B.Tech. Computer Science and Engineering (Artificial Intelligence and Machine Learning)
MODIFIED SCHEME OF EXAMS W.E.F 2026-27
KURUKSHETRA UNIVERSITY, KURUKSHETRA
SEMESTER-VII

S.No.	Course No./ Code	Subject	L:T:P	Hours/ Week	Credits	End Semester Exam (Marks)	Internal assessment (Marks)	Practical Exam (Marks)	Total (Marks)	Duration of exam (Hours)
1	B23-CAM-401	Cloud Computing	3:0:0	3	3	70	30	0	100	3
2	--	Professional Elective-III	3:0:0	3	3	70	30	0	100	3
3	B23-CAM-403	Soft Computing	3:0:0	3	3	70	30	0	100	3
4	B23-CAM-405	Software Engineering	3:0:0	3	3	70	30	0	100	3
5	B23-CAM-407	Cloud Computing Lab	0:0:4	4	2	0	40	60	100	3
6	B23-CAM-409	Soft Computing Lab	0:0:4	4	2	0	40	60	100	3
7	B23-CAM-411	Project-II	0:0:8	8	4	0	100	100	200	3
8	B23-CAM-413	Industrial Training-II	0:0:2	2	1	--	100	--	100	3
9	B23-VAC-401/403/405/407/409/411	NCC/NSS/Sports/ Yoga/ Technical or Cultural Club/Society activities	0:0:2	2	1		100	--	100	
TOTAL				32	22	280	500	220	1000	

Note:

- The course of both Program Elective and Open Elective will be offered at 1/3rd strength or 20 students (whichever is smaller) of the section.
- B23-CAM-413 is a mandatory non-credit course in which the students will be required to get passing marks.
- B23-VAC-401/403/405/407/409/411 are single credit value added courses in which NCC/NSS/Sports/Yoga/Technical or Cultural Club/Society activities will be joined by students in first year and will be evaluated in 7th semester by the institute based upon continuous evaluation model as per guidelines.

Professional Elective-III	
Machine Learning for Data Science	B23-PCA-415
Unix and Linux Programming	B23-PCA-417
Simulation and Modelling	B23-PCA-419

B23-CAM-401	Cloud Computing						
Lecture	**Tutorial**	**Practical**	**Credit**	**End Semester Exam**	**Internal Assessment**	**Total**	**Time**
3	**-**	**-**	**3**	**70**	**30**	**100**	**3 Hrs.**
Purpose	To introduce various concepts of Cloud Computing.						
	Course Outcomes						
CO1	Introduction & Overview of Computing Paradigm.						
CO2	To Study Cloud Computing Architecture.						
CO3	To study the concepts of Service Management in Cloud Computing.						
CO4	To study the concepts of Cloud Security.						

Unit-I

Overview of Computing Paradigm: Recent trends in Computing, Grid Computing, Cluster Computing, Distributed Computing, Utility Computing, Cloud Computing, evolution of cloud computing.
Introduction to Cloud Computing: Cloud Computing (NIST Model), introduction to Cloud Computing, History of Cloud Computing, Cloud service providers, Properties, Characteristics & Disadvantages, Pros and Cons of Cloud Computing, Benefits of Cloud Computing, Cloud computing vs. Cluster computing vs. Grid computing.

Unit-II

Cloud Computing Architecture: Cloud computing stack, Comparison with traditional computing architecture (client/server), Services provided at various levels, How Cloud Computing Works, Role of Networks in Cloud computing, protocols used, Role of Web services,
Service Models (XaaS) - Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS), Deployment Models- Public cloud, Private cloud, Hybrid cloud, Community cloud.

Unit-III

Service Management in Cloud Computing: Service Level Agreements (SLAs), Billing & Accounting, comparing Scaling Hardware: Traditional vs. Cloud, Economics of scaling: Benefitting enormously, Managing Data- Looking at Data, Scalability & Cloud Services, Database & Data Stores in Cloud, Large Scale Data Processing.
Cloud Applications: Scientific Applications – Health care, Geoscience and Biology. Business and Consumer Applications- CRM and ERP, Social Networking, Media Applications and Multiplayer Online Gaming.
Case study: Eucalyptus, Microsoft Azure, Amazon EC2.

Unit-IV

Cloud Security: Infrastructure Security, Network level security, Host level security, Application level security, Data security and Storage, Data privacy and security Issues, Jurisdictional issues raised by Data location, Identity & Access Management, Access Control, Trust, Reputation, Risk, Authentication in cloud computing, Client access in cloud, Cloud contracting Model.

Suggested books:

1. *Cloud Computing Bible,* Barrie Sosinsky*, Wiley-India,* 2010
2. *Cloud Computing: Principles and Paradigms,* Editors: Rajkumar Buyya, James Broberg, Andrzej M. Goscinski*, Wile,* 2011
3. *Cloud Computing: Principles, Systems and Applications, Editors:* Nikos Antonopoulos, Lee Gillam*, Springer,* 2012

Cloud Security: A Comprehensive Guide to Secure Cloud Computing, Ronald L. Krutz, Russell Dean Vines*, Wiley-India,* 2010

B23-PCA-415	Machine Learning for Data Science						
Lecture	**Tutorial**	**Practical**	**Credit**	**End Semester Exam**	**Internal Assessment**	**Total**	**Time**
3	**0**	**0**	**3**	**70**	**30**	**100**	**3 Hrs**
Purpose	To gain the knowledge of Machine Learning technologies, data processing, use and evaluation of data sets, use of ML tools and study of advances in AI						
Course Outcomes (CO)							
CO1	Recognize the role of learned models in data science and determine appropriate model type(s) and analysis methods based on data type(s) and application.						
CO2	Prepare/preprocess data, apply machine learning algorithms to it, and analyze the models learned.						
CO3	Explain the fundamental categories of machine learning (supervised, unsupervised, etc.) and particular algorithms within each category.						
CO4	Iteratively refine the data representation, algorithm choice and parameters, and analysis methods to optimize system performance.						

