|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Subject Name:** | | | | | | | Software Engineering and Project Management | | | | | | | | | | | | | | | | | | | **Subject Code:** | | | | | | | | TMC 206 | | |
|  | | | | | | |  | | | | | | | | | | | | | | | | | | |  | | | | | | | |  | | |
| **Course Name:** | | | | | | | Master of Computer Applications (MCA) | | | | | | | | | | | | | | | | | | |  | | | | | | | |  | | |
|  | | | | | | |  | | | | | | | | | | | | | | | | |  | | | | | | | | |  | | | |
| **1** | **Contact Hours:** | | | | | | | | | 45 | | |  | | | | | | | | | | | | | | **L** | | 3 | | | **T** | | 0 | **P** | 0 |
|  |  | | | | | | | | |  | | |  | | | | | | | | | | | | | |  | |  | | |  | |  |  |  |
| **2** | **Examination Duration (Hrs):** | | | | | | | | | | | | | | |  | **Theory** | | | 0 | 3 |  | **Practical** | | | | | 0 | | 0 | |  | | | | |
|  |  | | | | | | | | | | | | | | |  |  | | |  |  |  |  | | | | |  | |  | |  | | | | |
| **3** | **Relative Weightage:** | | | | | | | | | |  | | | | **CWE:** | | | | 25 | **MTE:** | | | 25 | | **ETE:** | | | | 50 | | | |  | | | |
|  |  | | | | | | | | | |  | | | |  | | |  | |  | | |  | |  | | | |  | | | |  | | | |
| **4** | **Credits:** | | | | | 0 | | 3 | |  | | | | | | | |  | |  | | |  | |  | | | |  | | | |  | | | |
|  |  | | | | |  | |  | |  | | | | | | | |  | |  | | |  | |  | | | |  | | | |  | | | |
| 5 | **Pre-Requisite:** | | | | | | | | | Basics of Information Technology | | | | | | | | | | | | | | | | | | | | | | | | | | |
|  |  | | | | | | | | |  | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 6 | **Subject Area:** | | | | | | | | | Computer Science | | | | | | | | | | | | | | | | | | | | | | | | | | |
|  |  | | | | | | | | |  | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 7 | **Objective:** | | | | | | | | To familiarize students with the procedures and methods of software development and how to manage and produce efficient & cost-effective software systems. | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|  |  | | | | | | | | | | |  | | | | | | | | | | | | | | | | | | | | | | | | |
| 8 | **Learning Outcome:** | | | | | | | | | | |  | | | | | | | | | | | | | | | | | | | | | | | | |
|  | A student who successfully fulfills the course requirements will be able to: | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|  | **CO 1** | | | | Understand and apply software engineering principles and development life cycle models in real life projects. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|  | **CO 2** | | | | Understand and analyze the importance of quality and reliability in software projects. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|  | **CO 3** | | | | Develop software requirement specification and design documents for software projects. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|  | **CO 4** | | | | Evaluate software in terms of size, cost and schedule using project management principles. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|  | **CO 5** | | | | Create and develop test cases using black box and white testing techniques. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|  | **CO 6** | | | | Understand and differentiate various development methodologies including Agile development, Component-Based development, and Cleanroom development. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|  |  | | | |  | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 9 | | **Details of the Course:** | | | | | | | | | | | |  | | | | | | | | | | | | | | | | | | | | | | |
| **Unit No.** | | | | **CONTENT** | | | | | | | | | | | | | | | | | | | | | | | | | | | **CONTACT HOURS** | | | | | |
| **1** | | | | **Introduction:** Introduction to Software**,** Software Crisis, Software Processes & Characteristics, Introduction to Software engineering, Software life cycle models, Process Models. Software quality.  **Software Quality and Reliability:** Introduction**,** Importance of Software Reliability, Failure and Faults, Reliability Models, Software Quality Models, compression of CMM and ISO 9001, Six-Sigma, just in time, total quality management. | | | | | | | | | | | | | | | | | | | | | | | | | | | **8** | | | | | |
| **2** | | | | **Software Project Management:** Software Project Management life cycle**, Software Project Planning Project Estimation**: Size Estimation: Lines of Code & Function Count, Cost Estimation Models, Risk Estimation and Management, Effort estimation, Project monitoring and control.  **Project Scheduling:** Basic Concepts, Project Scheduling- Basic Principles, Relationship between People and Effort, Task Network, Scheduling, Gantt and PERT charts, Staffing. | | | | | | | | | | | | | | | | | | | | | | | | | | | **9** | | | | | |
| **3** | | | | **Software Requirements Analysis & specifications:** Requirement Engineering, Elicitation techniques, Requirements analysis using DFD, Data dictionaries & ER Diagrams, Requirements documentation, Characteristics & organization of SRS, IEEE Standard for SRS.  **Software Design:** Design Concepts, Cohesion & Coupling, Function Oriented Design, Object Oriented Design, Software coding techniques and guidelines. | | | | | | | | | | | | | | | | | | | | | | | | | | | **9** | | | | | |
| **4** | | | | **Software Testing:** Objectives, lifecycle, Testing process, Design of test cases, Levels of Testing, Debugging, testing techniques, Blackbox testing techniques, Whitebox testing techniques, Introduction to functional testing & Structural testing, Object oriented testing and web-based software testing. | | | | | | | | | | | | | | | | | | | | | | | | | | | **8** | | | | | |
| **5** | | | | **Software Maintenance:** Management of Maintenance, Maintenance Process, Maintenance Models, Reverse Engineering, Software Re-engineering, Software Configuration Management. Software Configuration Management vs Software maintenance.  **Advanced Software Development Methodologies:** Agile method Methodology; Extreme programming; Rapid application development; Clean Room Software Development; Component Based Software Development. | | | | | | | | | | | | | | | | | | | | | | | | | | | **8** | | | | | |
|  | | | | **TOTAL** | | | | | | | | | | | | | | | | | | | | | | | | | | | **42** | | | | | |
|  | | | |  | | | | | | | | | | | | | | | | | | | | | | | | | | |  | | | | | |
|  | | | |  | | | | | | | | | | | | | | | | | | | | | | | | | | |  | | | | | |
| **10** | | **Suggested Books:** | | | | | | | | | | | | | | | | | | | | | | | | | | | | |  | | | | | |
| **Sl. NO.** | | | **NAME OF AUTHERS/BOOKS/PUBLISHERS** | | | | | | | | | | | | | | | | | | | | | | | | | | | | | **YEAR OF PUBLICATION** | | | | |
| **1** | | | R. S. Pressman, “Software Engineering – A Practitioner’s Approach”, McGraw Hill Int. , 7th Ed. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 2010 | | | | |
| **2** | | | Stephen R. Schach, “Classical & Object Oriented Software Engineering”, 7th Ed., TMH | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 2007 | | | | |
| **3** | | | Pankaj Jalote, “An Integrated Approach to Software Engineering”, Narosa, 3rd Ed. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 2005 | | | | |
| **4** | | | K. K. Aggarwal and Yogesh Singh, “Software Engineering”, New Age International, 4th Ed. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 2009 | | | | |
| **5** | | | Carlo Ghizzi , Mehdi Jazayeri and Dino Mandrioli, “Fundamental of Software Engineering”, PHI, 3rd Ed. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 2009 | | | | |