JYOTI NIVAS COLLEGE AUTONOMOUS
SYLLABUS FOR 2018 BATCH AND THEREAFTER

Programme: B.C.A **Semester: III**

SOFTWARE ENGINEERING

Course Code:18BCAIIIT4 **No. of Hours: 60**

COURSE OBJECTIVES:

- This course is to introduce the concepts of software engineering and the various phases in software development in order to furnish the students in developing the software project.
- To understand the stages of software development life cycle.
- To understand various process models and software engineering testing principles.

LEARNING OUTCOMES:

- To understand the nature of software development and Software Life cycle.
- To make the students to understand the methods visualizing and analyzing software requirements.
- To make the students aware of the importance of software engineering principles in designing software projects.
- To prepare the students to flourish the skills needed to scheduling concept and risk management.

UNIT - I **10 HRS**

Introduction to Software Engineering: Defining Software Engineering – Software Engineering and Engineering - Software characteristics, components -The evolution role of software- Software crisis, Software myths, Software engineering technology- Different phases of a software project

Software process: Process model: Classic Life cycle model, Spiral model, Comparisons

UNIT - II **15 HRS**

Requirement Engineering –Requirement engineering tasks- requirement analysis and elicitation-Inception-elicitation-elaboration-negotiation-Softwareprototyping-Specification and validation

System Modeling – elements of analysis model -data modeling concepts-Object Oriented analysis-Scenario based modeling-Flow oriented model -Data dictionary Information flow.

UNIT- III **10 HRS**

Software design and Software Engineering – the design process – design principles – design concepts – effective modular design – design heuristics for effective modularity

Objects Oriented Design - design of object oriented system –the system design process

Mapping Requirements into Software Architecture – transform mapping – transaction mapping – User interface design – interface design activities.

UNIT - IV **15 HRS**

Software testing techniques – software testing fundamentals – white box testing – Basis path testing – Control structure tests – Black box testing.

Software testing strategies **–** A strategic approach to software testing – validation tests – system testing – the art of debugging – software quality, Metrics- software measurement, Software Risks.

UNIT - V **10 HRS**

Software Reengineering – Reverse engineering – Building blocks for CASE – taxonomy of CASE tools.

Cost Estimation- Factors affecting the cost estimation-different techniques of cost estimation-empirical method of cost estimation-COCOMO model

REFERENCES:

1. Roger S Pressman. Software Engineering: A Practitioner's Approach. Tata Mc Graw Hill Publishers. Seventh edition.
2. Watts S Humphrey. A Discipline for Software Engineering. Pearson Education Publishers. First edition.
3. Ian Somerville. Software Engineering. Pearson Education Publishers.Fifth edition.