The prominent models include the Waterfall Model, Agile Model, Spiral Model, and V-Model.

**Waterfall Model**

The **Waterfall Model** is one of the earliest and simplest development approaches. It follows a sequential, phase-wise progression: requirements, design, implementation, testing, and

Its Half solved only

Buy Complete assignment from us

**Price – 190/ assignment**

**MUJ Manipal University Complete SolvedAssignments session JULY-AUG 2024**

buy cheap assignment help online from us easily

we are here to help you with the best and cheap help

**Contact No – 8791514139 (WhatsApp)**

**OR**

**Mail us-** **bestassignment247@gmail.com**

**Our website -** [**www.assignmentsupport.in**](http://www.assignmentsupport.in)

**2.i. Explain the various guidelines for data design.**

**ii. Discuss various functions of architectural design.**

**Ans 2.**

**i. Explain the Various Guidelines for Data Design**

Data design is a crucial aspect of software development, ensuring that data structures and storage mechanisms meet the needs of the software system. Effective data design involves creating a structure that supports the system's requirements and optimizes performance, accessibility, and integrity. The guidelines for data design focus on achieving efficiency, consistency, and scalability.

One primary guideline is data normalization, which minimizes data redundancy and ensures that

**3.i. Explain different software reliability metrics.**

**ii. How reliability in a software system can be achieved? Explain**

**Ans 3.**

**i. Explain Different Software Reliability Metrics**

Software reliability metrics are quantitative measures used to assess and predict the reliability of a software system. Reliability, in this context, refers to the likelihood that software will perform its intended functions correctly without failure over a specified period. These metrics help developers and testers identify issues and enhance software quality.

One of the most common metrics is the Mean Time to Failure (MTTF). MTTF measures the average time between two consecutive software failures. A higher MTTF value indicates better software reliability, as the software functions longer without encountering issues. Another critical

**Set-II**

**4.i. List the various characteristics of software testing.**

**ii. Write a short note on**

**a. White Box Testing**

**b. Black Box Testing**

**Ans 4.**

**i. List the Various Characteristics of Software Testing**

Software testing is a crucial process in the software development life cycle, ensuring that the system functions as intended while meeting quality and reliability standards. The key characteristics of software testing revolve around thoroughness, accuracy, and efficiency.

One characteristic of software testing is completeness, which ensures that all features and functionalities are thoroughly tested under different conditions. Testing verifies that the software meets the specified requirements without any functional gaps. Another important characteristic is

**5. Define Software maintenance and explicate its various tasks.**

**Ans 5.**

**Software Maintenance and Explicate its Various Tasks**

Software maintenance refers to the process of modifying, updating, and enhancing software applications after their deployment to correct errors, improve performance, and ensure they continue to meet evolving user requirements. It is an essential phase of the software development life cycle, as software systems need to adapt to technological, business, and environmental changes. Maintenance ensures the software remains functional, efficient, and reliable throughout

**6.i. Briefly explain the Process of Agile Software Development.**

**ii. Differentiate traditional Software Engineering and Modern Engineering.**

**Ans 6.**

**i. Briefly Explain the Process of Agile Software Development**

Agile software development is an iterative and incremental process that emphasizes collaboration, flexibility, and customer feedback. It aims to deliver high-quality software in shorter cycles, accommodating changing requirements and ensuring continuous improvement. The Agile process is guided by principles outlined in the Agile Manifesto, which values individuals, working software, customer collaboration, and responsiveness to change.

The Agile development process begins with **requirements gathering** and creating a product backlog, a prioritized list of features or tasks that need to be implemented. Unlike traditional