Course Objectives: To apply principles of software development and evolution. To specify, abstract, verify, validate, plan, develop and manage large software and learn emerging trends in software engineering.


Software Processes: Software process models (Waterfall, Incremental, and Evolutionary process models and Agile), software quality concepts, process improvement, software process capability maturity models, Personal Software process and Team Software Process, Overview of Agile Process.


Software Design: System design principles: levels of abstraction (architectural and detailed design), separation of concerns, information hiding, coupling and cohesion, Structured design (top-down functional decomposition), object-oriented design, event driven design, component-level design, data-structured centered, aspect oriented design, function oriented, service oriented, Design patterns.


Software Quality Assurance: Software Quality Control and Quality Assurance, ISO 9000 Certification for Software Industry, SEI CMM and Comparison of ISO & SEICMM.

Software Evolution: Software development in the context of large, pre-existing code bases, Software evolution, Characteristics of maintainable software, Software Reengineering, Software reuse.

Formal Methods: Role of formal specification and analysis techniques in the software development cycle. Formal approaches to Software Modeling and Analysis.


Laboratory Work: Implementation of Software Engineering concepts using tools like
Rational Suite etc. Exposure to CASE tools like Rational Software suit, Turbo Analyst, Silk Suite.

**Recommended Books:**