Detailed Syllabus

Course Code	**23B61CA221**	**Semester** Even	**Semester** 4th **Session** 2024 -2025 **Month from** January to May 2025
Course Name	**Software Engineering**		
Credits	3-0-0	**Contact Hours**	3

Faculty (Names)	**Coordinator(s)**	Dr. Shweta Rani
	Teacher(s) (Alphabetically)	Preeti Mittal, Dr. Shweta Rani

COURSE OUTCOMES		**COGNITIVE LEVELS**
CO1	Explain the fundamental principles of software engineering.	Understand (Level 2)
CO2	Prepare software requirements specification by identifying the software requirements.	Apply (Level 3)
CO3	Apply UML modeling for software design	Apply (Level 3)
CO4	Apply various software metrics to analyze the project and to build a software project plan.	Apply (Level 3)
CO5	Design software test cases by applying software testing principles	Create (Level 6)

Module No.	**Title of the Module**	**Topics in the Module**	**No. of Lectures for the module**
1.	Introduction to Software Engineering	Introduction to software engineering Principles, Software Cost, Software Process, Software Development Life Cycle, Software process models (build and fix model, waterfall model, Incremental process model, Evolutionary- Prototype and Spiral models, Agile Models	6
2.	Requirement Engineering	Types of requirement: Functional, Non-functional, Requirement Elicitation, Analysis, Specification, SRS, Requirement Verification and Validation.	**9**
3.	Software Design	Use case diagram, State diagram, Activity Diagram, Class Diagram, Sequence diagram. Design Modularity: Coupling Cohesion.	**8**
4.	Software Metrics and Software Project Planning	Size-Oriented Metric, Function-oriented Metric, Halstead's Software Metric, Information Flow Metric, Object-oriented Metric, Class-Oriented Metric, COCOMO Model. Project planning, Project Scheduling: network diagram, Gantt Chart, CPM and PERT.	**7**
5.	Software Testing	Verification, Validation, Code Inspection, Test Plan, Test case specification, Level of Testing: Unit, Integration and System Testing White-Box Testing, Basis Path Testing, Control Structure	**9**

		Testing: Condition Testing, Data Flow Testing Black-Box Testing: Equivalence class partitioning, Boundary Value Analysis, Decision table testing, Mutation Testing and regression Testing	
6.	Software Maintenance	Software Maintenance, Maintenance Models, Re-engineering and Reverse Engineering	**3**
		Total number of Lectures	**42**

Evaluation Criteria

Components	**Maximum Marks**
T1	20
T2	20
End Semester Examination	35
TA	25 (Assignments /Quiz/ Mini Project : 20 Attendance : 05)
Total	**100**

Project based learning: Each student works on different case study in Tutorial and Assignments. They utilize the concepts taught in lecture and develop project in a group of 3-4.

The course emphasized on the skill development for employability in software industry by engaging students on Software Development methodologies. Various activities are carried out to enhance the student's software development skills. Some of them are study of various software process models and their applicability, progress tracking, size estimation techniques, software testing strategies, etc.

Recommended Reading material: Author(s), Title, Edition, Publisher, Year of Publication etc. (Text books, Reference Books, Journals, Reports, Websites etc. in the IEEE format)

Text Book(s):	
1.	Roger S. Pressman, "Software Engineering: A practitioner approach", Fifth Edition, Mc=Graw Hill.
2.	Sommerville, "Software Engineering", Seventh Edition - Addison Wesley.
Reference Book(s):	
3.	Grady Booch, James Rumbaugh, Ivar Jacobson, "The Unified Modeling Language User Guide", Addison Wesley, Reading, Massachusetts, May 2005
4.	Richard Thayer, "Software Engineering Project Management", Second Edition -Wiley-IEEE Computer Society Press.
5.	B. Bezier, "Software Testing Techniques", Second Edition- International Thomson Computer Press.
6.	Pankaj Jalote, "An Integrated Approach to Software Engineering" Third addition, Springer Press
7	Rajib Mall, "Fundamentals of Software Engineering", PHI
8	K. K. Aggarwal and Yogesh Singh, Software Engineering, New Age International Publishers.

CO-PO and CO-PSO Mapping:

CO Code	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2
CO1	3	2	2	1	1	1	2	3	2
	This CO highly requires basic knowledge of computing	This CO is moderately applied in identifying and analyzing the problems.	This CO moderately requires to develop computer applications .	This CO slightly requires in tools usage and resource.	This CO slighly requires to develop communicate effective reports etc.	This CO slightly requires in project management and team work.	This CO moderately mapped as applying domain knowledge in life long learning.	This CO highly requires knowledge of software development.	This CO moderately requires knowledge of soft skills and language proficiency
CO2	2	3	3	2	3	2	2	3	3
	This CO moderately requires applying domain knowledge in study of mathematics and computing	This CO highly requires in formulate the problem identification and its solution	This CO highly requires to develop computer applications by collecting requirements	This CO moderately requires in tools usage and resource	This CO highly requires in creating effective reports, documents and presentation	This CO Moderately requires in project management and team work.	This CO moderately mapped as applying domain knowledge in lifelong learning	This CO highly requires knowledge of software development	This CO highly requires knowledge of soft skills and language proficiency
CO3	1	3	3	3	3	2	1	3	2
	This CO slightly requires applying domain knowledge in study of mathematics and	This CO highly requires in formulate the problem identification and its solutio	This CO highly requires to develop computer applications by collecting requirements	This CO moderately requires in tools usage and resource	This CO highly requires in creating effective reports, documents and	This CO Moderately requires in project management and team work.	This CO slighly mapped as applying domain knowledge in lifelong learning	This CO highly requires knowledge of software development	This CO moderately requires knowledge of soft skills and language proficiency

	computing	n			presentation				
CO4	1	3	2	3	2	3	1	3	1
	This CO moderately requires applying domain knowledge in study of mathematics and computing	This CO highly requires in formulate the problem identification and its solution	This CO highly requires in formulate the problem identification and its solution	This CO moderately requires in tools usage and resource	This CO moderately requires in creating effective reports, documents and presentation	This CO highly requires in project management and team work	This CO slighly mapped as applying domain knowledge in lifelong learning	This CO highly requires knowledge of software development ng	This CO slightly requires knowledge of soft skills and language proficiency
CO5	1	2	2	1	2	3	1	3	1
	This CO moderately requires applying domain knowledge in study of mathematics and computing	This CO moderately requires in formulate the problem identification and its solution	This CO moderately requires in formulate the problem identification and its solution	This CO slightly requires in tools usage and resource	This CO moderately requires in creating effective reports, documents and presentation	This CO highly requires in project management and team work	This CO slightly mapped as applying domain knowledge in lifelong learning	This CO highly requires knowledge of software development	This CO slightly requires knowledge of soft skills and language proficiency
Avg	**1.6**	**2.6**	**2.4**	**2**	**2.2**	**2.2**	**1.4**	**3**	**1.8**

Detailed Syllabus
Lecture-wise Breakup

Course Code	23B61CA222	**Semester** EVEN **(specify Odd/Even)**	**Semester** IV **Session** 2024 -2025 **Month from** Jan'25 **to** June'25
Course Name	Operating System		
Credits	3-0-0	**Contact Hours**	3

Faculty (Names)	**Coordinator(s)**	Mr. Noor Mohammad
	Teacher(s) (Alphabetically)	Mr. Noor Mohammad, Ms. Shagun Gupta

COURSE OUTCOMES		**COGNITIVE LEVELS**
CBAC208.1	Discuss the architecture and role of operating systems in managing hardware and software resources.	Understand (Level-2)
CBAC208.2	Demonstrate concurrency techniques for efficient process and thread management.	Apply (Level-3)
CBAC208.3	Explain memory management techniques to optimize resource allocation, improve system performance, and ensure efficient execution of processes.	Apply (Level-3)
CBAC208.4	Compare various disk scheduling algorithms and utilize IO management techniques.	Analyze (Level-4)
CBAC208.5	Recommend OS components and design choices for real-world applications.	Evaluate (Level-5)