UNIT-I

Introduction: Artificial Intelligence (AI), Machine Learning (ML), and Deep Learning (DL). Types of Machine Learning.

Prerequisites for Machine Learning: Linear Algebra, Probability and Statistics, Calculus, Programming.

Tools and Environments

UNIT-II

Data Preparation and Processing: Data Collection and Exploration, Data Cleaning and Transformation Handling missing data, Normalizing and standardizing, Encoding categorical variables, Outlier detection and treatment.

Feature Engineering: Feature selection, Feature extraction: Component Analysis (PCA) and Linear Discriminant Analysis (LDA), Creating derived features

Tools and Techniques: Pandas for data manipulation and cleaning, scikit-learn's preprocessing module for scaling and encoding, matplotlib/ seaborn for visualization, Feature-engine for automated feature engineering tasks.

UNIT-III

Machine Learning Algorithms: Supervised Learning Algorithms, Supervised learning Algorithms, Unsupervised Learning Algorithms, Reinforcement Learning Algorithms.

UNIT-IV

Neural Networks: NN, ANN**Deep Learning**: FNN, CNN, RNN

Advanced Topics in Deep Learning: Transfer Learning: Generative Adversarial Networks (GANs), Attention Mechanisms and Transformers, Ensemble Learning.

Model Evaluation and Validation: Training and Testing Data, Evaluation Metrics, Avoiding Overfitting and Underfitting, Model Selection and Hyperparameter Tuning, Machine Learning Deployment and Monitoring. Ethics and Responsible AI.

Case Studies: Healthcare, Finance, Autonomous Vehicles, Natural Language Processing (NLP).

Suggested Books:

1. Lior Rokach, Oded Maimon, Erez Shmueli, Machine Learning for Data Science Hanbook,Springer International Publishing AG
2. Machine Learning in Data Science Using Python, R. Nageswara Rao,Dreamtech Press / Wiley India
3. Prateek Grawal, Charu Gupta, Anand Sharma, Vishu Madaan, Nisheeth Joshi, Machine Learning and Data Science: Fundamentals and Applications by Wiley-Scrivener
4. Daniel D. Gutierrez ,Machine Learning and Data Science, Technics Pubns Llc
   Zdravko I, Botev, , Dirk P. Kroese,, Thomas Taimre, Data Science and Machine Learning: Mathematical and Statistical Methods, Chapman and Hall/CRC

B23-PCA-417	Unix and Linux Programming						
Lecture	**Tutorial**	**Practical**	**Credit**	**End Semester Exam**	**Internal Assessment**	**Total**	**Time**
3	**-**	**-**	**3**	**70**	**30**	**100**	**3 Hrs.**
Purpose	Expertisation in computational programming skills on Unix/Linux Environment.						
Course Outcomes (CO)							
CO1	Learning of simple & advanced commands with features and characteristics of Unix /Linux Systems.						
CO2	Exploring knowledge of programming development skills using Shell, Filters, editors and other utilities.						
CO3	Analyzing the programming behaviour based on programming development/management on Unix /Linux Systems.						
CO4	Developing creativity as system administrative with networking expertisation in Unix /Linux Systems.						

Unit 1

Module 1: Introduction to Linux/Unix and Fedora Environment . Overview: History of UNIX and Linux (GNU/Linux) and features. Architecture: Kernel, Shell, and User Utilities. Basic Commands: ls, cd, pwd, mkdir, rm, cp, mv, cat. Package Management: Installing and upgrading software. Permissions: chmod, chown, chgrp, and default file permissions (umask). File compression and decompression: tar, gzip, zip commands.

Unit 2

The Shell and Text Processing. Introduction to Bash: Shell variables, input/output redirection, pipes (|), and command substitution. File Handling: Inodes, hard links, and symbolic links (ln). Text Processing Tools: grep, sed, awk, head, tail, sort, wc, uniq. Editor: vi or vim editor basics. Script Basics: Shebang (#!/bin/bash), running scripts, comments and simple programs. Variables & Scope: System and User-defined variables, environment variables. Control Structures: if-else, case statements, for loops, while loops.

Unit 3

The C Language Environment : C language programming in Unix/Linux, C compiler options, C Shell operators, C Shell Script & programming. Project Management: MakeFile Utility for keeping program up-to-date & its use for dependency calculations, dynamic linking and loading of libraries modules, static and shared libraries, dynamic . Searching pattern :grep, egrep, fgrep.

Unit 4

Networking and Administration. Security: User authentication and password administration. Networking Commands: ip, ifconfig, ping, netstat, dig. Firewall: Configuring firewall(Fedora default) or iptables. SSH: Secure shell configuration and key-based authentication. System Management: Controlling services (systemctl start/stop/enable/status). User Management: useradd, usermod, userdel, /etc/passwd. Processes: ps, top, kill, nice, bg, fg. Process Control: fork(), exec(), wait(), exit(). job control : at, batch, crontab.

