Pimpri Chinchwad Education Trust's

PIMPRI CHINCHWAD COLLEGE OF ENGINEERING

SECTOR NO. 26, PRADHIKARAN, NIGDI, PUNE 411044

(An Autonomous Institute Approved by AICTE and Affiliated to SPPU, Pune)

Curriculum Structure and Syllabus
of
Final Year B. Tech. Computer Engineering
(Regulations 2023)



Effective from Academic Year 2026-27

Institute Vision

To be one of the top 100 Engineering Institutes of India in coming five years by offering exemplarily Ethical, Sustainable and Value Added Quality Education through a matching ecosystem for building successful careers.

Institute Mission

1. Serving the needs of the society at large through establishment of a state-of-art Engineering Institute.
2. Imparting right Attitude, Skills, Knowledge for self-sustenance through Quality Education.
3. Creating globally competent and Sensible engineers, researchers and entrepreneurs with an ability to think and act independently in demanding situations.

EOMS Policy

"Knowledge Brings Freedom"

"We at PCCOE are committed to offer exemplarily Ethical, Sustainable and Value Added Quality Education to satisfy the applicable requirements, needs and expectations of the Students and Stakeholders.

We shall strive for technical development of students by creating globally competent and sensible engineers, researchers, and entrepreneurs through Quality Education.

"Knowledge Brings Freedom"

We are committed for Institute's social responsibilities and managing Intellectual property.

We shall achieve this by establishing and strengthening state-of-the-art Engineering Institute through continual improvement in effective implementation of Educational Organizations Management Systems (EOMS)."

Pimpri Chinchwad Education Trust's

Pimpri Chinchwad College of Engineering

Course Approval Summary

A) Board of Study - Department of Computer Engineering

Sr. No.	Name of the Course	Course Code	Page number	Signature and stamp of BoS
1	Distributed System	BCE27PC01/ BCE28PC01	10	
2	Object Oriented Modelling and System Design	BCE27PC02/ BCE2/PC02	12	
3	PEC 4-DevOps	BCE27PE01/ BCE28PE03	15	
4	PEC 4-Agentic AI Design	BCE27PE02/ BCE28PE04	18	
5	Research Methodology	BCE27EL01/ BCE28EL03	20	
6	PEC 5 -MOOC	BCE28PE01	24	
	PEC 6- MOOC	BCE28PE02	24	
7	Internship	BCE28EL02/ BCE27EL03	27	
8	Project	BCE28EL01/ BCE27EL02	30	

Chairman
BoS, Computer Engineering
PCET's, Pimpri Chinchwad College of Engineering
Sector No. 26, Pradhikaran, Nigdi, Pune-

Approved by Academic Council:

Chairman
Academic Council
PCET's, Pimpri Chinchwad College of Engineering
Sector No. 26, Pradhikaran, Nigdi, Pune-44

Chairman, Academic Council

Pimpri Chinchwad College of Engineering

INDEX

Sr. No.	Content	Page No.
1	List of Abbreviations in Curriculum Structure	1
2	Course wise curriculum Distribution	3
3	Curriculum Structure – B.Tech. Semester VII	7
4	Curriculum Structure – B.Tech. Semester VIII	8
5.	Course Syllabus Semester VII	9
6.	Course Syllabus Semester VIII	23
7.	Vision and Mission of Computer Engineering Department	41

"Knowledge Brings Freedom"

LIST OF ABBREVIATIONS IN CURRICULUM STRUCTURE

Sr. No.	Abbreviation	Type of Course
1.	BSC	Basic Science Course
2.	ECC	Engineering Core/ Science Course
3.	HSMC	Humanities, Social Sciences and Management Course
4.	PCC	Programme / Professional Core Course
5.	PEC	Programme / Professional Elective Course
6.	OEC	Open Elective Course
7.	PROJ	Project
8.	INTR	Internship
9.	AC	Audit Course
10.	MC	Mandatory Course
11.	LS	Life Skill
12.	PFC	Proficiency Course
13.	MO	MOOC Course
14.	L	Lecture
15.	P	Practical
16.	T	Tutorial
17.	H	Hours
18.	CR	Credits
20.	FA	Formative Assessment
21.	SA	Summative Assessment
22.	TW	Term Work
23.	OR	Oral
24.	PR	Practical

CURRICULUM FRAMEWORK
(Regulations 2023)

