Ganpat University
॥ विद्यया समाजोत्कर्षः ॥

Faculty of Computer Applications

Programme	BCA Honors (Cyber Security)	**Branch**	Computer Applications
Semester	II	**Version**	1.0.0.0
Effective from Academic Year	2026-2027	**Effective for the batch Admitted in**	June 2026
Subject Code	U102B3SAD	**Subject Name**	SOFTWARE ANALYSIS AND DESIGN

Teaching scheme						**Examination scheme (Marks)**			
(Per week)	**Lecture (DT)**		**Practical (Lab.)**		**Total**		**CE**	**SEE**	**Total**
	L	TU	P	TW					
Credit	4		-	-	4	Theory	50	50	100
Hours	4		-	-	4	Practical	-	-	-

Objective:

Students will be able to comprehend various software process models, the software engineering development process, UML diagrams, project planning, project scheduling, and software risk analysis at the end of the course.

Pre-requisites:

The fundamentals of how software is developed should be well understood by the students.

Course Outcomes :

Name of CO	Description
CO1	Gain a comprehensive understanding of software engineering principles, focusing on systematic analysis, design, Software Development Life Cycle phases, and effective Requirements Engineering.
CO2	Design and analyze systems using OOAD principles and Solve real-world problems through case studies.
CO3	Understand the principles of software architecture, Analyze and design software systems with a focus on architectural patterns.
CO4	Analyze case studies to identify testing and QA best practices.
CO5	Understanding project management principles, creating project plans and estimates for software projects, and identifying and managing risks in such projects.

Mapping of CO and PO

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	2	1	0	0	0	1
CO2	3	3	3	2	1	0	0	1
CO3	3	3	3	2	0	0	0	1
CO4	2	3	2	1	1	0	1	1
CO5	2	2	2	1	2	3	2	1

Content:

Unit		Hrs

1	**Introduction to Software Analysis and Design** Introduction to Software Engineering, Basics of Software Analysis and Design, Software Development Life Cycle (SDLC), Requirements Engineering, Importance of Analysis and Design in Software Development	12
2	**Object-Oriented Analysis and Design (OOAD)** Basics of Object-Oriented Programming, Object-Oriented Analysis (OOA) concepts, Object-Oriented Design (OOD) principles, Unified Modeling Language (UML), Case Studies in OOAD	12
3	**Design Patterns and Architecture** Introduction to Design Patterns, Creational, Structural, and Behavioral Design Patterns, Software Architecture fundamentals, Architectural Patterns, Case Studies on Software Architecture	12
4	**Software Testing and Quality Assurance** Basics of Software Testing, Types of Testing: Unit Testing, Integration Testing, System Testing, Acceptance Testing, Testing Techniques and Tools, Quality Assurance in Software Development, Case Studies on Software Testing and QA	12
5	**Software Project Management** Basics of Project Management, Software Project Planning and Estimation, Risk Management in Software Projects, Agile Methodologies, Case Studies in Software Project Management	12

Practical Content:
Not Applicable

Text Books:	
1	"Software Engineering: A Practitioner's Approach" by Roger S. Pressman

Reference Books:	
1	"Requirements Engineering: Fundamentals, Principles, and Techniques" by Klaus Pohl and Chris Rupp
2	"Object-Oriented Analysis and Design with Applications" by Grady Booch, Robert A. Maksimchuk, Michael W. Engle
3	"UML Distilled: A Brief Guide to the Standard Object Modeling Language" by Martin Fowler
4	"Design Patterns: Elements of Reusable Object-Oriented Software" by Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides
5	"Software Architecture in Practice" by Len Bass, Paul Clements, Rick Kazman
6	"Foundations of Software Testing" by Dorothy Graham, Erik Van Veenendaal, Isabel Evans
7	“Effective Software Test Automation" by Kanglin Li
8	"Software Engineering: A Practitioner's Approach" by Roger S. Pressman
9	"Agile Estimating and Planning" by Mike Cohn

Web References / MOOC / Certification Course	
1	https://www.coursera.org/courses?query=software%20design
2	https://www.coursera.org/courses?query=software%20engineering
3	https://www.computer.org/product/education/software-design-course
4	https://onlinecourses.nptel.ac.in/noc19_cs48/preview
5	https://www.edx.org/learn/software-engineering

Question Paper Scheme:	
	End Semester Examination Duration: (2 Hours Theory Examination) **Note for Examiner:** - Q-1 Any Five out of Seven (25 Marks) Q-2 Any Two out of Three (06 Marks) Q-3 Mandatory question (05 Marks) Q-4 Any Two out of Three (08 Marks) Q-5 Any Two out of Three (06 Marks) *The question paper must comprehensively address all Course Outcomes (COs), align Taxonomy levels, and ensure complete syllabus coverage.*