

COURSE FILE
For
Software Engineering (ECSE205L)

Faculty Name : Dr. Anurag Goswami
Dr. Rupak Chakraborty

Course Type : Foundation

Semester and Year: 4th Semester and 2nd Year

L-T-P : 3-0-2

Credits : 4

Department : Computer Science Engineering

Course Level : UG

SCHOOL OF ENGINEERING AND APPLIED SCIENCES

Department of Computer Science Engineering



Bennett University
Greater Noida, Uttar Pradesh
ECSE105L: Computational Thinking and Programming

Course Type:	Foundation
--------------	------------

L	T	P	Credits
3	0	2	4

Pre-requisites: NA

Course Learning Outcomes:

CLO1: Demonstrate understanding of Software Engineering as an iterative and systematic process.

CLO2: Apply development life cycle through the version control system, IDE, UML, etc.

CLO3: Design the software development process to complement the technical understanding of software products.

Module 1 (Contact hours: 8)

Importance of Software Engineering, Discipline of Software Engineering Lifecycle, Engineering standards in building, testing, operation and maintenance of the computer and software systems.
Software Process Model Introduction, Waterfall Process, Spiral Process, Evolutionary Prototyping Process, Agile Process, Choosing a Model, Lifecycle Documents, Version Control System: Introduction to Git, Git Demo: Git + Eclipse, Git

Module 2 (Contact hours: 6)

Requirements Engineering: General RE Definition, Functional and Non-functional Requirements, User and System Requirements, Modelling Requirements, Analysing Requirements, Requirements Prioritization, Requirements Engineering Process and steps, Creating SRS, Requirements Inspections
OO Software and UML: Object Orientation Introduction, UML Structural Diagrams: Class Diagrams, Component Diagram, UML Structural Diagram: Deployment Diagram., UML Behavioural Diagram: Use Case, Use Case Diagram: Creation Tips, UML Behavioural Diagrams: Sequence, UML Behavioural Diagrams: State Transition Diagram

Module 3 (Contact hours: 7)

Agile Development Methods: Cost of Change, Agile Software Development, Extreme Programming (XP), XP's Values and Principles, Test First Development, Refactoring, Pair Programming, Continuous Integration, Testing Strategy, High-Level Scrum Process
Unified Software Process: Use-Case Driven, Inception Phase, Elaboration Phase, Construction Phase, Transition Phase, Phases and Iterations
Software Evolution: Evolution processes, Legacy Systems, Software Maintenance. Situations during software evolution and maintenance.
Software Reengineering and Refactoring: Reasons to Reengineer and Refactor, Advantages, Refactoring Demo, Refactoring Risks, Cost of Refactoring, When Not to Refactor.

Module 4 (Contact hours: 7)

Software Architecture: What is Software Architecture? Advantages and use of architectural models. Architectural patterns. Different architectural patterns (Layered, MVC, Repository, Pipe and Filter). Design Patterns: Patterns Catalogue, Pattern Format, Factory Method Pattern, Strategy Pattern, Choosing a Pattern, Negative Design Patterns
Software Testing: Black Box Testing Failure, Fault and Error, Verification Approaches, Pros and Cons of Approaches, Testing Introduction, Alpha and Beta Testing, Black-Box Testing, Systematic Functional Testing Approach, Test Data Selection, Equivalence Partitioning and Boundary Value Analysis, Create and Evaluate Test Case Specifications,

Generate Test Cases from Test Case Specifications, White-Box Testing: Coverage Criteria
Intro, Statement Coverage, Control Flow Graphs, Test Criteria

Lab Experiments

The course will cover labs in the following areas:

Getting used to eclipse and GitHub, hands-on session on software design via Visual Paradigm,
Designing and inspecting SRS, software testing using Junit approach.

Suggested Textbooks:

- 1) *Pressman R., Software Engineering, A Practitioner's Approach, McGraw Hill International (2014).*
- 2) *Sommerville I., Software Engineering 10 edition, Person Publications Publishing Company (2015).*

References:

- 1) *Software Engineering Essentials*
<https://www.edx.org/course/software-engineering-essentials-tumx-seecx>
- 2) *Software Engineering: Introduction*
<https://www.edx.org/course/software-engineering-introduction-ubcx-softenglx>
- 3) *Udemy for Visual Paradigm*
<https://www.udemy.com/visual-paradigm-essential/>
- 4) *Official Visual Paradigm training*
<https://www.visual-paradigm.com/training/visual-paradigm-essential/>
- 5) *Tutorials Point for Design Pattern*
https://www.tutorialspoint.com/design_pattern/

Evaluation Component:

Components of Course Evaluation	Percentage
Mid Term Examination	20
End Term Examination	30
Lab Continuous Evaluation	15
Project	20
Quiz	15