LIST OF ABBREVIATIONS

Sr. No.	Abbreviation	Type of Course
1	BSC	Basic Science Course
2	ESC	Engineering Science Course
3	PCC	Programme Core Course
4	PEC	Programme Elective Course
5	MDM	Multidisciplinary Minor
6	OEC	Open Elective Course
7	VSEC	Vocational and Skill Enhancement Course
8	AEC	Ability Enhancement Course
9	EEM	Entrepreneurship/Economics/Management Course
10	IKS	Indian Knowledge System
11	VEC	Value Education Course
12	ELC	Experiential Learning Course
13	LLC	Liberal Learning Course

"Knowledge Brings Freedom"

COURSE WISE CREDIT DISTRIBUTION

Sr. No.	Type of Course	No. of Courses	Total Credits	
			No.	%
1	Basic Science Course (BSC)	8	14	9
2	Engineering Core Course (ECC)	5	12	8
3	Programme Core Course (PCC)	15	46	29
4	Programme Elective Course (PEC)	6	20	13
5	Multidisciplinary Minor (MDM)	6	14	9
6	Open Elective Course (OEC)	3	6	4
7	Vocational and Skill Enhancement Course (VSEC)	4	8	5
8	Ability Enhancement Course (AEC)	2	4	3
9	Entrepreneurship/ Economics/ Management Course (HSSM)	2	4	3
10	Indian Knowledge System (IKS)	1	2	1
11	Value Education Course (VEC)	2	4	3
12	Experiential Learning Courses	4	22	14
13	Liberal Learning Courses	2	4	3
	Total	**60**	**160**	**100**

"Knowledge Brings Freedom"

SEMESTER-WISE COURSE DISTRIBUTION

COURSE DISTRIBUTION: SEMESTER WISE										
SR NO	TYPE OF COURSE	NO. OF COURSES / SEMESTER								Total
		1	2	3	4	5	6	7	8	
1.	Basic Science Course (BSC)	4	4	-	-	-	-	-	-	8
2.	Engineering Core Course (ECC)	3	2	-	-	-	-	-	-	5
3.	Programme Core Course (PCC)	-	1	2	2	5	3	2	-	15
4.	Programme Elective Course (PEC)	-	-	-	-	1	2	1	2	6
5.	Multidisciplinary Minor (MDM)	-	-	1	1	2	1	1	-	6
6.	Open Elective Course (OEC)	-	-	2	1	-	-	-	-	3
7.	Vocational and Skill Enhancement Course (VSEC)	1	1	-	1	-	1	-	-	4
8.	Ability Enhancement Course (AEC)	1	-	-	1	-	-	-	-	2
9.	Entrepreneurship/ Economics/ Management Course (HSSM)	-	-	1	1	-	-	-	-	2
10.	Indian Knowledge System (IKS)	-	1	-	-	-	-	-	-	1
11.	Value Education Course (VEC)	-	-	1	1	-	-	-	-	2
12.	Experiential Learning Courses	-	-	1	-	-	-	1	2	4
13.	Liberal Learning Courses	1	1	-	-	-	-	-	-	2
Total		10	10	8	8	8	7	5	4	60

"Knowledge Brings Freedom"

SEMESTER-WISE CREDIT DISTRIBUTION

CREDIT DISTRIBUTION: SEMESTER WISE										
1 Lecture hour = 1 Credit, 2 Lab Hours = 1 Credit, 1 Tutorial Hour = 1 Credit										
SR NO	TYPE OF COURSE	CREDITS / SEMESTER								Total
		1	2	3	4	5	6	7	8	
1.	Basic Science Course (BSC)	7	7	-	-	-	-	-	-	14
2.	Engineering Core Course (ECC)	7	5	-	-	-	-	-	-	12
3.	Programme Core Course (PCC)	-	2	8	8	12	8	8	-	46
4.	Programme Elective Course (PEC)	-	-	-	-	4	8	4	4	20
5.	Multidisciplinary Minor (MDM)	-	-	2	2	4	2	4	-	14
6.	Open Elective Course (OEC)	-	-	4	2	-	-	-	-	6
7.	Vocational and Skill Enhancement Course (VSEC)	2	2	2	-	-	2	-	-	8
8.	Ability Enhancement Course (AEC)	2	-	-	2	-	-	-	-	4
9.	Entrepreneurship/ Economics/ Management Course (HSSM)	-	-	2	2	-	-	-	-	4
10.	Indian Knowledge System (IKS)	-	2	-	-	-	-	-	-	2
11.	Value Education Course (VEC)	-	-	2	2	-	-	-	-	4
12.	Experiential Learning Courses	-	-	-	2	-	-	4	16	22
13.	Liberal Learning Courses	2	2	-	-	-	-	-	-	4
Total		20	20	20	20	20	20	20	20	160