Module No.	**Title of the Module**	**Topics in the Module**	**No. of Lectures for the Module**
1	Introduction to Operating Systems	Operating System: Objectives & Functions, Evolution of Operating Systems: Serial Processing, Simple Batch Systems, Multi-programmed Batch Systems, Time-Sharing Systems, System Calls.	3
2	Process Management	Process Management: Process Scheduling and Operations; Inter-process Communication, Communication in Client–Server Systems, Process Synchronization, Critical-Section Problem, Peterson's Solution, Semaphores, Synchronization. Threads: Multicore Programming, Multithreading Models, Thread Libraries, Implicit Threading, Threading Issues.	8
3	CPU Scheduling & Deadlocks	CPU Scheduling: Scheduling Criteria and Algorithms (FCFS, SJF, Round Robin, Priority Scheduling), Thread Scheduling, Multiple Processor Scheduling, Real-Time CPU Scheduling. Deadlocks: Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Avoidance and Detection, Recovery from Deadlock	10
4	Memory Management	Contiguous Memory Allocation, Swapping, Paging, Segmentation, Demand Paging, Page Replacement Algorithms, Allocation of Frames, Thrashing, Memory-Mapped Files.	6
5	File Systems	File System: Concepts and Architecture, Directory Structure (Single-Level, Two-Level, Tree-Structured,	6

		Acyclic Graph), File Allocation Methods: Contiguous, Linked, Indexed, File Access Methods, File System Mounting, Free-Space Management, efficiency and performance, comparison of UNIX and windows.	
6	Storage Management	Mass-Storage Structure, Disk Structure, Scheduling & Algorithms (FCFS, SSTF, SCAN, C-SCAN) and Management, RAID Structure, Disk Formatting and Partitioning.	4
7	Input/Output Systems	I/O Hardware, Device Drivers, I/O Scheduling and Buffering, Application I/O Interface, Kernel I/O Subsystem, Transforming I/O Requests to Hardware Operations.	4
8	Virtual Machines	VMs: Para-virtualization, Type 0, Type 1, Type 2	1
		Total number of Lectures	**42**

Evaluation Criteria

Components	**Maximum Marks**
T1 Examination	20
T2 Examination	20
T3 Examination	35
TA	
1. Attendance	05
2. Assignments (10), PBL (10)	20
Total	**100**

Project based learning: Demonstrate the application of key operating system concepts, such as process management, memory management, file systems, and I/O systems, by developing a group project. In teams of up to 3 students, create a simulation or small-scale implementation of an operating system component, such as a process scheduler, memory manager, or file system, to showcase the practical application of the concepts covered in class.

Recommended Textbooks: Author(s), Title, Edition, Publisher, Year of Publication etc.

1.	A. Silberschatz, P. B. Galvin, and G. Gagne, *Operating System Concepts Essentials, 2nd Edition*. Wiley Global Education, 2013.
2.	W. Stallings, *Operating systems: internals and design principles*. Upper Saddle River, N.J.: Pearson/Prentice Hall, 2009.

Recommended Reference Books: Author(s), Title, Edition, Publisher, Year of Publication etc.

1.	A. S. Tanenbaum and A. S. Woodhull, *Operating systems design and implementation*. Upper Saddle River, Nj Prentice-Hall, 1997.
2.	A. S. Tanenbaum, *Modern Operating Systems*. Pearson, 2001.
3.	G. J. Nutt, *Operating Systems*. Addison-Wesley Longman, 2000.
4.	C. P. Crowley, *Operating Systems*. McGraw-Hill Science, Engineering & Mathematics, 1996.

CO-PO-PSO Mapping

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PSO1**	**PSO2**
CO1	1						2		1
	Slightly mapped as students gain the knowledge on basics of operating system.						Moderately mapped, as students apply the operating system concepts in their professional development.		Slightly mapped as OS concepts can be used for the compatibility of application development.

CO2	2	2					2		2
	Moderately mapped as it involves knowledge of mathematics and computing fundamentals.	Moderately mapped as it involves identification and formulation of solution to scheduling problems.					Moderately mapped, as students apply the concepts of threads and process synchronization learnt in continuing professional development.		Moderately mapped as the concepts of threads and process synchronization can be used for the design of operating system**.**
CO3	2	2	1				2		2
	Moderately mapped as the concept of memory management and resource allocation utilized knowledge of mathematics and computing fundamentals.	Moderately mapped as it involves the identification of problems related to resource allocation and formulating their solutions to improve system performance.	Slightly as the students can apply the knowledge to design solutions to complex engineering problems related to memory management.				Moderately mapped, as the learned concepts of memory management techniques are applied to real-world scenarios.		Moderately mapped as the concepts of memory management techniques can be used in application development scenarios.
CO4	2	2					2		
	Moderately mapped as the concept of disk scheduling involves knowledge of mathematics and computing fundamentals.	Moderately mapped as the concept of I/O and disk management involves identification and analysis of problems.					Moderately mapped, as the learned concepts of I/O & disk management are applied to real-world scenarios.		
CO5		2					2		2
		Moderately as choosing OS design and components involves identification and analysis of problems.					Moderately as the learned concepts of OS can be used to choose suitable design.		Moderately as the learned concepts of OS can be used to choose suitable design for application development.
Avg.	**1.75**	**2**	**1**				**2**		**1.75**

Detailed Syllabus
Lecture-wise Breakup

Course Code	**23B12CA211**	**Semester** Even	**Semester** 4th **Session** 2024 -2025 **Month from** January to May 2025
Course Name	**FUNDAMENTALS OF AI**		
Credits	3-0-0	**Contact Hours**	3

Faculty (Names)	**Coordinator(s)**	Dr. Ruchin Gupta
	Teacher(s) (Alphabetically)	Dr. Ruchin Gupta

COURSE OUTCOMES		**COGNITIVE LEVELS**
CO1	Understand and apply the core concepts, methodologies, and problem-solving strategies in Artificial Intelligence.	(Apply) Level 3
CO2	Illustrate advanced search and optimization techniques to solve complex AI problems in diverse domains.	(Apply) Level 3
CO3	Explore knowledge representation, reasoning techniques, and probabilistic models to enhance decision-making.	(Analyze) Level 4
CO4	Develop AI-driven solutions using logic programming, and automated planning techniques.	(Create) Level 6

Module No.	**Title of the Module**	**Topics in the Module**	**No. of Lectures for the module**
1.	Introduction to AI	Introduction to AI, Types of AI, approaches to AI learning, Applications of AI, Intelligent Agents in AI, Turning test in AI.	**4**
2.	Uninformed search Algorithms	BFS, DFS, DLS, Iterative Deepening DFS, DLS	**6**
3.	Heuristic Search Techniques	Greedy BFS, A*, Heuristic functions.	**4**
4.	Local & Adversarial search	Optimization problems, hill climbing search, simulated annealing, local beam search, genetic algorithms. Min-Max Algorithm-Optimal decisions in games, alpha-beta pruning	**7**
5.	Constraint satisfaction problems	Inference in CSPs, back tracking search for CSPs.	**4**

6.	Knowledge and reasoning	knowledge-based agents, Knowledge Representation Techniques, propositional logics, and horn clauses, first order logic, Rules Inference in first order logic, unification and lifting, forward & backward chaining.	**6**
7.	Reasoning in AI	Probabilistic reasoning in AI, Bayes theorem in AI, Bayesian Belief Network. **Case study:** AI in Healthcare, AI in Agriculture, Engineering Applications in AI, Robotics & AI, Cognitive AI.	**5**
8.	Logic programming and planning techniques	Logic based representations (PL, FoL) and inference, Logic Programming: Prolog. Rule based representations, forward and backward chaining, matching algorithms. Planning Techniques: Goal Stack Planning, Constraint posting.	**6**
		Total number of Lectures	**42**

Evaluation Criteria

Components	**Maximum Marks**
T1	20
T2	20
End Semester Examination	35
TA	25
(Mini Project (10), Attendance (5), Assignment/Quiz/Programming Contest (10))	
Total	**100**

Project based learning: Each student works on different case study in Tutorial and Assignments. They utilize the concepts taught in lecture and develop project in a group of 3-4.

Recommended Reading material: Author(s), Title, Edition, Publisher, Year of Publication etc. (Text books, Reference Books, Journals, Reports, Websites etc. in the IEEE format)

Text Book(s):

1.	Russel and Norvig, 'Artificial Intelligence,' third edition, Pearson Education, PHI, (2015)
2.	Elaine Rich & Kevin Knight, 'Artificial Intelligence,' 3rd Edition, Tata McGraw-Hill Edition, Reprint (2008)
3.	Data science Handbook – Field cady- Wiley.

