Course Code	**:**	**BCS-051**
Course Title	**:**	**Introduction to Software Engineering**
Assignment Number	**:**	**BCA(V)051/Assignment/2026-27**
Maximum Marks	**:**	**100**
Weightage	**:**	**25%**
Last Date of Submission	**:**	**31st October, 2026 (For July 2026 Session)**
	:	**30th April, 2027 (For January 2027 Session)**

Note: This assignment has eight questions carrying 10 marks each for a total of 80 marks. Answer all questions. The remaining 20 marks are for Viva Voce. Draw suitable diagrams wherever necessary.

Q1:

(a) Explain the characteristics of a **good Software Requirements Specification (SRS)**. Discuss any **five characteristics** recommended for preparing an effective SRS document. **(5 Marks)**

(b) A university intends to develop an **Online Examination Management System**. Identify **five functional requirements** and **five non-functional requirements** that should be included in its SRS. **(5 Marks)**

Q2:

An organization plans to automate its **Library Management System**.

(a) Draw a **Context Diagram** showing the interaction between the system and external entities. **(4 Marks)**

(b) Develop a **Level-1 Data Flow Diagram (DFD)** consisting of at least **four major processes**. **(6 Marks)**

Q3:

A company wishes to develop an **Online Grocery Delivery System** using an object-oriented approach.

(a) Identify the major classes, their attributes and operations. **(5 Marks)**

(b) Draw a **Use Case Diagram** showing the interaction among Customer, Delivery Executive and Administrator. **(5 Marks)**

Q4:

A software company estimates that a project consists of **480 Function Points**.

The productivity of the development team is **12 Function Points per Person-Month**.

The project is expected to be completed in **8 months**.

(a) Calculate the estimated effort in person-months. **(3 Marks)**

(b) Estimate the required team size. **(3 Marks)**

(c) Discuss any **four advantages** of Function Point Analysis over LOC-based estimation. **(4 Marks)**

Q5:

(a) Explain the concept of **Cohesion** and **Coupling**. Illustrate different types of Cohesion and Coupling with suitable examples. **(5 Marks)**

(b) Why are **High Cohesion** and **Low Coupling** considered desirable in software design? **(5 Marks)**

Q6:

An e-Commerce company has completed the development of its shopping portal.

(a) Prepare **four Black Box Test Cases** for the Login Module. **(5 Marks)**

(b) Differentiate between: **(5 Marks)**

- Verification and Validation
- Alpha Testing and Beta Testing

with suitable examples.

Q7:

(a) Describe the major activities involved in **Software Project Management.** **(5 Marks)**

(b) Explain how **Risk Management** contributes to the successful completion of software projects. Illustrate the software risk management process with a suitable diagram. **(5 Marks)**

Q8:

(a) Compare any **five Software Development Life Cycle (SDLC) Models** based on: **(5 Marks)**

- Requirement Stability
- Risk Handling
- Customer Participation
- Cost
- Suitable Applications

(b) Discuss the role of the following in Software Engineering: **(5 Marks)**

- Software Configuration Management (SCM)
- Software Maintenance
- Software Quality Assurance (SQA)

Explain how these activities collectively improve software quality.

Q1:

(a) Explain the characteristics of a good Software Requirements Specification (SRS). Discuss any five characteristics recommended for preparing an effective SRS document. (5 Marks)

(b) A university intends to develop an Online Examination Management System. Identify five functional requirements and five non-functional requirements that should be included in its SRS.
(5 Marks)

Answer

(a) Characteristics of a good SRS

A Software Requirements Specification is a formal description of what a software system is expected to do and the constraints under which it must operate. It becomes a common reference for users, analysts, designers, developers and testers. A good SRS describes required behaviour clearly without prematurely prescribing the internal implementation.

Five important characteristics are:

- Correct: Every stated requirement should represent an actual need of the customer or user. Requirements that are not needed, or that contradict the real business process, should not appear in the document. Correctness is normally checked through reviews with stakeholders.
- Unambiguous: Each requirement should have only one reasonable interpretation. Vague words such as "fast", "user-friendly" or "adequate" should be replaced by measurable statements. For example, "the result page shall open within three seconds for 95% of requests" is clearer than "the result page shall open quickly".
- Complete: The SRS should cover all significant inputs, outputs, functions, interfaces, error conditions, constraints and responses to abnormal situations. It should also define what happens when information is missing or an operation fails.
- Consistent: No requirement should conflict with another requirement. The same term, data item or rule should have the same meaning throughout the SRS. For