# DA-IICT
# IT314 – Software Engineering

| | |
|---|---|
| 1. Course Title | Software Engineering |
| 2. Credit Structure | Lecture hours per week: 3<br>Tutorial hours per week: 0<br>Practical hours per week: 3<br>Total Credits:4.5 |
| 3. Course Code | IT632 |
| 4. Program/Semester | B.Tech. (ICT) – 6th Semester |
| 5. Category | Core |
| 6. Instructors | Manish Khare (manish_khare@daiict.ac.in) and Saurabh Tiwari (saurabh_t@daiict.ac.in) |
| 7. Course Objective | The following are the objectives of the course:<br><br>• The goal of this course is to provide the student with a solid understanding of the foundations of software engineering.<br>• The course provides a solid foundation for students who want to develop their career in the broad field of computing and specifically in the areas of Information and Communication technology.<br>• The course aims at teaching the students how to apply key engineering principles and mathematical models to application development projects.<br>• The course emphasizes the complete lifecycle of the software development process and the students learn how to design, develop, test, and deploy software using rigorous software engineering practices.<br>• The course will aid the students in developing the skills and abilities of applying the fundamental concepts of computing in industrial, business and other problems, in order to produce software solutions.<br>• The course also introduces the students to the role of software tools in the process of software development.<br><br>This course has been designed to provide the students with the opportunity to apply the software engineering principles learned in this course to a project, that is also a part of this course. Students will work on a significant software development project that may include any or all activities associated with creating a software solution to a client/customer problem. They would be taken through all the software engineering activities that are typically experienced from the initiation to the completion of a software development project. Special emphasis has been placed on defining the client/customer problem and determining requirements by either working with real clients on real world problems. Teams are encouraged to work autonomously following good software engineering practices, with guidance in the form of lectures and tutorials, from the course instructor and teaching assistants. In addition, issue based assistance is provided as and when required or as and when the same is sought by the team members. |

| | |
|---|---|
| 8. Expected Learning Outcomes | Upon completion of this course the student should be able to:<br><br>1. Enumerate and define the phases in the software development process.<br>2. Describe the activities performed in each of the phases and how each phase relates to the others.<br>3. Develop a coherent set of software requirements for a particular application.<br>4. Convert a set of requirements into a set of specifications that can be validated.<br>5. Apply any of several design methodologies to the design of a software work product.<br>6. Develop and implement a test plan that will adequately exercise a software work product with the purpose of discovering defects<br>7. Enumerate and define the steps in the post-implementation phases<br>8. Describe the activities associated with corrective, adaptive, and perfective maintenance<br>9. Describe the activities associated with the configuration management process and relate its importance during software development and maintenance.<br>10. Perform an impact analysis for a change request as it applies to a software work product.<br>11. Perform all software engineering tasks associated with developing a software system or product requiring a team of software engineers.<br>12. At the end of the project, assess a software development effort to determine the appropriate principles and practices that will maximize the probabilities for success. |
| 9. Additional Expectations | The student should be able to:<br>1. Analyze a software development project and determine the most appropriate software engineering principles and practices for the given situation.<br>2. Evaluate the effectiveness of a given set of software engineering practices and make recommendations for changes that can improve the software development project<br>3. Analyze a software development project to determine missing or inappropriate software engineering practices<br>4. Assess the quality of software engineering processes, practices, products, and artifacts associated with a software engineering development effort<br>5. Demonstrate interpersonal and team skills that support maximizing their team's effectiveness. |
| 10. Special Expectations | The student should be able to:<br>1. Work collaboratively and cooperatively with others as a team that produces the required software engineering work products.<br>2. Create and deliver a quality presentation (individually and as part of a team presentation) related to selected aspects of software engineering processes, practices and work products associated with a software engineering project. |

| | |
|---|---|
| 11. Books and Reference Material | • Roger S Pressman, "Software Engineering – A practitioner's Approach", McGraw Hill Higher Education.<br>• Pankaj Jalote, "An Integrated Approach to Software Engineering", Narosa Publications.<br>• Ivor Jacobson, "Object Oriented Software Engineering: A Use Case Driven Approach", Pearson Education Asia<br>• Ian Sommerville, "Software Engineering", Addison-Wesley.<br>• Richard Fairley, "Software Engineering Concepts", Tata McGraw-Hill.<br>• Ali Behforooz, "Software Engineering Fundamentals", Oxford University Press.<br>• Rajib Mall, "Fundamentals of Software Engineering", Prentice Hall of India |

## Course Content

1. **Introduction to Software Engineering (2 Lecture)**
2. **Software Life Cycle Models (2 Lecture)**
3. **Software Requirements: Analysis and Specifications (7 Lecture)**
4. **Software Project Planning (4 Lecture)**
5. **Software Design (7 Lecture)**
6. **Software Metrics (2 Lecture)**
7. **Software Reliability (2 Lecture)**
8. **Software Testing (7 Lecture)**
9. **Software Maintenance (3 Lecture)**
10. **Computer Aided Software Engineering (2 Lecture)**
11. **Agile Software Engineering (2 Lecture)**

## Grading Policy

In -Sem Exams – 20%

End-Sem Exam – 30%

Course Project Work – 30%

Lab, Assignments (or presentations), experiments – 20%

**Course Project Artifacts**

Specific to the software process model chosen for development of the course project.
For example, for Agile Process Model (SCRUM)
1. Requirements in the form of user story (both functional and non-functional)
2. Acceptance Criteria
3. Burn-down chart
4. Daily SCRUM planning and development of spirits.

**Marking Scheme for Projects:**

1. The group mark is everybody's mark, i.e., individual assessment of 15% and group assessment of 15%.
2. Everybody reports what they personally did, and separate marks are given to those components by the instructor. This can be done by using either a dossier of rough work or a division of labor report produced by team members.
3. Other group members report (confidentially or openly) the relative contributions of other group members to allow for an adjustment of the final grade.
4. Pop quizzes in class to ensure that students know the intimate details of the project.
5. Cross-validating with the results of individual work (possibly reducing the weight of group work for students who perform poorly on exams or individual assignments).

After successful completion of the course the student will have the ability to,

- Understand, analyze, build, and test software systems (P1, P2, P12).
- Apply design & modeling principles for software building using various tools and techniques (P3, P5, P11).
- Develop and implement software design in a large group setup (P9, P11).
- Create requirements specifications, detailed design and project management documents, along with necessary test plans. (P10, P11)

| P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 | P9 | P10 | P11 | P12 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| √ | √ | √ | | √ | | | | √ | √ | √ | √ |