Reference Book(s):

1.	Patrick Henry Winston, 'Artificial Intelligence,' Pearson Education (2003)
2.	G. Luger, W. A. Stubblefield, "Artificial Intelligence", Third Edition, Addison- Wesley, (2007)
3.	William F. Clocksin, Christopher S. Mellish-Programming in Prolog-Springer (2003)
4.	Introducing Data science by Davy Cielen, Arno D.B. Meysmen, Mohamed Ali

CO-PO and CO-PSO Mapping:

CO Code	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2
CO1	3	3	2	1			1	3	1
CO2	3	3	2	1			1	3	1
CO3	3	3	2	1			1	3	1
CO4	3	3	2	1			1	3	1
AVG	**3**	**3**	**2**	**0.25**			**1**	**3**	**1**

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2
CO1	3	3	2	1	NA	NA	1	3	1
	CO involves the application of core AI concepts	CO emphasizes problem-solving and analysis in the context of AI	CO is linked to the design and development of AI solutions	It allows to select and apply appropriate techniques in AI			AI and related fields are rapidly evolving	CO is strongly related to developing skills in AI	CO contributes to the development of programming skills, and problem-solving in real-world scenarios
CO2	3	3	2	1	NA	NA	1	3	1
	CO involves the application of core AI concepts	CO emphasizes problem-solving and analysis in the context of AI	CO is linked to the design and development of AI solutions	It allows to select and apply appropriate techniques in AI			AI and related fields are rapidly evolving	CO is strongly related to developing skills in AI	CO contributes to the development of programming skills, and problem-solving in real-world scenarios
CO3	3	3	2	1	NA	NA	1	2	1
	CO involves the application of core AI concepts	CO emphasizes problem-solving and analysis in the context of AI	CO is linked to the design and development of AI solutions	It allows to select and apply appropriate techniques in AI			AI and related fields are rapidly evolving	CO is strongly related to developing skills in AI	CO contributes to the development of programming skills, and problem-solving in real-world scenarios
CO 4	3	3	2	1	NA	NA	1	3	1
	CO involves the application of core AI concepts,	CO emphasizes problem-solving and	CO is linked to the design and development	CO gives slight exposure to tools /techniques.			AI and related fields are rapidly evolving	CO is strongly related to developing skills in AI	CO contributes to the development

		analysis in the context of AI	of AI solutions						of programming skills, and problem-solving in real-world scenarios

Detailed Syllabus
Lecture-wise Breakup

Course Code	23B12CA213	**Semester** EVEN **(specify Odd/Even)**	**Semester** IV **Session** 2024 -2025 **Month from** Jan'25 **to** June'25
Course Name	Introduction to Big Data and Data Analytics		
Credits	3-0-0	**Contact Hours**	3

Faculty (Names)	**Coordinator(s)**	Mr. Noor Mohammad
	Teacher(s) (Alphabetically)	Mr. Noor Mohammad

COURSE OUTCOMES		**COGNITIVE LEVELS**
CBAC209.1	Discuss the concept and significance of Big Data in contemporary business and technology.	Understand (Level 2)
CBAC209.2	Demonstrate understanding of data frameworks for storage, processing, and analytics.	Apply (Level 3)
CBAC209.3	Construct queries to perform database operations.	Apply (Level 3)
CBAC209.4	Use programming skills for data analysis and visualization.	Apply (Level 3)
CBAC209.5	Explore basic machine learning models to uncover patterns and insights from data.	Analyze (Level 4)

Module No.	**Title of the Module**	**Topics in the Module**	**No. of Lectures for the Module**
1	Introduction to Big Data	Overview and Definition, Characteristics of Big Data (5Vs), Basic Terminologies: Datasets, Data Analysis, Data Analytics (Descriptive, Diagnostic, Predictive, Prescriptive) Business Intelligence (BI), Key Performance Indicators (KPI), Different Types of Data: Structured, Unstructured, Semi-structured.	4
2	Big Data Ecosystem and Technologies	File Systems and Distributed File Systems, Hadoop Architecture and HDFS, MapReduce Programming Model, Overview of NoSQL Databases, Introduction to Spark and its Components, Sharding & Replication, CAP Theorem, Comparison of Traditional RDBMS vs. NoSQL Databases for Big Data.	8
3	Data Preprocessing and Cleaning	Data Preprocessing Overview: Cleaning, Integration, Transformation, Handling Missing Data, Duplicates, and Inconsistent Data, Feature Selection and Feature Engineering Techniques, Pre-processing a Sample Dataset using Python Libraries (Pandas and NumPy)	6
4	NoSQL Databases	Overview of NoSQL Databases and their types (Key-Value, Columnar, Document, Graph databases), Introduction to MongoDB: Basics, Features and Applications, CRUD Operations, Importing and Exporting Data, MongoDB Integration with Big Data Tools Using PyMongo.	8
5	Data Visualization	Data Visualization Importance and Tools: Tableau, Power BI, Python (Matplotlib, Seaborn), Types of Visualizations: Bar Charts, Line Charts, Scatter Plots, Heatmaps, Dashboards.	6

6	Introduction to Machine Learning	Supervised, Unsupervised, Reinforcement Learning, Steps in ML: Data Collection, Pre-processing, Training and Evaluating a Model, Algorithms: Regression, Clustering, Decision Tree.	10
		Total number of Lectures	**42**

Evaluation Criteria

Components	**Maximum Marks**
T1 Examination	20
T2 Examination	20
T3 Examination	35
TA	
1. Attendance	05
2. Assignments (10), PBL (10)	20
Total	**100**

Project based learning: Demonstrate the application of key concepts in Big Data and Data Analytics, such as data collection, data preprocessing, data visualization, and implementing analytics models, by developing a group project. In teams of up to 3 students, create a real-world simulation or small-scale implementation of a machine learning model. For example, build a data processing pipeline using tools like Hadoop or Spark, perform sentiment analysis on social media data, or develop a dashboard for visualizing large datasets. This mini-project will showcase the practical application of Big Data and Analytics concepts learned in class.

Recommended Textbooks: Author(s), Title, Edition, Publisher, Year of Publication etc.

1.	N. Marz and J. Warren, *Big Data: principles and best practices of scalable real-time data systems*. Shelter Island, Ny: Manning, 2015.
2.	P. C. Bruce, A. Bruce, and P. Gedeck, *Practical statistics for data scientists: 50+ essential concepts using R and Python*. Sebastopol, Ca: O'reilly Media, Inc, 2020.

Recommended Reference Books: Author(s), Title, Edition, Publisher, Year of Publication etc.

1.	T. Erl, W. Khattak, and P. Buhler, *Big data fundamentals : concepts, drivers & techniques*. Boston: Prentice Hall ; Vancouver, Bc, 2016.
2.	C. N. Knaflic, *Storytelling with Data: A Data Visualization Guide for Business Professionals*. Hoboken, New Jersey: Wiley, 2015.
3.	W. Mckinney, *Python for data analysis : data wrangling with pandas, NumPy, and IPython*. O'reilly, 2017.
4.	T. E. White, *Hadoop : the definitive guide*. Beijing Etc.: O'reilly Media, 2012.

CO-PO-PSO Mapping

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PSO1**	**PSO2**
CO1			1				2	1	
			Slightly mapped with designing of applications utilizing Big Data.				Moderately mapped, as the concepts of Big Data and its analytics play a significant role in enhancing career growth by fostering lifelong learning and professional development	Slightly mapped as the student will be able to understand the concept of Big Data.	
CO2	1	2					2	2	

	Slightly mapped as it contains the computing fundamentals behind the architecture of Hadoop.	Moderately mapped as it contains the analysis and formulation for efficient storage & processing.					Moderately mapped, as the learnt concept of working of Hadoop and NoSQL can enhance career growth.	Moderately mapped as the student will understand the concept of Hadoop and NoSQL database.	
CO3		2	2	2			2	2	1
		Moderately mapped as writing queries involve analysis of problem.	Moderately mapped as the knowledge of MongoDB's queries can be used to design applications.	Moderately mapped as queries will execute on open-source software.			Moderately mapped, as writing queries in MongoDB requires continuous learning, leads to the development of lifelong learning skills.	Moderately mapped as the knowledge of MongoDB can be used to design AI/ML applications.	Slightly mapped the concept of writing queries can be utilized in mobile/web application development.
CO4		2	2	2			2	2	2
		Moderately mapped as data visualization involves analysis.	Moderately mapped as the knowledge of data visualization using Python library can used in development of an application.	Moderately mapped as it utilizes the libraries of Python.			Moderately mapped, as the knowledge of Python leads to the development of lifelong learning skills.	Moderately mapped as the concept of data visualization can be used in AI/ML application.	Moderately mapped as it involves the programming proficiency using Python.
CO5	2	2	2	2			2	2	2
	Moderately mapped as machine learning models involve computing fundamentals.	Moderately mapped as designing ML models involves the analysis of problems.	Moderately mapped as it involves the designing of ML models.	Moderately mapped as it involves open-source software.			Moderately mapped, as the knowledge of designing models leads to the development of lifelong learning skills	Moderately mapped as the concept of ML models are utilized in AI/ML application	Moderately mapped as the ML models can integrate with web/mobile application.
Avg.	**1.5**	**2**	**1.75**	**2**			**2**	**1.8**	**1.67**