Case Study : Database Server and SSH server in linux

Suggested Books :

1. Sumitbha Das : Unix – Concept and Applications, Fourth Edition TMH, 2015
2. B.M Harwani, Unix and Shell Programming, Oxford University Press, 2013
3. Neil Matthew, Richard Stones : Beginning Linux Programming, 4th. Edition, Wrox-Shroff, 2011.
4. Welsh & Kaufmann : Running Linux, O' Reiley & Associates, 2013.

B23-PCA-419	Simulation and Modelling						
Lecture	**Tutorial**	**Practical**	**Credit**	**End Semester Exam**	**Internal Assessment**	**Total**	**Time**
3	-	-	**3**	**70**	**30**	**100**	**3 Hour**
Purpose	To introduce the principles and paradigms of Computer Modeling and Simulation for solving a wide variety of problems. In addition, how to use simulator to simulate the live systems.						
Course Outcomes (CO)							
CO1	Learn the basic concepts of System, System Modeling, types of Models, simulation, and need of simulation.						
CO2	Learn the simulation of continuous and discrete systems with the help of different examples.						
CO3	Learn the concept of generation of uniformly and non-uniformly distributed random numbers.						
CO4	Learn the simulation of queuing system and PERT.						

Unit-1

Modeling: System Concepts, system boundaries and environment, continuous and discrete systems, system modeling, types of Models, Model validation, Principles & Nature of Computer modeling.

Simulation: Introduction, Basic nature of simulation, when to simulate, Advantages, disadvantages and limitations of simulation, Concepts of simulation of continuous and discrete system with the help of example.

Unit-2

Continuous System Simulation: Analog vs. digital simulation, continuous simulation vs. numerical integration, simulation of a chemical reactor, simulation of a water reservoir system.

Discrete system simulation: Fixed time-step vs. event-to-event model, Monte-Carlo computation vs. stochastic simulation, generation of random numbers, and generation of non-uniformly distributed random numbers.

Unit-3

Simulators for the Live systems: Simulation of queuing Systems: basic concepts of queuing theory, simulation of single server, two server and more general queuing system.

Simulation of PERT network: Network model of a project, analysis of an activity network, critical path computation, uncertainties in activity durations, simulation of an activity network.

Unit-4

Simulation of inventory control systems: Elements of inventory theory, inventory models, generation of Poisson and Erlang variates, simulator for complex inventory systems.

Design and Evaluation of Simulation Experiments: Variance reduction techniques. Experiment layout and Validation. **Case Study**: SciLab, Octave.

Text Books:

1.Gordon G.: System simulation, Prentice-Hall of India Pvt. Ltd. New Delhi 1993

2.NarsinghDeo: System Simulation with Digital Computer: PHI New Delhi, 1993

Reference books:

1. Neelankavil Frances: Computer Simulation and Modelling, John Wiley & Sons, NewYork, 1987.

2. Payne, James A.: Introduction to simulation: Programming Techniques and Methods of Analysis, McGraw-Hill International Editions, Computer Science services, New York (1998).

3. 3.ReitamJulian: Computer Simulation Experiments, Wiley Interscience 1971.

B23-CAM-403	SOFT COMPUTING						
Lecture	**Tutorial**	**Practical**	**Credit**	**End Semester Exam**	**Internal Assessment**	**Total**	**Time**
3	**-**	**-**	**3**	**70**	**30**	**100**	**3 Hrs.**
Purpose	Soft Computing deals with imprecision, uncertainty, partial truth, approximation to achieve practicability, robustness, and low cost solution for complex problems in real world using neural networks, fuzzy systems, evolutionary computation with optimization approaches to design intelligent systems.						
Course Outcomes (CO)							
CO1	Explain the concepts of soft computing, neural networks, fuzzy logic, and genetic algorithms.						
CO2	Apply neural network models and learning algorithms to solve classification and regression problems.						
CO3	Implement fuzzy inference systems for decision-making applications. Design and implement genetic algorithms for optimization problems.						
CO4	Develop hybrid soft computing systems for real-world applications.						

Unit 1
Introduction to Soft Computing and Neural Networks

Introduction to Soft Computing -- Concept of Computing: Hard vs Soft Computing, Characteristics of Soft Computing, Components: Neural Networks, Fuzzy Logic, Genetic Algorithms, Applications of Soft Computing, Artificial Neural Networks (ANN) -- Biological Neuron vs Artificial Neuron, McCulloch-Pitts Model, Perceptron Model, Learning Rules: Hebbian, Delta Rule, Adaline and Madaline, Multilayer Perceptron (MLP) -- Architecture of MLP, Backpropagation Algorithm, Error Minimization and Gradient Descent, Applications of ANN.

Unit 2
Fuzzy Logic Systems

Introduction to Fuzzy Logic -- Crisp Sets vs Fuzzy Sets, Membership Functions, Fuzzy Set Operations, Fuzzy Relations -- Cartesian Product, Composition of Fuzzy Relations, T-norms and T-conorms, Fuzzy Inference Systems -- Mamdani Model, Sugeno Model, Fuzzification and Defuzzification Methods, Rule Base and Decision Making, Applications of Fuzzy Logic

Unit 3
Genetic Algorithms and Evolutionary Computing

Introduction to Genetic Algorithms (GA) -- Biological Background, Basic Terminologies: Population, Chromosome, Gene, Fitness Function, GA Operators -- Selection Methods, Crossover Techniques, Mutation Techniques, GA Algorithm and Flow -- Encoding Schemes, Convergence Criteria, Advantages and Limitations, Applications of GA -- Optimization Problems, Scheduling, Feature Selection

Unit 4
Hybrid Systems and Soft Computing Techniques

Neuro-Fuzzy Systems – Architecture, Adaptive Neuro-Fuzzy Inference System (ANFIS), Particle Swarm Optimization (PSO), Swarm Intelligence, PSO Algorithm, Introduction to Other Evolutionary Techniques -- Differential Evolution, Ant Colony Optimization, Hybrid Soft Computing Systems -- GA plus ANN, Fuzzy plus ANN, Real-world Case Studies.

