

[illegible]

3 is High, 2 is Moderate, 1 is Low & - is Not Applicable

Tentative Lecture Plan

Units	Syllabus Details	Hours required to complete	Course Outcome
1	Overview of Software Engineering What is Software Engineering, why it is required to study and learn as a part of Computer Science and Engineering course, Concepts about Software, SE activities	1	CO1
2	What is well Engineered Software, Software Engineering key challenges. Reasons for Software project failure, Software Engineering Approach, ETVX approach	1	CO1
3	Software Project milestones, Software project deliverables, Issues of professional responsibility	1	CO1
4	Software Development Lifecycle Software Development Life Cycle (SDLC) Waterfall model, prototyping and interactive enhancement.	1	CO1
5	Software Development Lifecycle Prototyping and interactive enhancement.	1	CO1, CO2
6	Iterative and Spiral Model	1	CO1, CO2
7	RAD and Agile process models and Test Driven Development.	1	CO1, CO2
8	FDD and TDD	1	CO1, CO2
9	Requirement Phase Requirement Elicitation, Problem Analysis	1	CO1, CO2
10	Feasibility study of requirements Operational, Technical and Economic.	1	CO1, CO2
11	Software Requirement Specification and Validation Decision Table	1	CO1, CO2
12	Software Requirement Specification and Validation Requirements Prioritization.	1	CO1, CO2
13	Decision Tree	1	CO1, CO2
14	Planning of Software Project: Process Planning, Estimation, COCOMO Model, Project Scheduling and Staffing	1	CO1, CO2, CO3
15	Practise on Planning of Software Project	1	CO1, CO2, CO3
16	Design Phase Differences between analysis and design activities	1	CO1, CO2, CO3
17	Problem partitioning	1	CO1, CO2, CO3
18	Problem partitioning, abstraction, top down and bottom up design.	1	CO1, CO2, CO3
19	Desirable characteristics of good software	1	CO1, CO2, CO3
20	Data flow diagram, Structured approach	1	CO2, CO3
21	Data flow diagram Functional versus object oriented approach	1	CO2, CO3
22	Design Specification	1	CO2, CO3
23	Coding standards and coding guidelines	1	CO1, CO2, CO3
24	Coding structured programming, programming style, code reviews and inspections and documentation	1	CO2, CO3
25	Code Inspections	1	CO2, CO3
26	Testing different types and principles of software testing	1	CO2, CO3, CO4
27	Levels of testing, functional testing	1	CO2, CO3, CO4
28	Structural testing, test plan	1	CO2, CO3, CO4
29	Test cases specification, reliability assessment	1	CO2, CO3, CO4
30	Design test cases unit test cases, system integration test cases	1	CO2, CO3, CO4
31	Maintenance Necessity of software maintenance, the types of software maintenance.	1	CO2, CO3, CO4
32	Legacy software products and the problems in their maintenance	1	CO2, CO3, CO4
33	Risk Management, Quality Management.	1	CO3, CO4
34	Software Configuration Management (SCM), Project Monitoring plan	1	CO3, CO4

35	Software Quality control and Quality assurance, Metrics for Software Quality, Integrating Metrics within the Software Process, Reliability issues and metrics	1	CO4, CO5
36	Metrics for Process and Projects Process Metrics, Project Metrics, Size Oriented Metrics	1	CO4, CO5
37	Function-Oriented Metrics, Reconciling LOC and FP Metrics	1	CO4, CO5
38	Role of Management in software development.	1	CO4, CO5
39	Object Oriented Metrics, UseCase Oriented Metrics, Role of metrics and measurement	1	CO4, CO5
40	The Road Ahead The new Software Engineering Process, technology as a driver, new modes of representing information, new tools and techniques	1	CO6
Total lecture required		40*	

*Number of lectures may vary.

Book Details

Text Books

1. R. S. Pressman, Software Engineering: A Practitioners Approach, 5th Ed, McGraw-Hill, 2001.
2. Sommerville, Software Engineering, 7th Ed, Addison-Wesley, 2005.

Reference Books

1. Pankaj Jalote, Software Engineering: A Precise Approach, 3rd Edition, 2013.
2. Rajib Mall, Fundamentals of Software Engineering, 3rd Edition, 2013.
3. C. Ghezzi, M. Jazayeri and D. Mandrioli, Fundamentals of Software Engineering, 2nd Ed, Prentice Hall of India, 2003.

Online course work/ Massive Open Online Course/ Open source web material

Lecture notes, online references

Evaluation Scheme (Theory/ Practical)

Evaluation Component	Exam Month	Exam Duration (in Hrs)	Mode of Examination	Weighted Marks
Attendance	Not Applicable	Not Applicable	Not Applicable	10.00
MSI	September	1	Pen-Paper	15.00
MSII	October	1	Pen-Paper	20.00
Project and Seminar	November	Not Applicable	Online	15.00
Lab Assignment	Not Applicable	Not Applicable	Online	10.00
Comprehensive Exam	December	2	Pen-Paper	30.00

Mode of Practical Exam

Offline class work

List of Tentative Practical

List of Practical Project Work

- Introduction of Collaboration Tools.
- Software Requirements Specification (SRS) Document preparation
- Use Cases diagrams
- Class Diagrams
- Sequence Diagram
- Test cases preparation
- Preparation of Project Deployment plan (Development to Production environment)

Test cases preparation

Problem based on Software Metric

Problem based on CPM

Project Work:

Create SRS based on a given system

Create RDD and Design document based on the above system

Project Discussion Individual Groups (Coding issues)

Project Discussion Individual Groups (Testing)

Preparation of Project Deployment plan (Development to Production environment)

Course outcome mapping with evaluation components:

CO	Comprehensive Exam	Lab Assignment	MSI	MSII	Project and Seminar
CO1	3	3	3	2	3
CO2	3	3	3	2	2
CO3	3	3	3	3	3
CO4	3	3	2	3	3
CO5	3	2	2	3	3
CO6	3	1	1	1	2
Max.	3	3	3	3	3

3 is High, 2 is Moderate, 1 is Low & - is Not Applicable

Make up Policy

Students who are likely to miss a component of evaluation due to any genuine reason may be given a make-up for that component by the Course In-Charge. The students are required to approach the Course In-Charge immediately for the same before the conduct of the evaluation component. It is the responsibility of the student to approach the Course In-Charge. The Course In-Charge will not allow makeup, if a student approaches 7 days after the evaluation component (Student Handbook R 35). No makeup for Quiz component.

Plagiarism

We are committed to uphold the standards of academic integrity and honesty. Plagiarism in any form is unacceptable and will be treated seriously (Student Handbook R 49).

Grading Policy

Marks obtained in all the components of evaluation shall be totaled and the final marks shall be converted in the letter grades, namely A, B, C, D, E and NC. The grading is relative and normally, it is centered around the average of a class. Mid-Semester grading will be announced after the completion of about 50% of the evaluation components (Student Handbook R 40).

University Attendance Policy

Students are requested to go through the Student Handbook for better understanding of the attendance policy. Students is advised to regularly check his/her attendance on ERP. In case of any discrepancy in attendance record, student should report only through e-mail/written communication (no oral communication will be entertained) to the Course In-Charge in the same week itself, otherwise request will be not considered (Student Handbook R 37 and NU Attendance policy).

Consultation Hours

All information regarding course will be posted on Moodle. Students are requested to check Moodle regularly. Additionally, the student may approach the Course-In-Charge by email to schedule appointments.