(8)

T&E. INFT SEM V C-scheme June 2026

Time: 3 Hours **Total Marks: 80**

10/06/26.

Instructions:

(1) Questions 1 is Compulsory.
(2) Attempt any three questions from remaining.
(3) Assume suitable data wherever necessary.

Q1 Attempt any four. **20**

a) Explain the generic software process framework and describe how it supports different prescriptive process models.
b) Distinguish between functional and non-functional requirements with suitable examples for an AI-based Public Health Monitoring System.
c) Explain the LOC and FP estimation techniques and state their merits & limitations.
d) Describe the importance of design concepts such as abstraction, modularity, architecture, and refactoring.
e) What is Software Configuration Management (SCM)? Explain any two SCM activities.
f) Explain unit testing and integration testing for Web-based distributed applications.

Q2 a) Describe the Evolutionary, Incremental, and Agile process models. Compare them based on adaptability, risk handling, and customer involvement. **10**

b) Explain the characteristics and uses of Architectural Design Patterns such as Layered Architecture, MVC, and Client-Server. **10**

Q3 a) Describe in detail the requirements engineering process. **10**

b) Explain forward engineering, reverse engineering, restructuring, and software reengineering with examples from legacy e-governance systems. **10**

Q4 a) Explain the COCOMO II estimation model with a complete example. Discuss its advantages in modern large-scale software development. **10**

b) Describe the Formal Technical Review (FTR) process. Why is FTR important for quality assurance and risk mitigation? **10**

Q5 a) Explain the User Interface Design Process, golden rules of design, and evaluation criteria for a Smart City Traffic Management Application. **10**

b) Describe Earned Value Analysis (EVA) with a numerical example. Explain SPI, CPI, CV, and SV. **10**

Q6) a) What is software risk? Explain risk identification, assessment, projection, and RMMM planning, with an example from digital payment software. **10**

b) Explain software testing strategies for WebApps, including unit, integration, system, acceptance, validation, and regression testing. **10**

****