Text Books

1. S. N. Sivanandam and S. N. Deepa, *Soft Computing*, Wiley India.
2. K. T. Chaturvedi, *Soft Computing and its Applications*, Springer.
3. J.S.R. Jang, C.T. Sun and E. Mizutani, *Neuro-Fuzzy and Soft Computing*, Pearson.

Reference Books

1. Timothy J. Ross, *Fuzzy Logic with Engineering Applications*, Wiley.
2. David E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Pearson.
3. Simon Haykin, *Neural Networks and Learning Machines*, Pearson.

B23-CAM-405	Software Engineering						
Lecture	**Tutorial**	**Practical**	**Credit**	**End Semester Exam**	**Internal Assessment**	**Total**	**Time**
3	**-**	**-**	**3**	**70**	**30**	**100**	**3 Hrs.**
Purpose	To gain a broad understanding of the discipline of software engineering and its application to the development and management of software process.						
Course Outcomes (CO)							
CO1	To understand the basic concepts of Software Engineering.						
CO2	To learn about the skills that will enable to construct high quality software.						
CO3	To understand the software process models.						
CO4	To understand the fundamental concept of requirements engineering and Analysis Modelling.						

Unit 1

Introduction: Introduction to Software Engineering, Software Characteristics, Software Crisis, The Evolving role of Software, Software Development Life Cycle (SDLC) Models: Water Fall Model, Prototype Model, Spiral Model, Evolutionary Development Models, Iterative Enhancement Models, RAD, V Model.

.

Unit 2

Software Requirement Specification: Requirement Engineering Process: Elicitation, Analysis, Documentation, Review and Management of User Needs, Feasibility Study, Data Flow Diagrams, Decision Tables, SRS Document, IEEE Standard for SRS.

Software Quality: Software Quality, Concept of Software Quality Assurance (SQA), SEI-CMM Model. Introduction to Software Risk Management and Software Configuration Management

Unit 3

Software Design: Basic Concept of Software Design, Modularization, Design Structure Charts, Pseudo Codes, Flow Charts, Coupling and Cohesion.

Design Strategies: Function Oriented Design, Object Oriented Design, Top-Down and Bottom-Up Design.

Software Measurement and Metrics: Various Size Oriented Measures: Halstead‟s Software Science, Function Point (FP) Based Measures, COCOMO, Cyclomatic Complexity Measures: Control Flow Graphs.

Unit 4

Software Construction: Software construction fundamentals, minimizing complexity, Top-Down and Bottom –Up programming, structured programming, Compliance with Design and Coding Standards.

Testing: Testing Objectives, Unit Testing, Integration Testing, system testing, Acceptance Testing, Regression Testing, Structural Testing, Functional Testing, debugging.

Maintenance: key issues, Types of software Maintenance, Cost of Maintenance, Software Re-Engineering.

Text Books:

1. R. S. Pressman, Software Engineering: A Practitioners Approach, McGraw Hill.
2. K. K. Aggarwal and Yogesh Singh, Software Engineering, New Age International Publishers.

Reference Books:

1. Pankaj Jalote, Software Engineering, Wiley India.
2. Rajib Mall, Fundamentals of Software Engineering, PHI Publication.
3. Ian Sommerville, Software Engineering, Addison Wesley.

B23-CAM-407	Cloud Computing Lab						
Lecture	**Tutorial**	**Practical**	**Credit**	**Internal Assessment**	**Practical Exam**	**Total**	**Time**
--	**--**	**4**	**2**	**40**	**60**	**100**	**3 Hour**
Purpose	Design and implement various cloud related applications.						
Course Outcomes (CO)							
CO1	Hands on experiments on Hadoop.						
CO2	To Install Google App Engine and create web applications. .						
CO3	To Demonstrate the use of Map and Reduce tasks.						
CO4	To run applications like word count.						

List of Practicals

1. Write a program to use the API's of Hadoop to interact with it.
2. Install a C compiler in the virtual machine created using virtual box and execute Simple Programs
3. Install Google App Engine. Create hello world app and other simple web applications using python/java.
4. Use GAE launcher to launch the web applications.
5. Show the virtual machine migration based on the certain condition from one node to the other.
6. Write a word count program to demonstrate the use of Map and Reduce tasks.
7. Find procedure to set up the one node Hadoop cluster and run simple applications like word count.

B23-CAM-409	SOFT COMPUTING LAB						
Lecture	**Tutorial**	**Practical**	**Credit**	**Internal Assessment**	**Practical Exam**	**Total**	**Time**
--	**--**	**4**	**2**	**40**	**60**	**100**	**3 Hour**
Purpose	To Design and implement various Soft Computing Techniques/Algorithms.						
Course Outcomes (CO)							
CO1	Train various neural networks and learning algorithms						
CO2	Implement fuzzy operations for decision-making applications.						
CO3	Design and implement genetic algorithm for optimization problems.						
CO4	Develop hybrid soft computing systems.						

