

## ***BC 205 SOFTWARE ENGINEERING***

### **Course Description and Objectives:**

This course will be helpful for the student to understand the concept of a software life cycle, the role of process models and how to apply key elements and common methods for elicitation and analysis to produce a set of software requirements. To distinguish between different types and levels of testing (unit, integration, systems, and acceptance).

**Course Outcomes:** After completing the course attendees will be able to:

- appreciate the wider engineering issues that form the background to developing complex, evolving (software-intensive) systems;
- plan a software engineering process to account for quality issues and non-functional requirements;
- Employ a selection of concepts and techniques to complete a small-scale analysis and design project.
- Interact with a client to elicit input, and communicate progress.
- Employ group working skills - including general organization, planning and time management, and inter-group negotiation, etc.

### **UNIT-1 Introduction:**

Introduction to SE, evolving the rules of flow changing there of se, myths Software process: A layered technology, process framework, CMMI process assessment

### **UNIT-2 Process models**

Process models, waterfall incremental models, evolutionary models, unified process models.

### **UNIT-3 Software Engineering Practice:**

SE practice, communication practice, planning practices, modeling practices, construction practice, deployment requirements engineering. Tasks, initiating it developing use cases. Building analysis model.

### **UNIT-4 Data Modeling:**

Data modeling concepts, O-O analysis, and scenario based modeling, flow-oriented modeling, and class based modeling, types of architecture design concepts, U-I design golden rules.

### **UNIT-5 Testing Techniques**

Testing strategies: test strategies for conventional S/W testing strategies for od s/w, validation testing, system testing, art of debugging testing, tactics – black box while box, basis path testing, control structures testing, blackbox testing BVA out methods.

**TEXT BOOKS**

Roger S. Pressman "Software Engineering, A practitioner's Approach", 6<sup>th</sup> ed., McGraw-Hill International Edition, 2008.

**REFERENCE BOOKS**

1. Sommerville "Software Engineering", 7th ed., Pearson education, 2008.
2. Shely Cashman Rosenblatt," Systems Analysis and Design" 1<sup>st</sup> ed., Thomson Publications, 2006.