Detailed Syllabus
Lecture-Wise Breakup

Course Code	23B12CA215	**Semester IV(EVEN) (specify Odd/Even)**	**Session 2024-25 Month from** January to December
Course Name	Introduction to Cryptography and its Applications		
Credits	3-0-0	**Contact Hours**	3
Faculty(Names)	**Coordinator(s)**	Dr. Mradula Sharma	
	Teacher(s) (Alphabetically)	Dr. Mradula Sharma	

COURSE OUTCOMES		**COGNITIVE LEVES**
CBAC213.1	Describe the principles and workings of cryptographic algorithms.	Understand Level (Level2)
CBAC213.2	Explain symmetric and asymmetric encryption methods.	Understand Level (Level2)
CBAC213.3	Provide cryptographic protocols to enhance information security applications.	Apply Level (Level 3)
CBAC213.4	Assess the security needs of an application and suggest appropriate cryptographic solutions.	Analyze Level (Level 4)

Module No.	**Title of the Module**	**Topics in the Module**	**No. of Lectures for the module**
1.	Introduction to Cryptography	Overview of Cryptography and Its Necessity Terminology and Fundamental Concepts Historical Ciphers and Their Evolution, Traditional Encryption Techniques: substitutions and Transposition Techniques	8
2.	Symmetric Key Cryptography	Block Ciphers and Stream Ciphers Data Encryption Standard (DES) and Advanced Encryption Standard (AES), Modes of Operation	10
3.	Asymmetric Key Cryptography	Principles of Public Key Cryptography RSA Algorithm: Setup, Encryption, Decryption Key Management and Distribution	8
4.	Cryptographic Hash Functions and Digital Signatures	Hash Functions: Properties and Examples Applications of Hashing: Data Integrity, Password Storage Digital Signatures and Their Importance Public Key Infrastructure (PKI)	8
5.	Cryptography in Practice	Protocols: SSL/TLS, SSH, IPSec Cryptographic Standards and Their Implications Recent Advances in Cryptography: Quantum Cryptography, Post-Quantum Cryptography Ethical and Legal Considerations in	8

		Cryptography	
		Total number of Lectures	42
Evaluation Criteria			
Components		**Maximum Marks**	
T1		20	
T2		20	
End Semester Examination		35	
TA (Project Based Learning(10), Attendance(10), Quiz/Assignment(5))		25	
Total		**100**	
In this **project-based learning activity**, each student pair will select a cryptography algorithm, such as DES or AES, and design a project to address a specific real-world security challenge, applying and analyzing their chosen algorithm to enhance data protection and integrity in practical scenarios.			
Text Book:			
1	William Stallings, "Cryptography and Network Security: Principles and Practice," 7th Edition, Pearson Education.		
2	Jonathan Katz and Yehuda Lindell, "Introduction to Modern Cryptography," 2nd Edition, Chapman and Hall/CRC.		
Reference Book:			
1	Bruce Schneier, "Applied Cryptography: Protocols, Algorithms, and Source Code in C," 2nd Edition, Wiley.		
2	Niels Ferguson, Bruce Schneier, and Tadayoshi Kohno, "Cryptography Engineering: Design Principles and Practical Applications," Wiley.		

COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PSO 1**	**PSO2**
CBAC 213.1	1 Basics of cryptography	1 Analyze the security issues	1 Use cryptography in day to day works		2 Apply cryptography knowledge to tools	2 Use cryptography in network communication	1 Lifelong learning of Security essentials	2 Use security knowledge to build secure systems	2 Apply knowledge gained in various hackathons
CBAC 213.2	3 Symmetric/ Asymmetric key cryptography use in Internet and network communication	1 Apply knowledge of ciphers to build systems	2 Build secure network applications		2 Decide which cipher suite to use in network applications	2 Asses cryptography based security measures		2 Usage of symmetric key encryption systems	2 Apply various encryption knowledge in various online competitions
CBAC 213.3	2 Basics of Cryptograph	1 Use cryptogra	3 Analyzing Hashing		2 Using PKI to gather	2 Use security protocols to		3 Build netwo	3 Use of Hash

	y Protocols	phy Protocols to enhance the security	and digital signature		information	identify vulnerability in systems.		rk scanning tools	algorithms in competitions
CBAC 213.4	2 Cryptography in Practice		2 Assess Recent Advances in Cryptography	1 Compare the Various applications of Cryptography	2 Understand Ethical and Legal Considerations in Cryptography	2 Defend systems against malpractice		2 Defending systems against malware	2 Ethical and Legal Considerations in Cryptography

Detailed Syllabus
Lecture-wise Breakup

Course Code **NBA Code**	23B12CA216	**Semester: 4th**	**Semester:** 4th **Session:** EVEN-2025 **Months:** January-June
Course Name	Machine Learning		
Credits	**3-0-0**	**Contact Hours**	**3**

Faculty (Names)	**Coordinator(s)**	Dr. Imran Rasheed
	Teacher(s) (Alphabetically)	Dr. Imran Rasheed

COURSE OUTCOMES		**COGNITIVE LEVELS**
CO 1	Understand the fundamental concepts of Machine Learning and its real-world applications.	Understand (Level 2)
CO 2	Apply techniques for data exploration, analysis, and manipulation to Kaggle datasets. Use tools and libraries to import, summarize, and visualize data effectively.	Apply (Level 3)
CO 3	Demonstrate supervised machine learning algorithms and methodologies to address challenges and solve practical problems.	Apply (Level 3)
CO 4	Illustrate unsupervised machine learning and the various dimensionality reduction techniques.	Apply (Level 3)
CO 5	Examine the various concepts related to the training of the neural network model.	Analyze (Level 4)

Module No.	**Title of the Module**	**Topics in the Module**	**No. of Lectures for the module**
1.	Introduction to Machine Learning	Definition, Goals, and Applications of ML, Types of Learning: Supervised, Unsupervised, and Reinforcement Learning, Steps in a Machine Learning Pipeline, Challenges in Machine Learning	8
2.	Data Preprocessing and Exploration	Importance of Data in ML, Handling Missing Values, Outliers, and Duplicates, Data Scaling and Normalization, Exploratory Data Analysis (EDA), Feature Engineering and Feature Selection	8
3.	Supervised Learning Algorithms	Regression: Linear Regression, Logistic Regression, Classification: Decision Trees, k-Nearest Neighbors (k-NN), Support Vector Machines (SVM), Evaluation Metrics: Accuracy, Precision, Recall, F1 Score, Confusion Matrix	8
4.	Unsupervised Learning Algorithms	Clustering: K-Means, Hierarchical Clustering, Dimensionality Reduction: Principal Component Analysis (PCA), Singular Value Decomposition, Applications of Unsupervised Learning	8

5.	Introduction to Neural Networks and Deep Learning	Perceptron's, Multilayer perceptron, Gradient descent, Multilayer networks, Backpropagation Algorithm, Confusion matrix, Loss functions, Bias-variance trade-off, overfitting-underfitting, Activation functions, Introduction to Convolutional Neural Network, building blocks of CNN Case Studies: Applications of Neural Networks in Computer Vision and Natural Language Processing	10
		Total number of Lectures	**42**

Evaluation Criteria

Components	**Maximum Marks**
T1	20
T2	20
End Semester Examination	35
Attendance	05
Mini Project	20
Total	**100**

Project Based Learning: Students in a group of 3-4 will take some real-world problem and apply machine learning concepts to solve the problem in a meaning way. Students can able to understand the core logic about data handling and processing using machine learning models.

Recommended Reading material: Author(s), Title, Edition, Publisher, Year of Publication etc. (Text books, Reference Books, Journals, Reports, Websites etc. in the IEEE format)

1.	Chollet, Francois. Deep learning with Python. Simon and Schuster, 2021.
2.	Machine Learning - A Complete Exploration of Highly Advanced Machine Learning Concepts, Best Practices and Techniques by Peter Bradley, Draft2digital, 25 June 2019.
3.	Bengio, Yoshua, Ian Goodfellow, and Aaron Courville. Deep learning. Vol. 1. Cambridge, MA, USA: MIT press, 2017.