List of Practicals

1. Design and train a neural network for logic OR, AND, XOR.
2. Implementation of Perceptron Neural Network.
3. Implementation of Multi-Layer Perceptron Neural Network.
4. Implementation of Backpropagation Neural Network.
5. Perform Fuzzy Set Operations -- Membership functions.
6. Perform Fuzzy Set Operations -- Union, Intersection, Complement
7. Implementation of Genetic Algorithm
8. Implement Travelling Salesman Problem for Optimization using GA.
9. Implement Particle Swarm Optimization Technique.
10. Implementation of Neuro-Fuzzy System

B.Tech. Computer Science and Engineering (Artificial Intelligence and Machine Learning)
MODIFIED SCHEME OF EXAMS W.E.F 2026-27
KURUKSHETRA UNIVERSITY, KURUKSHETRA
SEMESTER-VIII

S. No.	Course No./Code	Subject	L: T:P	Hours/Week	Credits	End Semester Exam (Marks)	Internal Assessment (Marks)	Practical Exam (Marks)	Total (Marks)	Duration Of exam (Hours)
1	--	Professional Elective-IV	3:0:0	3	3	70	30	0	100	3
2	--	Professional Elective-V	3:0:0	3	3	70	30	0	100	3
3	B23-CAM-402	Project-III / Industrial Project/ Research Lab/ /Start-ups / Entrepreneurship	0:0:20	20	10	0	200	200	400	3
TOTAL				**26**	**16**	**140**	**260**	**200**	**600**	

Note:

- The course of both Professional Elective will be offered at 1/3rd strength or 20 students (whichever is smaller) of the section.
- In case of semester-long project work done in industry/external institute, the Program Elective- IV and Open Elective-IV may be offered in online mode through MOOC courses offered by SWAYAM/NPTEL portal. These courses may be done from 3rd semester till completion of the degree.
- In case of semester-long project work done in industry the Program Elective- IV and Programme Elective-V may be offered in online mode through MOOC courses offered by SWAYAM/NPTEL.

Program Elective -IV		**Program Elective -V**	
Agile Software Engineering	B23-PCA-404	Social Networks	B23-PCA-410
Software Verification, Validation and Testing	B23-PCA-406	AI and Robotics	B23-PCA-412
Software Project Management	B23-PCA-408	Neural Network and Fuzzy Logic	B23-PCA-414
		Health Care and Analytics	B23-PCA-416

B23-PCA-404	Agile Software Engineering						
Lecture	**Tutorial**	**Practical**	**Credit**	**End Semester Exam**	**Internal Assessment**	**Total**	**Time**
3	**-**	**-**	**3**	**70**	**30**	**100**	**3 Hrs.**
Purpose	Introduces the business value of adopting Agile approaches and provide complete understanding of the Agile development practices.						
Course Outcomes							
CO1	Understand the background and driving forces for taking an Agile approach to software development.						
CO2	Understand the business value of adopting Agile approaches.						
CO3	Drive development with unit tests using Test Driven Development.						
CO4	Apply design principles and refactoring to achieve Agility.						

Unit I: Fundamentals of Agile

The Genesis of Agile, Introduction and background, Agile Manifesto and Principles, Overview of Scrum, Extreme Programming, Feature Driven development, Lean Software Development, Agile project management, Design and development practices in Agile projects, Test Driven Development, Continuous Integration, Refactoring, Pair Programming, Simple Design, User Stories, Agile Testing, Agile Tools

Unit II: Agile Scrum Framework

Introduction to Scrum, Project phases, Agile Estimation, Planning game, Product backlog, Sprint backlog, Iteration planning, User story definition, Characteristics and content of user stories, Acceptance tests and Verifying stories, Project velocity, Burn down chart, Sprint planning and retrospective, Daily scrum, Scrum roles – Product Owner, Scrum Master, Scrum Team, Scrum case study, Tools for Agile project management.

Unit III: Agile Testing

Agile Testing Planning and Managing Testing Cycle, Agile Lifecycle and its impact on testing, Principles of Agile Testing, Agile Testing Techniques, Test-Driven Development, User Acceptance Tests, Test Automation.
Agile Project Management Scheduling in an agile project, scheduling challenges, estimating costs, monitoring project progress, burning down the product backlog, reporting, controlling the project

Unit IV: Agile Software Design and Development

Agile design practices, Role of design Principles including Single Responsibility Principle, Open Closed Principle, Liskov Substitution Principle, Interface Segregation Principles, Dependency Inversion Principle in Agile Design, Need and significance of Refactoring, Refactoring Techniques, Continuous Integration, Automated build tools, Version control.

Suggested Books:

- Ken Schawber, Mike Beedle, *Agile Software Development with Scrum*, Pearson publications
- Robert C. Martin, *Agile Software Development, Principles, Patterns and Practices*, Prentice Hall
- Lisa Crispin, Janet Gregory, *Agile Testing: A Practical Guide for Testers and Agile Teams*, Addison Wesley
- Alistair Cockburn, *Agile Software Development: The Cooperative Game*, Addison Wesley
- Mike Cohn, *User Stories Applied: For Agile Software,* Addison Wesley
- Enterprise-Scale Agile Software Development James Schiel Latest edition, CRC Press
  Succeeding with Agile: Software Development Using Scrum Mike Cohn Latest edition, Addison-Wesley

B23-PCA-406	Software Verification and Validation and Testing						
Lecture	**Tutorial**	**Practical**	**Credit**	**End Semester Exam**	**Internal Assessment**	**Total**	**Time**
3	**0**	**0**	**3**	**70**	**30**	**100**	**3**
Program Objective (PO)	**The objective of this course is to provide the in-depth coverage of software quality models and software testing strategies. It focuses on test case generation techniques and testing levels. It also focuses on testing different kinds of software.**						
Course Outcomes (CO)							
CO1	To develop test cases for any problem						
CO2	To pursue testing on any level of software design by using different testing strategies						
CO3	To learn the test management and testing activities by using different testing methods.						
CO4	To apply testing and quality model of software testing in achieving high- quality software.						