"Knowledge Brings Freedom"

Curriculum Structure

BTech Computer

"Knowledge Brings Freedom"

CURRICULUM STRUCTURE
(Scheme A)
Final Year B.Tech. (Computer Engineering) Semester – VII

Final Year Computer Engineering (Regulations 2023) (With effect from Academic Year 2026-27)																
Semester VII																
Course Code	Course Name	Credit Scheme				Teaching Scheme (Hours/Week)				Evaluation Scheme and Marks						
		L	P	T	Total	L	P	T	O*	FA		SA	TW	PR	OR	Total
										FA1	FA2					
	MDM Projcct	-	4	-	4	-	8	-	-	-	-	-	100	-	-	100
Theory cum Practical																
BCE27PC01	Distributed System	4			4	2	4	-	2	10	10	30	25	-	25	100
BCE27PC02	Object Oriented Modelling and System Design	4			4	2	4	-	2	10	10	30	50	-	-	100
BCE27PE01/ BCE27PE02	Program Elective 4	4			4	2	4	-	2	10	10	30	50	-	-	100
BCE27EL01	Research Methodology	4			4	2	4	-	2	10	10	30	50	-	-	100
Total		**20**		-	**20**	**8**	**24**	-	**8**	**40**	**40**	**120**	**275**	-	**25**	**500**

"Knowledge Brings Freedom"

L-Lecture, **P**-Practical, **T**-Tutorial, **FA**–Formative Assessment, **SA** – Summative Assessment, **TW**-Term Work, **OR**-Oral, **PR**-Practical, ***O****: *Others(Self Learning)*

Note: Students are required to ensure that the same course is not selected under multiple categories such as Core Courses, Programme Electives, Open Electives, or through the Minor Degree in Multidisciplinary Studies. Each course should be credited only once towards the degree requirements

List of Courses – Programme Elective Course 4

Course Code	Course Name	
BCE27PE01	DevOps	Choose any one
BCE27PE02	Agentic AI Design	

CURRICULUM STRUCTURE
(Scheme A)

Final Year B.Tech. (Computer Engineering) Semester – VIII

Final Year Computer Engineering (Regulations 2023) (With effect from Academic Year 2026-27)																
Semester VIII																
Course Code	Course Name	Credit Scheme				Teaching Scheme (Hours/Week)				Evaluation Scheme and Marks						
										FA		SA	TW	PR	OR	Total
		L	P	T	Total	L	P	T	O*	FA1	FA2					
BCE28PE01	PEC 5-MOOC	2	-	-	2	2	-	-	2	-	50	-	-	-	-	50
BCE28PE02	PEC 6-MOOC	2	-	-	2	2	-	-	2	-	50	-	-	-	-	50
BCE28EL01	Project	-	4	-	4	-	8	-	-	-	-	-	100	-	50	150
BCE28EL02	Internship	-	12	-	12	-	24	-	-	-	-	-	150	-	100	250
Total		**4**	**16**	**-**	**20**	**4**	**32**	**-**	**4**	**50**	**50**	**-**	**250**	**-**	**150**	**500**

L-Lecture, **P**-Practical, **T**-Tutorial, **FA**–Formative Assessment, **SA** – Summative Assessment, **TW**-Term Work, **OR**-Oral, **PR**-Practical, ***O****: *Others(Self Learning)*

Note: Students are required to ensure that the same course is not selected under multiple categories such as Core Courses, Programme Electives, Open Electives, or through the Minor Degree in Multidisciplinary Studies. Each course should be credited only once towards the degree requirements

CURRICULUM STRUCTURE
(Scheme B)
Final Year B.Tech. (Computer Engineering) Semester – VII

Final Year Computer Engineering (Regulations 2023) (With effect from Academic Year 2026-27)																
Semester VII																
Course Code	Course Name	Credit Scheme				Teaching Scheme (Hours/Week)				Evaluation Scheme and Marks						
		L	P	T	Total	L	P	T	O*	FA		SA	TW	PR	OR	Total
										FA1	FA2					
	MDM Project	-	4	-	4	-	8	-	-	-	-	-	100	-	-	100
BCE27EL02	Project	-	4	-	4	-	8	-	-	-	-	-	100	-	50	150
BCE27EL03	Internship	-	12	-	12	-	24	-	-	-	-	-	150	-	100	250
	Total	-	20	-	20	-	40	-	-	-	-	-	350	-	150	500