Reference Books

1.	Weidman, Seth. Deep learning from scratch: Building with python from first principles. O'Reilly Media, 2019.
2.	Alpaydin, Ethem. Introduction to machine learning. MIT press, 2020.
3.	Ravichandiran, Sudharsan. Hands-On Deep Learning Algorithms with Python: Master deep learning algorithms with extensive math by implementing them using TensorFlow. Packt Publishing Ltd, 2019.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	**3**	**2**		**2**	**1**							**2**		
CO2	**3**	**3**	**2**	**2**						**1**				
CO3	**3**	**3**	**3**	**3**	**3**	**2**			**2**	**1**				
CO4	**3**	**3**	**2**		**3**							**2**		
CO5	**3**	**3**	**2**	**3**	**3**							**2**		

Justification

CO1: Understand the fundamental concepts of Machine Learning and its real-world applications.
PO1 (Engineering Knowledge): High correlation (3) as understanding Machine Learning requires knowledge of mathematics, statistics, and computational methods.
PO2 (Problem Analysis): Moderate correlation (2) as it involves analyzing problems to apply Machine Learning concepts.
PO4 (Conduct Investigations): Moderate correlation (2) as Machine Learning involves investigating data-driven problems.
PO5 (Modern Tool Usage): Moderate correlation (2) as understanding Machine Learning involves familiarity with tools and libraries.
PO6 (Engineer and Society): Moderate correlation (2) as real-world applications highlight societal impact.
PO12 (Life-long Learning): Moderate correlation (2) as Machine Learning is a rapidly evolving field.

CO2: Apply techniques for data exploration, analysis, and manipulation to Kaggle datasets. Use tools and libraries to import, summarize, and visualize data effectively.
PO1 (Engineering Knowledge): High correlation (3) as data exploration and analysis require engineering knowledge.
PO2 (Problem Analysis): High correlation (3) as it involves identifying and analyzing data-related problems.
PO3 (Design/Development): Moderate correlation (2) as data manipulation is part of designing solutions.
PO4 (Conduct Investigations): High correlation (3) as data exploration is a key part of investigating problems.
PO5 (Modern Tool Usage): High correlation (3) as it involves using tools like Pandas, Matplotlib, and Seaborn.
PO6 (Engineer and Society): Moderate correlation (2) as data analysis can have societal implications.
PO9 (Individual and Teamwork): Moderate correlation (2) as data projects often involve teamwork.
PO10 (Communication): Moderate correlation (2) as visualizing and summarizing data requires effective communication.

CO3: Demonstrate supervised machine learning algorithms and methodologies to address challenges and solve practical problems.

PO1 (Engineering Knowledge): High correlation (3) as supervised learning requires knowledge of algorithms and methodologies.
PO2 (Problem Analysis): High correlation (3) as it involves analyzing and solving problems using supervised learning.
PO3 (Design/Development): High correlation (3) as supervised learning is used to design solutions.
PO4 (Conduct Investigations): High correlation (3) as it involves investigating and solving practical problems.
PO5 (Modern Tool Usage): High correlation (3) as it involves using tools like Scikit-learn and TensorFlow.
PO6 (Engineer and Society): Moderate correlation (2) as supervised learning has societal applications.
PO9 (Individual and Teamwork): Moderate correlation (2) as solving problems may involve teamwork.
PO10 (Communication): Moderate correlation (2) as presenting solutions requires communication skills.

CO4: Illustrate unsupervised machine learning and the various dimensionality reduction techniques.

PO1 (Engineering Knowledge): High correlation (3) as unsupervised learning requires knowledge of algorithms and techniques.
PO2 (Problem Analysis): High correlation (3) as it involves analyzing data using unsupervised methods.
PO3 (Design/Development): Moderate correlation (2) as unsupervised learning can be used to design solutions.
PO4 (Conduct Investigations): High correlation (3) as it involves investigating data patterns.
PO5 (Modern Tool Usage): High correlation (3) as it involves using tools like Scikit-learn and PCA.
PO12 (Life-long Learning): Moderate correlation (2) as unsupervised learning is an evolving field.

CO5: Examine the various concepts related to the training of the neural network model.

PO1 (Engineering Knowledge): High correlation (3) as neural networks require knowledge of mathematics and algorithms.
PO2 (Problem Analysis): High correlation (3) as it involves analyzing and solving problems using neural networks.
PO3 (Design/Development): Moderate correlation (2) as neural networks are used to design solutions.
PO4 (Conduct Investigations): High correlation (3) as it involves investigating and training models.
PO5 (Modern Tool Usage): High correlation (3) as it involves using tools like TensorFlow and PyTorch.
PO12 (Life-long Learning): Moderate correlation (2) as neural networks are a rapidly evolving field.

Detailed Syllabus
Lecture-wise Breakup

Subject Code	23B61CA223	**Semester: (specify Odd/Even):**	**Semester: 1st Session: 2024-2025 Month: January-June 2025**
Subject Name	Java Full Stack Development-I (Core Java)		
Credits	**3-1-0**	**Contact Hours**	**4**

Faculty (Names)	**Coordinator(s)**	Preeti Mittal
	Teacher(s) (Alphabetically)	Preeti Mittal and Dr. Shelendra Pal

COURSE OUTCOMES		**COGNITIVE LEVELS**
CBAC211.1	Explain the use of various programming constructs in Java using practical problems	Understand (level 2)
CBAC211.2	Describe the methods for managing multiple tasks concurrently	Understand (level 2)
CBAC211.3	Demonstrate strategies to prevent application crashes	Apply (level 3)
CBAC211.4	Illustrate input-output operations and database connectivity	Apply (level 3)

Module No.	**Subtitle of the Module**	**Topics in the Module**	**No. of Lectures for the module**
1	Fundamentals of programming in Java	The Java Virtual Machine (JVM) Naming Conventions in Java The main() method Primitive Data Types Variables Constants Reading Input using Scanner class Displaying Output Command Line Arguments Operators Control Flow	5
2	Classes and Objects in Java	Classes and Objects Class and Instance Variables Constructors Instance Methods Class Methods Passing and returning objects Method Overloading OOPS core principles - Inheritance, Polymorphism,	5

		Encapsulation and Abstraction Garbage collection	
3	Inheritance	Superclass and Subclass protected Members this and super keywords Method Overriding final variables, methods and classes	4
4	Packages and Interfaces	Packages Polymorphic behavior Abstract Classes Creating and using interfaces Default and static interface methods Functional interfaces	3
5	Enumerations and Autoboxing	Enumerations Type Wrappers Autoboxing	2
6	Generic Classes and Methods	Generic Classes Generic Methods Generic Interfaces Using Wildcard Arguments	2
7	Utility Classes	String Handling StringTokenizer Date Calendar	2
8	Exception Handling	When to Use Exception Handling Java Exception Hierarchy Checked vs. Unchecked Exceptions Catching an Exception finally Block Declaring New Exception Types throw and throws clause Stack Unwinding and Obtaining Information from an Exception Object Chained Exceptions try-with-Resources: Automatic Resource Deallocation	4
9	Java Collections API	Arrays Collection Framework Collections Interfaces – Collection, List, Map, Set Concrete Collections – ArrayList, HashMap, HashSet, Iterating through Collections	4
10	Multithreading (Concurrency)	Concepts of Multithreading Difference between process and thread Thread States and Lifecycle Creating threads using Thread class and Runnable interface Synchronization Thread Priorities Inter thread Communication	4
11	File Handling in Java	Explore java.io package Perform basic operations like create, read, update, delete on files using java.io	2

12	Java Database Connectivity (JDBC)	Introduction Establishing JDBC connection Performing create, read, update, delete operations	3
13	Lambdas & Streams	Lambda Expressions Stream basics	2
		Total number of Lectures	42

Evaluation Criteria	
Components	**Maximum Marks**
T1	20
T2	20
End Semester Examination	35
TA	25 (Attendance (5), Assignment/Tutorial/ Quiz (15), MiniProject (5))
Total	**100**

Project based learning: Create a Javaweb application in groups of maximum 4 students each, to illustrate the concepts covered in class.