UNIT – I

Introduction: Overview of software evolution, SDLC, Testing Process, Terminologies in Testing: Error, Fault, Failure, Verification, Validation, Definition of software testing, Test Cases, Test Oracles, Testing Process, Limitations of Testing.

UNIT - II

Functional Testing: Boundary Value Analysis, Equivalence Class Testing, Decision Table Based Testing, Cause Effect Graphing Technique.

Structural Testing: Path testing, DD-Paths, Cyclomatic Complexity, Graph Metrics, Data Flow Testing, Mutation testing.

UNIT - III

Testing Activities: Unit Testing, Levels of Testing, Integration Testing, System Testing, Debugging, Domain Testing.

Reducing the number of test cases: Prioritization guidelines, Priority category, Scheme, Risk Analysis, Regression Testing, and Slice based testing

UNIT - IV

Overview of SQM: Concepts of Software Quality, Quality Attributes, Software Quality Models: McCall, Boehm, ISO-9000, CMM.

Miscellaneous topics: Stress Testing, Ad hoc testing: Buddy testing, Exploratory testing, Agile and extreme testing.

Suggested Books:

1. Naresh Chauhan "Software Testing Principles and Practices" Oxford Publications, 2012
2. Louise Tamres, "Software Testing", Pearson Education Asia, 2002
3. Robert V. Binder, "Testing Object-Oriented Systems-Models, Patterns and Tools", Addison Wesley.
4. William Perry, "Effective Methods for Software Testing", John Wiley & Sons, New York.
5. CemKaner, Jack Falk, Nguyen Quoc, "Testing Computer Software", Second Edition, Van Nostrand Reinhold, New York.
6. K.K. Aggarwal & Yogesh Singh, "Software Engineering", 2nd Ed., New Age International Publishers, New Delhi, 2005
7. Boris Beizer, "Software Testing Techniques", Second Volume, Second Edition, Van Nostrand Reinhold, New York.

B23-PCA-408	Software Project Management						
Lecture	**Tutorial**	**Practical**	**Credit**	**End Semester Exam**	**Internal Assessment**	**Total**	**Time**
3	**0**	**0**	**3**	**70**	**30**	**100**	**3**
Program Objective (PO)	The course provides an understanding of the purpose, methods and benefits of process management by exposing the student to the concepts, practices, processes, tools and techniques used in process management for software development.						
Course Outcomes (CO)							
CO1	To understand the basic concepts, terminologies and issues of software project management.						
CO2	To apply appropriate methods and models for the development of solutions.						
CO3	To analyze the cost-benefits of calculations so as to optimize the selection strategy						
CO4	To evaluate methods, models and technologies towards achieving project success						

UNIT I-PROJECT CONCEPTS AND ITS MANAGEMENT

Project life cycle models-ISO 9001 model-Capability Maturity Model-Project Planning-Project tracking-Project closure. Evolution of Software Economics – Software Management Process Framework: Phases, Artifacts, Workflows, Checkpoints – Software Management Disciplines: Planning / Project Organization and Responsibilities / Automation / Project Control – Modern Project Profiles

UNIT II-COST ESTIMATION

Problems in Software Estimation – Algorithmic Cost Estimation Process, Function Points, SLIM (Software

Life cycle Management), COCOMO II (Constructive Cost Model) – Estimating Web Application Development – Concepts of Finance, Activity Based Costing and Economic Value Added (EVA) – Balanced Score Card.

UNIT III-SOFTWARE QUALITY MANAGEMENT

Software Quality Factors – Software Quality Components – Software Quality Plan – Software Quality Metrics – Software Quality Costs – Software Quality Assurance Standard – Certification – Assessment.

UNIT IV-SOFTWARE MANAGEMENT AND METRICS

Software Configuration Management – Risk Management: Risk Assessment: Identification / Analysis / Prioritization – Risk Control: Planning / Resolution / Monitoring – Failure Mode and Effects Analysis (FMEA) – Defect Management – Cost Management. Software Metrics – Classification of Software Metrics: Product Metrics: Size Metrics, Complexity Metrics, Halstead's Product Metrics, Quality Metrics, and Process metrics.

REFERENCES

1. Ramesh Gopalaswamy ,*"Managing and global Software Projects"*, Tata McGraw Hill Tenth Reprint, 2011.
2. Roger S.Pressman, *"Software Engineering- A Practitioner's Approach",* 7th Edition ,Mc Graw Hill, 2010.
3. Daniel Galin, "*Software Quality Assurance: from Theory to Implementation*", Addison- Wesley, 2003.
4. Pankaj Jalote, "*Software Project Management in Practice*", Pearson, 2002.
5. Stephen H, Kan, "*Metrics and Models in Software Quality Engineering*", Addison- Wesley, 2002.

Bob hughes and mike cotterell, *"Software project Management"* second edition,1999.

