GANPAT UNIVERSITY

FACULTY OF COMPUTER APPLICATIONS

Programme	Master of Computer Applications	**Branch/Spec.**	Computer Application
Semester	II	**Version**	1.0.0.2
Effective from Academic Year	2024-25	**Effective for the batch Admitted in**	June 2024
Subject Code	P12A3SE	**Subject Name**	Software Engineering

Teaching scheme						**Examination scheme (Marks)**			
(Per week)	**Lecture (DT)**		**Practical (Lab.)**		**Total**		**CE**	**SEE**	**Total**
	L	TU	P	TW					
Credit	2	0	1	0	3	Theory	40	60	100
Hours	2	0	2	0	4	Practical	20	30	50

Objective:

- To understand essential concepts of requirements engineering and analysis modeling.
- To understand the various phases of a software project.
- To learn various process models and metrics, software project risk analysis and management.
- To learn UML software modeling, basic, advanced and behaviors modeling.

Pre-requisites:

- Student must have knowledge of Software Development Process and Object Oriented Concepts.

Course Outcomes :

- **1 = Slight (Low); 2 = Moderate (Medium); 3 = Substantial (High); "-" = No Correlation**

Name of CO	Description
CO1	Understand the software engineering concepts and compare software process models for solving defined problems.
CO2	Apply software metrics and analyze project risks to support effective project planning and management.
CO3	Construct UML structural models using classes, attributes, operations, and relationships for a given system.
CO4	Develop advanced UML structural and behavioral diagrams to model complete system interactions and workflows.

Mapping of CO and PO

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	2	2	1	1	1	1
CO2	2	3	2	2	1	3	1	1
CO3	2	2	3	3	1	1	1	1
CO4	1	2	3	3	1	1	1	2

Content:

Unit	SECTION – I	Hrs
1	**Introduction to Software Engineering:** Defining software, characteristics of software, attributes of a good software, software product, software development life cycle, applications of software, software engineering practices, software myths. **Process Models:** Generic process model (defining framework activity, identifying task set), process assessment & improvement, the Waterfall Model, Incremental Process Models: The Incremental Model, The RAD Model, Evolutionary Software Process Models: Prototyping, The Spiral Model, Concurrent Development Model, Specialized Process Models: Component-Based Development, Aspect oriented Software Development.	10
2	**Metrics for Process and Projects:** Software process and project metrics, Software measurement: Size Oriented Metrics, Function Oriented Metrics. **Risk analysis and Management:** Management activities, project planning, project scheduling, risk analysis and management, Reactive versus proactive risk strategies, Software risks, Risk identification, Risk projection, Risk refinement, Risk mitigation, monitoring, and management	6
	SECTION – II	
3	**Introduction to UML:** Overview of UML, Conceptual Model of UML, Common Mechanisms in the UML, Architecture, Software Development Life Cycle, UML modelling with example. **Basic Structural modeling** : Classes: Names, Attributes, Operations, Organizing Attributes and Operations, Responsibilities Advanced Classes: Classifiers, Visibility, Scope, Abstract Root Leaf and Polymorphic Elements, Multiplicity, Attributes, Operations, Relationships: Dependency, Generalization, Association Advanced Relationships: Dependency, Generalization, Association, Realizations.	6
4	**Advanced Structural modeling:** Class Diagrams: Common Uses, Common modelling Techniques, Interface Types and Roles: Operations, Relationships, Understanding an Interface, Types and Roles, Packages: Names, Owned Elements, Visibility, Importing and Exporting, Generalization, and Standard Elements. **Behavioural modelling Interactions:** Object and Roles, Links and connectors, Use Case Diagrams, Interaction Diagram: Sequence Diagram, Collaboration Diagram. Activity Diagram: Action and Activity States, Transactions, Branching, Forking and Joining, Swim lanes Diagram.	10

Practical Content:

Laboratory work will be based on the above syllabus with minimum 15 experiments to be incorporated which will be considered for evaluation purpose.

Text Books:

1	Roger S. Pressman, “Software Engineering – A Practitioner‟s Approach”, Mc Graw-Hill International Edition 7th Edition 2012.
2	The Unified Modeling Language User Guide by Grady Booch, James Rumbaugh, Ivar Jacobson, Pearson 1st Edition 1998.

Reference Books:

1	Ian Sommerville, “Software Engineering”, 9th Edition, Pearson Education Asia, 2011.
2	Pankaj Jalote, “Software Engineering, A Precise Approach”, Wiley India, 2010.

3	Rajib Mall, “Fundamentals of Software Engineering”, Third Edition, PHI Learning Private Limited, 2009.
4	Stephen R.Schach, “Object Oriented and Classical Software Engineering”,8th Edition Tata McGraw-Hill Publishing Company Limited, 2007.

MOOC/Certification Courses:

1	https://www.edx.org/course/software-engineering-essentials
2	https://www.my-mooc.com/en/mooc/software-engineering-introduction/
3	https://www.classcentral.com/course/swayam-software-engineering-14293
4	https://onlinecourses.nptel.ac.in/noc21_cs65/preview
5	https://www.coursera.org/specializations/software-design-architecture
6	https://epgp.inflibnet.ac.in/Home/ViewSubject?catid=7

Question Paper Scheme:

University Examination Duration: 3 Hours

Note for Examiner: -

(I) Questions 1 and 4 are compulsory with no options.

(II) Internal options should be given in questions 2, 3, 5 and 6.

SECTION – I

Q.1 –8 Marks

Q.2 –11 Marks

Q.3 –11 Marks

SECTION - II

Q.4 –8 Marks

Q.5 –11 Marks

Q.6 –11 Marks