Recommended Reading material: Author(s), Title, Edition, Publisher, Year of Publication etc. (Text books, Reference Books, Journals, Reports, Websites etc. in the IEEE format)

	Text Books
1	Herbert Shildt: *Java: The Complete Reference*, 12th Edition, McGraw-Hill, 2021.
2	Y. Daniel Liang: *Intro to Java Programming, Comprehensive Version*, 10thEdition, Pearson, 2018.
	Reference Books
1	Kathy Sierra, Bert Bares & Trisha Gee: *Head First Java,* 3rd Edition, O'Reilly Media, 2022.
2	Paul Deitel and Harvey Deitel: *Java: How to Program, Late Objects*, 11th Edition, Pearson, 2021.

CO-PO and CO-PSO Mapping:

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PSO1**	**PSO2**
CBAC211.1	2	2	2	-	1	1	1	1	2
	Moderately be able to apply knowledge of basic programming structures in Java to provide solutions for complex computer applications	Moderately be able to identify and analyze problems in applied computer science using various programming structures in Java	Design computer applications that meet the specified societal, health, safety, legal and cultural needs with appropriate consideration to ethics, environment and sustainability using programming structures in Java to a moderate extent		Communicate effective reports, design documentation and make effective presentations using various programming structures available in Java to a slight extent	Slightly engage in project management and team work	Slightly recognize the need for and develop the ability to engage in continuous learning to a moderate extent and be apprised of additional features being introduced in newer version of Java	Slightly develop proficiency in software development methodologies and tools in order to design, implement, and test solutions across major core areas of AI&ML, Cyber security and Data analytics using various programming structures available in Java	Moderately be able to develop programming efficiency in full stack development using various programming structures in Java
CBAC211.2	2	2	2		1	1	1	1	1
	Moderately be able to manage multiple tasks concurrently in Java to provide solutions for complex computer applications	Moderately be able to analyze problems in applied computer science by learning how to handle multiple tasks concurrently using Java	Design computer applications that meet the specified societal, health, safety, legal, and cultural needs with appropriate consideration to ethics, environment, and sustainability to a moderate extent with the ability to handle multiple tasks concurrently using Java		Communicate effective reports, design documentation, and make effective presentations with the ability to handle multiple tasks concurrently using Java to a slight extent	Slightly engage in project management and teamwork	Slightly recognize the need for and develop the ability to engage in continuous learning and be apprised of new methodologies being introduced	Develop proficiency in software development methodologies and tools in order to design, implement, and test solutions across major core areas of AI&ML, Cyber security and Data analytics to a slight extent by managing multiple tasks concurrently in Java	Slightly be able to develop programming efficiency in full stack development with the use of handling multiple tasks concurrently in Java

CBAC211.3	2	2	2	-	1	1	1	1	2
	Moderately be able to apply knowledge of preventing applications from crashing and giving meaningful error messages to users to provide solutions for complex computer applications	Moderately be able to identify and analyze problems in applied computer science by learning how to prevent applications from crashing using Java	Design computer applications that meet the specified societal, health, safety, legal, and cultural needs with appropriate consideration to ethics, environment, and sustainability to a moderate extent with the ability to prevent applications from crashing using Java		Communicate effective reports, design documentation, and make effective presentations	Slightly engage in project management and teamwork	Recognize the need for and develop the ability to engage in continuous learning to a slight extent and be apprised of new methodologies being introduced	Slightly develop proficiency in software development methodologies and tools in order to design, implement, and test solutions across major core areas of AI&ML, Cyber security and Data analytics with the ability to prevent applications from crashing using Java	Moderately be able to develop programming efficiency in full stack development by preventing applications from crashing and giving meaningful error messages to users
CBAC211.4	1	1	1	-	1	1	1	1	1
	Slightly be able to manage database connectivity and input-output operations in Java to provide solutions for complex computer applications	Slightly be able to analyze problems in applied computer science by learning how to handle database connectivity and input-output operations using Java	Design computer applications that meet the specified societal, health, safety, legal, and cultural needs with appropriate consideration to ethics, environment, and sustainability to a slight extent with the ability to handle database connectivity and input-output operations using Java		Communicate effective reports, design documentation, and make effective presentations with the ability to handle database connectivity and input-output operations using Java to a slight extent	Slightly engage in project management and teamwork	Slightly recognize the need for and develop the ability to engage in continuous learning and be apprised of new methodologies being introduced to perform database connectivity and input-output operations in Java	Slightly develop proficiency in software development methodologies and tools in order to design, implement, and test solutions across major core areas of AI&ML, Cyber security and Data analytics using inout-output operations and database connectivity in Java	Slightly be able to develop programming efficiency in full stack development with the use of handling database connectivity and input-output operations in Java
Average	**1.75**	**1.75**	**1.75**	**0**	**1.0**	**1.0**	**1.0**	**1.0**	**1.5**

Detailed Syllabus
Lecture-wise Breakup

Subject Code	23B31HS212	**Semester: Even**	**Semester IV Session 2024-2025 Month: January-June**
Subject Name	**LIFE SKILLS**		
Credits	**2(2-0-0)**	**Contact Hours**	**2 (2-0-0)**
Faculty (Names)	**Coordinator(s)**	Dr Harleen Kaur	
	Teacher(s) (Alphabetically)	Dr Harleen kaur Prof. Monika Suri	

CO Code	Course Outcomes	Cognitive Levels
CO 212.1	Demonstrate an understanding of the primary concepts of communication skills by classifying between hard and soft skills, and describing their importance in improving personal and professional effectiveness	*Understanding - C2*
CO 212.2	Apply effective behavioral communication methods by recognizing barriers, examining several communication types, and employing strategies to overcome obstacles for improved interpersonal interactions	*Applying - C3*
CO 212.3	Analyze organizational proficiencies by practicing self-management, time-management, and stress-management strategies to address real-world challenges, thereby improving productivity and personal well-being.	*Analyzing - C4*
CO 212.4	Critically evaluate leadership styles and team-building dynamics by leveraging theories of team formation, negotiation, and persuasion to foster teamwork and influence others within a group setting.	*Evaluating - C5*

Module No.	Title of the Module	Topics in the module	No. of Lectures for the module
1.	Communication Skills	• Introduction to Life Skills • Basic Concepts and Relevance of Life skills for Personal & Professional Life • Difference between Hard & Soft Skills • Dimensions of Life skills	4
2.	Behavioral Communication	• Definition • Process • Types • Significance • Barriers in Behavioral Communication • Ways to overcome the barriers	4
3.	Professional & Managerial Skills	• Self-Management • Time-Management • Stress-Management • Conflict-Management	8
4.	Career Management	• Problem Solving • Decision Making • Group Discussion • Interview Skills • E-mail etiquette	6
5.	Leadership & Team Building	• Team vs Group • Team Formation	6

		• Team Roles • Leadership: Definition & Types • Styles & Theories • Art of Influencing Others: Persuasion & Negotiation	
• Total number of Hours			28

Evaluation Criteria
Components Maximum Marks

Mid Term 30 (Report & Presentation: Learning Life Skills through Scriptures/ Holy Books)
End Term 40 (Written during End Term)
TA 30 (Project Based Learning /Quiz)

Total 100

Project Based Learning: Students will create their own individual Life Skills Journal based on reflection of their learnings throughout the Semester

Recommended Reading material: Author(s), Title, Edition, Publisher, Year of Publication etc. (Text books, Reference Books, Journals, Reports, Websites etc. in the IEEE format)	
1.	S. P. Robbins, *Organizational Behaviour*, 9th ed. New Delhi, India: Prentice-Hall, 2001.
2.	E. Smith, S. Hoeksema, B. Fredrickson, and G. Loftus, *Introduction to Psychology*. Belmont, CA: Thompson and Wadsworth, 2003.
3.	D. Goleman, *Working With Emotional Intelligence*. New York, NY: Bantam Books, 1998.
4.	S. Bishop, *Assertiveness Skills Training*. New Delhi, India: Viva Books, 2004.
5.	A. B. Lynn, *50 Activities for Developing Emotional Intelligence*. New Delhi, India: Ane Books, 2003
6.	S. Thiagarajan and G. M. Parker, *Teamwork and Teamplay: Games and Activities for Building and Training Teams*. San Francisco, CA: Jossey-Bass, 1999.
7.	A. Grant, *Think Again: The Power of Knowing What You Don't Know*, Viking, 2021.
8.	S. Covey, *The 7 Habits of Highly Effective People: Powerful Lessons in Personal Change*, 30th Anniversary Edition, Simon & Schuster, 2020.
9.	D. Pink, *The Power of Regret: How Looking Backward Moves Us Forward*, Riverhead Books, 2022
10.	R. Cialdini, *Influence, New and Expanded: The Psychology of Persuasion*, Harper Business, 2021.
11.	S Kumar, H Kaur, and I Rampal, *Corporate Communication Unlocked: Building Business Communication Skills*. New Delhi: Atlantic Publishers, 2024.