B23-PCA-410	Social Networks						
Lecture	**Tutorial**	**Practical**	**Credit**	**End Semester Exam**	**Internal Assessment**	**Total**	**Time**
3	**-**	**-**	**3**	**70**	**30**	**100**	**3 Hrs.**
Program Objective (PO)	This emerging and innovative field provides insight into the latest communication mechanisms used in online social networks. It focuses on identifying and representing relationships and predicting links through the effective integration of graph theory, matrices, clustering techniques, and equivalence concepts. The course enables students to understand the fundamental principles of social networks, including link prediction, community detection, and network structure analysis.						
Course Outcomes (CO)							
CO1	Understand clustering, community detection, and structural relationships among nodes to determine their relative importance within a network.						
CO2	Explain key concepts such as homophily, selection, closure, segregation, relationship formation, and graph construction for efficient network representation.						
CO3	Analyze social networks using measures such as betweenness, centrality, equivalence relations, centralization, clustering coefficient, structural cohesion, PageRank, and random graph models, and perform visualization and empirical analysis of network data.						
CO4	Interpret and synthesize network analysis results using epidemic models, generative and experimental models, and equivalence concepts to analyze complex datasets and improve recommendation systems.						

Unit-I (Fundamentals of Social Networks)

Introduction to Social Networks, Google Page Rank, Link Prediction, Importance of Acquaintances, Web Graph, Introduction: Emergence of Connectedness, Granovetter's Strength of weak ties, Triads, clustering coefficient and neighborhood overlap, Structure of weak ties, bridges, and local bridges, Emeddedness, Betweenness Measures and Graph Partitioning, Finding Communities in a graph (Brute Force Method), Community Detection Using Girvan Newman Algorithm, Strong and Weak Relationship.

Unit-II (Network Formation and Social Structure)

Introduction to Homophily, Selection and Social Influence, Foci Closure and Membership Closure, Introduction to Fatman Evolutionary model, Triadic Closure, Spatial Segregation: An Introduction, Schelling Model Implementation, Positive and Negative Relationships – Introduction, Structural Balance, Creating graph, displaying it and counting unstable triangles, Equal Coin Distribution, Random Walk Coin Distribution.

Unit-III (Network Metrics and Models)

Metrics in social network analysis (Betweenness, Centrality, Equivalence relation, Centralization, Clustering coefficient and Structural cohesion), Diffusion in Networks, Impact of Communities on Diffusion, Cascade and Clusters, Introduction to Hubs and Authorities, Hubs and Authorities, PageRank as a Matrix Operation, Introduction to Power Law, Rich Get Richer Phenomenon, Implementing a Random Graph (Erdos-Renyi Model).

Unit-IV (Advanced Topics and Application)

Rich Get Richer - The Long Tail, Epidemics- An Introduction, Simple Branching Process for Modeling Epidemics, Basic reproductive number, SIR and SIS spreading models, Percolation model, Milgram's Experiment, The Generative Model, Decentralized Search, Basic of Equivalence concepts in Social Networks.

Suggested Books:

1. Networks, Crowds, and Markets — David Easley and Jon Kleinberg, Cambridge University Press.
2. Social and Economic Networks — Matthew O. Jackson, Princeton University Press.
3. Mining the Social Web — Matthew A. Russell, O'Reilly Media, Second Edition.
4. Introduction to Social Network Methods — Robert A. Hanneman and Mark Riddle, University of California, Riverside.
5. Social Network Analysis: Theory and Applications — Free Wikibook, available online.
6. Social Network Analysis — John Scott and Peter J. Carrington, SAGE Publications.

B23-PCA-412	AI and Robotics						
Lecture	**Tutorial**	**Practical**	**Credit**	**End Semester Exam**	**Internal Assessment**	**Total**	**Time (Hrs)**
3	**0**	**0**	**3**	**70**	**30**	**100**	**3**
Course Objectives	To enlighten the students about the fundamentals of Robotics and the integration of AI in robotics applications.						
Course Outcomes							
At the end of this course, the student should be able to understand							
CO1	The basic concepts related to the Robot, parts of Robots, End Effectors and to make familiar with the various Drive systems for Robots.						
CO2	The Kinematics, Dynamics and operation of various Sensors and their Applications in Robots.						
CO3	The Machine Vision and its Applications, and various Control Systems used in Robots.						
CO4	Integrate AI with ROS for robotic applications.						

UNIT I

Fundamentals of Robotics: Definition, History, Development in Robot Technology, Characteristics, Basic Components, Robot Anatomy, Robot Generations, Robot Selection, Present and Future Applications in manufacturing, healthcare, defense, and service industries.

Robots Drive Systems and End Effectors: Robot Classification: Arm Geometry, Degrees of Freedom, Power Sources, Types of Motion, Path Control; Robot End Effectors: Mechanical Grippers, Vacuum, Magnetic, Adhesive; Special Purpose Grippers, Process Tooling, Compliance, Robot Drive Systems: Hydraulic, Pneumatic, and Electric Systems.

UNIT II

Kinematics and Dynamics: Forward kinematics: Denavit–Hartenberg convention, examples; Inverse kinematics: Analytical, geometric, numerical methods; Dynamics modelling: Newton-Euler, Lagrangian formulation, Jacobian, velocity kinematics.

Sensors: Requirements of a Sensor, Sensor Classification; Principle, Advantages, Disadvantages and Applications of the following Sensors: Position Sensors, Velocity Sensors, Acceleration Sensors, Torque Sensors, Micro Switches, Visible Light and Infrared Sensors, Touch and Tactile Sensors, Proximity Sensors, Range Finder, Sniff Sensors, Taste Sensors, Vision Sensors, Voice Recognition Devices, Voice Synthesizers, RCC.

UNIT III

Machine Vision and Control System: Visual Sensing, Architecture of Robotics Vision System, Machine Vision: Image Acquisition - Vidicon Tube and CCD; Digitization, Image Processing, Image Analysis, Depth Measurement with Vision System, Image Interpretation, Segmentation, Image Data Compression, Machine Vision Application, Control Systems: Basic Robot Control System, PLC, PID, CNC, and MPU. Robot programming: Languages, offline vs online, simulation tools, Integration of sensors and actuators in robotic systems.

