MAULANA AZAD NATIONAL INSTITUTE OF TECHNOLOGY
BHOPAL - 462003

Name of Program **B.Tech** **Semester- IV** **Year-II**

Name of Course Software Engineering

Course Code CSE 225

Core / Elective / Other Core

Prerequisite:

1. Learn an OOP Language: Java, C++, Python etc. choose one and try to master it. There are many tutorials/courses that cover these on Coursera and edX.
2. Algorithms and Data Structures: This is like the most important field of Computer Science. Being good at Algorithms and Data Structures is ALWAYS a plus point. Knowing how to implement a particular solution in the most efficient way is key for a Software Developer. Working out problems alongside on platforms like CodeChef, Hacker Rank will help you practice what you read.
3. Choose a Platform: iOS, Android, Windows, Web etc. Choose one and work towards building your knowledge accordingly. Get to know the Platform in and out.
4. Analyze: This is the best way to learn and gain knowledge of how things are happening. For example, if you want to work on Android, analyzing the source code (since it's open source) will help a lot. Participate on forums like Stack Overflow etc.

Course Outcomes:

On successful completion of the course, student be able to:

1. Use the techniques necessary for software engineering practice.
2. Apply software engineering perspective through software design and construction, requirements analysis, verification, and validation, to develop solutions to modern problems such as security, data science, and systems engineering.
3. Apply new knowledge as needed, using appropriate learning strategies.

Description of Contents in brief:

1. Introduction to software engineering, software process & process models, Software metrics and measurements.
2. software project management, software project planning, scheduling and tracking, cost estimation methods
3. Requirements analysis: Principles, complexity, methods, structured analysis, SRS Documentation. Design principles: abstraction, refinement, modularity, control hierarchy, structured partitioning, design types and methods
4. Software coding: coding style, coding efficiency, capability maturity model (CMM), Software quality assurance, Software testing: Software testing techniques, choice and classification of test data, verification & validation methods.
5. Software maintenance, configuration management, system documentation, software reusability, CASE tools.

List of Text Books:

1. Software Engineering: A Practitioner's Approach by Roger S. Pressman, McGraw-Hill International edition.
2. Software Engineering by Ian Sommerville, Addison-Wesley. 5. Fundamentals of Software Engineering by Rajib Mall, PHI

List of Reference Books:

1. An Integrated Approach to Software Engineering, by Pankaj Jalote, Narosa Publishing House.
2. Software Engineering by K.K. Agarwal.