CO-PO-PSO Mapping

COs	**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PSO 1**	**PSO 2**
CO1	-	-	-	-	1	-	2	-	2
CO2	-	1	-	-	1	-	-	-	-
CO3	-	-	-	-	-	1	1	-	1
CO4	-	-	-	-	-	3	1	-	1
Avg	-	1.00	-	-	**1.00**	**2.00**	**1.33**	-	1.33

Detailed Syllabus
Lab-wise Breakup

Course Code	23B65CA224	**Semester:** Even	**Semester: I**V Session**:**2024 -2025 **Month from** Jan to May 2025
Course Name	Software Engineering Lab		
Credits	0-0-1	**Contact Hours**	2

Faculty (Names)	**Coordinator(s)**	Megh Singhal
	Teacher(s) (Alphabetically)	Dr. Amit Mishra, Aakriti Bhardwaj, Dr. Imran Rashid, Megh Singhal, Noor Mohammad, Dr. Shweta Rani

COURSE OUTCOMES		**COGNITIVE LEVELS**
CO1	Understand fundamental principles of software engineering and the software process models.	Understand Level (Level 2)
CO2	Develop a software requirement specification document and a suitable software project plan.	Apply Level (Level 3)
CO3	Apply design diagrams to represent the structure and functionality of the software system.	Apply Level (Level 3)
CO4	Examine the performance of software system using different testing techniques.	Analyze (Level 4)

Module No.	**Title of the Module**	**List of Experiments**	**CO**
1.	Introduction to Software Engineering Principles	Introduction to software engineering Principles (evolution, failures, changing nature of software, software myths, product, process, software crisis and need of testing), Software process models (build and fix model, waterfall model, Incremental process model, Evolutionary- Prototype and Spiral models, Agile models.	CO1
2.	Requirement Engineering and Project Planning	Types of requirement, Feasibility studies, Requirement Elicitation, Analysis, Specification, SRS, Requirement Verification and Validation. Project planning, Project Scheduling: network diagram, Gant Chart, CPM and PERT.	CO2
3.	Software Design and Software Metrics	Use case diagram, State diagram, Activity Diagram, Class Diagram, Sequence diagram. Design Modularity: Coupling Cohesion. Size-Oriented Metric, Function-oriented Metric, Halstead's Software Metric, Information Flow Metric, Object-oriented Metric, Class-Oriented Metric, COCOMO Model.	CO3

4.	Software Testing	Black box testing techniques: Equivalence class testing, Boundary value analysis, Decision table testing. White box testing: Path testing, Data flow and mutation testing, Levels of testing- unit testing, integration and system testing,	CO4

Evaluation Criteria

Components	**Maximum Marks**
Lab Test 1	20
Lab Test 2	20
Day-to-Day	60 (Evaluations, Project, Attendance)
Total	**100**

Project based learning: Each student in a group of 3-4 have to work on a mini-project, in which they will create Software Requirements Specification (SRS) document and design the software diagrams. Further, the software implementation should be followed with testing reports. This enhances the understanding of students towards different software engineering concepts and also help them during their employability.

Recommended Reading material: Author(s), Title, Edition, Publisher, Year of Publication etc. (Text books, Reference Books, Journals, Reports, Websites etc. in the IEEE format)

Text Books

1.	Pressman, Roger S. Software engineering: a practitioner's approach. Palgrave Macmillan, 2005.
2.	Sommerville , "Software Engineering" , Seventh Edition - Addison Wesley.
3.	KK Aggarwal, Software Engineering.

Reference Books

4.	Grady Booch, James Rumbaugh, Ivar Jacobson, The Unified Modeling Language User Guide, Addison Wesley, Reading, Massachusetts, May 2005
5.	Richard Thayer , "Software Engineering Project Management", Second Edition -Wiley-IEEE Computer Society Press.
6.	B. Bezier, "Software Testing Techniques", Second Edition- International Thomson Computer Press.
7.	Pankaj Jalote, "An Integrated Approach to Software Engineering" Third addition , Springer Press

BCA (BACHELOR OF COMPUTER APPLICATIONS)

PROGRAMME EDUCATIONAL OBJECTIVES:

PEO 1: To impart core theoretical as well as practical skills in software development, to build competencies, for creating real-world computer applications in diverse domains.

PEO 2: To imbibe lifelong learning in graduates and prepare them for successful careers in software and IT-enabled industry as well as in entrepreneurship, research and higher studies with all the ethics and professionalism.

PEO 3: To develop strong oral and written communication skills in graduates to effectively convey technical concepts and collaborate with team members, clients, and stakeholders.

PROGRAMME OUTCOMES (POs):

PO1: Apply Basic knowledge: Apply the knowledge of mathematics, science and computing fundamentals to provide solutions for complex computer applications.

PO2: Problem analysis: Identify, formulate, research literature, and analyze problems in applied computer science.

PO3: Design/development of applications: Design computer applications that meet the specified societal, health, safety, legal and cultural needs with appropriate consideration to ethics, environment and sustainability.

PO4: Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern IT tools including database management, networking AI & ML with an understanding of their limitations.

PO5: Communication: Communicate effective reports, design documentation and make effective presentations.

PO6: Project management and team work: Demonstrate knowledge and understanding of the management principles and apply them to one's own work, and also as a member and a leader in a team, to manage projects in multidisciplinary environments.

PO7: Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological changes in computer applications.

Program Specific outcomes (PSOs):

PSO1: To develop proficiency in software development methodologies and tools in order to design, implement, and test solutions across major core areas of AI&ML, Cyber security and Data analytics.

PSO2: To develop a versatile skill set inculcating soft skills, programming proficiency in full stack Web and Mobile application development.

CO-PO-PSO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2
CO1	**2**	**1**	**1**	**1**			**1**	**1**	**1**
	It focuses on understanding the fundamental principles of software engineering	It involves problem analysis to identify software process models	Understanding fundamentals to apply software process models	Understanding fundamentals to apply software process models			It slightly contributes to lifelong learning	Understanding fundamentals to apply software process models	It contributes to designing applications that involve principles of software engineering
CO2	**2**	**2**	**1**	**2**		**2**	**1**	**2**	**2**
	It aligns with developing SRS	It involves problem analysis to build the suitable software project plan	It slightly contributes to designing applications that involve principles of software engineering	It aligns with developing SRS		Understanding management principles and team world quality	It aligns with lifelong learning.	It aligns with developing SRS	It contributes to designing applications that involve principles of software engineering
CO3	**2**	**2**	**1**	**2**			**1**	**2**	**2**
	It aligns with designing diagrams	It involves problem analysis to formulate software metrics	It slightly contributes to designing applications that involve principles of software engineering	It uses modern tool for designing diagrams			It aligns with lifelong learning	It uses modern tool for designing diagrams	It contributes to designing applications that involve principles of software engineering
CO4	**2**	**2**	**2**	**2**			**1**	**2**	**2**
	It aligns with applying basic knowledge	It involves performance analysis of software system	It contributes to designing applications that involve software system testing	It uses modern tool for performance analysis of software system			It aligns with lifelong learning	It involves performance analysis of software system	It contributes to software system testing for web and mobile applications
Avg.	**2**	**2**	**1.75**	**1.75**		**2**	**1**	**1.75**	**1.75**

Detailed Syllabus
Lecture-wise Breakup

Subject Code	23B65CA225	**Semester: Odd**	**Semester: 1st Session: 2024-2025 Month: Jan- June 2025**
Subject Name	Full stack Development-I (Core Java) Lab		
Credits	**0-0-1**	**Contact Hours**	**2**

Faculty (Names)	**Coordinator**	Ms. Neha (62)
	Teacher(s) (Alphabetically)	Ms. Neha, Ms. Preeti Mittal, Dr. Shelendra Pal, Dr. Shobhit Tyagi, Dr. Tanvi Gautam,

COURSE OUTCOMES		**COGNITIVE LEVELS**
CBAC255.1	Implement basic Java programs using Java constructs – loops, switch-case, arrays & strings.	Apply (level 3)
CBAC255.2	Implement all basic concepts of oops using java programming	Apply (level 3)
CBAC255.3	Examine various methods to prevent program crashing and accordingly generate meaningful messages for users	Analyze (level 4)
CBAC255.4	Illustrate the flow of data through input/output operations and database connectivity.	Evaluate (level 5)
CBAC255.5	Develop an application based on Java programming constructs	Create (level 6)