UNIT IV

Robotics Design & AI Integration: Path planning & navigation- graph-based methods: Dijkstra, A*; Sampling-based methods: RRT, PRM; Robot Operating System (ROS): architecture, nodes, topics, services; Human-Robot Interaction (HRI): interfaces, safety, usability; AI in robotics: vision-based navigation, autonomous vehicles, swarm robotics, Industrial Applications; Case studies: self-driving cars, warehouse automation, surgical robots.

Suggested Books:

1. James G. Keramas, "Robot technology fundamentals", **Delmar Publishers.**
2. Saeed B. Niku, "Introduction to robotics analysis, control and applications", **2nd ed., Wiley India.**

R. K. Mittal, I. J. Nagrath, "Robotics and Control", **TMH Education Pvt.**

B23-PCA-414	Neural Network and Fuzzy Logic						
Lecture	**Tutorial**	**Practical**	**Credit**	**End Semester Exam**	**Internal Assessment**	**Total**	**Time**
3	**-**	**-**	**3**	**70**	**30**	**100**	**3 Hrs.**
Purpose	Neural Networks and fuzzy logic systems enable computers to solve complex, non-linear, and uncertain real-world problems by mimicking human-like intelligence, learning capabilities, and reasoning.						
Course Outcomes (CO)							
CO1	Explain the fundamental concepts of neural networks and fuzzy logic systems.						
CO2	Apply neural network learning algorithms for classification and regression problems.						
CO3	Design and implement multilayer neural networks for real-world applications.						
CO4	Construct fuzzy inference systems for decision-making and control applications. Develop hybrid neuro-fuzzy systems for intelligent problem solving.						

Unit 1
Fundamentals of Neural Networks

Introduction to Neural Networks, Biological Neuron vs Artificial Neuron, McCulloch–Pitts Model, Activation Functions, Perceptron Model, Learning Rules: Hebbian, Delta Rule, Adaline and Madaline, Limitations of Single Layer Networks.

Unit 2
Multilayer Neural Networks Learning Basics

Multilayer Perceptron (MLP), Backpropagation Algorithm, Gradient Descent and Error Minimization, Overfitting and Regularization, Introduction to Convolutional Neural Networks (CNN), Introduction to Recurrent Neural Networks (RNN), Applications of Deep Learning.

Unit 3
Fundamentals of Fuzzy Logic

Introduction to Fuzzy Logic, Crisp Sets vs Fuzzy Sets, Membership Functions, Fuzzy Set Operations, Linguistic Variables, Fuzzy Relations, T-Norms and T-Conorms, Fuzzy Composition.

Unit 4
Fuzzy Inference Systems and Neuro-Fuzzy Models

Fuzzification and Defuzzification Methods, Mamdani Fuzzy Model, Sugeno Fuzzy Model, Rule Base Design, Adaptive Neuro-Fuzzy Inference System (ANFIS), Applications in Control Systems, Pattern Recognition and Decision Support.

Text Books

1. Neural Networks and Learning Machines – Simon Haykin, Pearson.
2. Fuzzy Logic with Engineering Applications – Timothy J. Ross, Wiley.
3. Neuro-Fuzzy and Soft Computing – J. S. R. Jang, C. T. Sun & E. Mizutani, Pearson.

Reference Books

1. Artificial Neural Networks – B. Yegnanarayana, PHI.
2. Fuzzy Sets and Fuzzy Logic – George J. Klir and Bo Yuan, PHI.

B23-PCA-416	Healthcare and Analytics						
Lecture	**Tutorial**	**Practical**	**Credits**	**End Semester Exam**	**Internal Assessment**	**Total**	**Time (Hrs.)**
3	**0**	**0**	**3**	**70**	**30**	**100**	**3 Hrs**
Purpose	This course's objective is to acquaint students with the healthcare system so they may comprehend its uses in analytics and simulations.						
Course Outcomes							
CO 1	Knowledge about basic overview of data analytics in healthcare.						
CO 2	The student will know about algorithms regarding phenotyping.						
CO 3	The student will learn about know the importance of clinical trials and prediction models.						
CO 4	To know how to use pervasive health analysis.						

UNIT I

Introduction to Healthcare Data Analytics- Electronic Health Records–Components of EHR- Coding Systems- Benefits of EHR. Barrier to Adopting EHR and its main Challenges.

UNIT-II

Algorithms regarding phenotyping: Phenotyping Algorithms. Challenges in Healthcare Data Analysis, Acquisition Challenges, Pre-processing, Transformation, Social Media Analytics for Healthcare.

UNIT-III

Advanced Data Analytics for Healthcare: Review of clinical trials, Prediction Models. Statistical Prediction Models, Alternative Clinical Prediction Models, Survival Models, Predictive Models for Integrating Clinical and Genomic Data.

UNIT-IV

Pervasive health analysis: Data Analytics for Pervasive Health, Fraud Detection in Healthcare, Pharmaceutical Discoveries and Clinical Decision Support Systems.

Text Books/Reference Books:

1. Chandan K. Reddy and Charu C Aggarwal, "*Healthcare data analytics*", Taylor & Francis, 2015
2. Hui Yang and Eva K. Lee, "*Healthcare Analytics: From Data to Knowledge to Healthcare Improvement*, Wiley, 2016.

Note: The paper setter will set the paper as per the question paper template provided.