Module No.	**Subtitle of the Module**	**Topics in the Module**	**No. of Labs for the module**
1	Fundamentals of programming in Java	Data Types, Variables, Constants, Input-Output classes, Command Line Arguments, Operators and type of operators, Control Flow	1
2	Classes and Objects in Java	Classes and Objects, Class and Instance Variables Constructors, Instance Methods, this keyword, finalize, Class Methods, Passing and returning objects, enum Types, Method Overloading, Arrays and Strings	1
3	Inheritance	Superclass and Subclass, protected Members this and super keywords, Method Overriding final variables, methods and classes	1
4	Packages and Interfaces	Packages, Polymorphic behavior, Abstract Classes	1

		Creating and using interfaces	
5	Enumerations, Autoboxing , Generics and Utility classes	Enumeration type, wrappers, Autoboxing, Generic Classes, String handling, String Tokenizer, Date, Calendar	2
6	Exception Handling	Checked and Unchecked Exceptions, Catching an Exception, finally Block, throw and throws clause Chained Exceptions, try-with-Resources: Automatic Resource Deallocation	1
7	Collections Framework	Collections Interfaces, classes Comparators, Legacy classes and Interfaces	2
8	Multithreading (Concurrency)	Creating threads using Thread class and Runnable interface, Thread priorities and Synchronization Inter thread Communication, Creating and Executing Threads with the Executor Framework	1
9	File Handling and JDBC in Java	Files: Streams- Byte streams, Character streams, Text input/output, Binary input/output, File management using File class. Connecting to Database:- Introduction to Databases and SQL Basics, Setting up JDBC (Driver, Connection),Performing CRUD Operations (Create, Read, Update, Delete), Prepared Statements and Callable Statements	2
10	Lambdas and Java Streams API	Introduction to Lambda Expressions, Introduction to Streams: Creating Streams from collections (Lists, Sets, Arrays, etc.), Intermediate operation (map(), filter(),sorted(), distinct(), limit() and skip()) Terminal Operations(Collect(), forEach(), reduce() count(), min(), max()), Parallel stream	2
		Total number of Labs	14

Evaluation Criteria

Components	**Maximum Marks**
Eval 1	15
Eval 2	15
Lab Test 1	20
Lab Test 2	20
PBL	15 (Students will submit the mini project in a group of 3-4 members)
Attendance	15
Total	**100**

Project based learning: Create a Java application in groups of maximum 4 students each, to illustrate the concepts covered in lab.

Recommended Reading material: Author(s), Title, Edition, Publisher, Year of Publication etc. (Text books, Reference Books, Journals, Reports, Websites etc. in the IEEE format)	
	Text Books
1	Herbert Shildt: *Java: The Complete Reference*, 12th Edition, McGraw-Hill, 2021.
2	Y. Daniel Liang: *Intro to Java Programming, Comprehensive Version*, 10th Edition, Pearson, 2018.
Reference Books	
1	Kathy Sierra, Bert Bares & Trisha Gee: *Head First Java,* 3rd Edition, O'Reilly Media, 2022.
2	Paul Deitel and Harvey Deitel: *Java: How to Program, Late Objects*, 11th Edition, Pearson, 2021.

CO-PO and CO-PSO Mapping:

COs (NBA Code)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2
CBAC255.1	**2**	**3**	**2**	**2**	**2**	**1**	**1**	**2**	**3**
CBAC255.2	**2**	**3**	**2**	**2**	**2**	**1**	**1**	**2**	**3**
CBAC255.3	**2**	**3**	**2**	**2**	**2**	**2**	**1**	**2**	**3**
CBAC255.4	**2**	**3**	**3**	**3**	**2**	**2**	**2**	**2**	**3**
CBAC255.5	**3**	**3**	**3**	**3**	**3**	**3**	**3**	**2**	**3**
Avg	**2.20**	**3.00**	**2.40**	**2.40**	**2.20**	**1.80**	**2.00**	**2.00**	**3.00**

CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PSO 1	PSO2
CBAC 255.1	2, Basic Java constructs contributes to foundational engineering knowledge in computer science and software	3, Java basics aids in problem decomposition and analysis, essential for effective	2, Java constructs supports designing solutions to engineering problems by enabling the developme	2, Mastery of Java basics aligns with modern engineering tool usage, essential for effective software development practices.	2, Clear communication in software development, facilitated by mastery of Java basics, is crucial for effective engineering activities.	1, Proficiency in Java basics supports effective project management and financial planning in	1, Java basics serve as a foundation for continuous learning, essential for adapting to	2, Java basics, including knowledge of arrays and strings, aids in understand suitable	3, Java basics enhances performance in programming competitions and technological challenge

	engineering.	engineering problem-solving.	nt of efficient algorithms and data structures.			software development projects.	technological advancements throughout an engineer's career.	data structures and algorithms for problem-solving.	s, demonstrating readiness for professional engagements.
CBAC 255.2	2, OOP concepts in Java contributes to foundational engineering knowledge in computer science and software engineering.	3, OOP principles aids in problem decomposition and analysis, essential for effective engineering problem-solving.	2, OOP concepts supports designing solutions to engineering problems by enabling the creation of well-structured, modular software systems.	2, OOP in Java aligns with modern engineering tool usage, essential for effective software development practices.	2, Clear communication in software development, facilitated by mastery of OOP, is crucial for effective engineering activities.	1, OOP supports effective project management and financial planning in software development projects.	1, Understanding OOP serves as a foundation for continuous learning, essential for adapting to technological advancements throughout an engineer's career.	2, OOP concepts, aids in understanding suitable solutions using appropriate data structures and algorithms.	3, OOP enhances performance in programming competitions and technological challenges, demonstrating readiness for professional engagements.
CBAC 255.3	2, Java programs with prevent program crashing contributes to foundational engineering knowledge in computer science and software engineering.	3, Java programs involving prevent program crashing aids in problem decomposition and analysis, essential for effective engineering problem-solving.	2, prevent program crashing supports designing solutions to engineering problems by enabling the development of robust, concurrent software systems.	2, prevent program crashing aligns with modern engineering tool usage, essential for effective software development practices.	2, Clear communication in software development, facilitated by understanding prevent program crashing is crucial for effective engineering activities.	2, prevent program crashing supports effective project management and financial planning in software development projects.	1, prevent program crashing serves as a foundation for continuous learning, essential for adapting to technological advancements throughout an engineer's career.	2, Java programs with prevent program crashing aids in understand suitable solutions using appropriate data structures and algorithms, especially in concurrent and error-prone scenarios.	3, prevent program crashing enhances performance in programming competitions and technological challenges, demonstrating readiness for professional engagements.

CBAC 255.4	2, Java flow of data contributes to foundational engineering knowledge in computer science and software engineering.	3, Java Collection Framework aids in problem analysis by enabling the selection of appropriate data structures for effective problem-solving.	3, Java flow of data supports designing solutions to engineering problems by facilitating the selection and implementation of efficient data structures.	3, Java flow of data aligns with modern engineering tool usage, essential for effective software development practices.	2, Clear communication in software development, facilitated by understanding the flow of data, is crucial for effective engineering activities.	2, Java flow of data supports effective project management and financial planning in software development projects by optimizing resource allocation and data management strategies.	2, Java flow of data serves as a foundation for continuous learning, essential for adapting to technological advancements throughout an engineer's career.	2, Java flow of data aids in identifying suitable data structures and algorithms for problem-solving, contributing to effective solution development.	3, Java flow of data enhances performance in programming competitions and technological challenges, demonstrating readiness for professional engagements.
CBAC 255.5	3, Developing applications based on Java programming constructs demonstrates advanced engineering knowledge in computer science and software engineering.	3, Developing Java applications requires thorough problem analysis and decomposition, essential for effective engineering problem-solving.	3, Developing applications in Java involves designing and developing solutions to engineering problems, demonstrating competency in software engineering practices.	3, Developing Java applications aligns with modern engineering tool usage, essential for effective software development practices in various domains.	3, Clear communication in software development is essential for developing Java applications that meet stakeholder requirements and adhere to project specifications.	3, Developing Java applications involves effective project management and financial planning to ensure successful software development projects.	3, Developing Java applications requires continuous learning and adaptation to technological advancements, fostering a commitment to lifelong learning in engineering.	2, Developing Java applications involves selecting suitable data structures and algorithms to solve real-life problems, demonstrating proficiency in PSO1.	3, Developing Java applications enables students to excel in programming competitions and technological challenges, showcasing readiness for professional engagements.