L-Lecture, **P**-Practical, **T**-Tutorial, **FA**–Formative Assessment, **SA** – Summative Assessment, **TW**-Term Work, **OR**-Oral, **PR**-Practical, ***O****: *Others(Self Learning)*

Note: Students are required to ensure that the same course is not selected under multiple categories such as Core Courses, Programme Electives, Open Electives, or through the Minor Degree in Multidisciplinary Studies. Each course should be credited only once towards the degree requirements

"Knowledge Brings Freedom"

CURRICULUM STRUCTURE (Scheme B)

Final Year B.Tech. (Computer Engineering) Semester – VIII

Final Year Computer Engineering (Regulations 2023) (With effect from Academic Year 2026-27)																
Semester VIII																
Course Code	Course Name	Credit Scheme				Teaching Scheme (Hours/Week)				Evaluation Scheme and Marks						
										FA		SA	TW	PR	OR	Total
		L	P	T	Total	L	P	T	O*	FA1	FA2					
BCE28PE01	PEC 5-MOOC	2	-	-	2	2	-	-	2	-	50	-	-	-	-	50
BCE28PE02	PEC 6-MOOC	2	-	-	2	2	-	-	2	-	50	-	-	-	-	50
BCE28PC01	Distributed System	4		-	4	2	4	-	2	10	10	30	25	-	25	100
BCE28PC02	Object Oriented Modelling and System Design	4		-	4	2	4	-	2	10	10	30	50	-	-	100
BCE28PE03 / BCE28PE04	Program Elective 4	4			4	2	4	-	2	10	10	30	50	-	-	100
BCE28EL03	Research Methodology	4			4	2	4	-	2	10	10	30	50	-	-	100
Total		**20**		**-**	**20**	**12**	**16**	**-**	**12**	**40**	**140**	**120**	**175**	**-**	**25**	**500**

L-Lecture, **P**-Practical, **T**-Tutorial, **FA**–Formative Assessment, **SA** – Summative Assessment, **TW**-Term Work, **OR**-Oral, **PR**-Practical, ***O****: Others(Self Learning)*

Note: Students are required to ensure that the same course is not selected under multiple categories such as Core Courses, Programme Electives, Open Electives, or through the Minor Degree in Multidisciplinary Studies. Each course should be credited only once towards the degree requirements

List of Courses – Programme Elective Course 4

Course Code	Course Name	
BCE28PE03	DevOps	Choose any one
BCE28PE04	Agentic AI Design	

Course Syllabus

BTech Semester - VII

"Knowledge Brings Freedom"

Program:			B. Tech. (Computer Engineering)						Semester: VII				
Course:			Distributed Systems						Code: BCE27PC01				
Credit			Teaching Scheme (Hrs./Week)				Evaluation Scheme and Marks						
							FA		SA	TW	OR	PR	Total
L	P	T	Lecture	Practical	Tutorial	O*	FA1	FA2					
4		-	2	4	-	2	10	10	30	25	25	-	100

Prior knowledge of Network and Security, IOT **is essential**

Course Objectives:

This course aims at enabling students:

1. Understand the fundamental concepts, design issues, and goals of distributed systems.
2. Explain communication models, synchronization, and resource sharing in distributed environments.
3. Apply concepts to design, implement, and evaluate distributed applications and services.
4. Explore emerging trends in distributed systems.

Course Outcomes:

After learning the course, the students will be able to:

1. Compare various types, architectures, and design issues of distributed systems.
2. Demonstrate communication mechanisms in distributed environments.
3. Apply algorithms for election, mutual exclusion, and clock synchronization to achieve coordination in distributed systems.
4. Analyze emerging distributed paradigms to design architectural solutions for small-scale distributed applications using modern tools.

Guidelines for Laboratory/Term Work Assessment:

1. Continuous assessment of laboratory work is done based on overall performance and Laboratory performance of students.
2. Each Laboratory assignment assessment should assign grade/marks based on parameters with appropriate weightage.
3. Suggested parameters for overall assessment as well as each Laboratory assignment assessment include- timely completion, performance, innovation, efficiency, punctuality, and neatness.

Guidelines for Laboratory Conduction:

1. 64-bit Linux (Ubuntu/Fedora) or Windows.
2. Programming Tools:Python, Java,Use of Anaconda or Jupyter Notebook is encouraged.
3. Frameworks / Platforms:gRPC, Docker, Kubernetes, Apache Hadoop, MongoDB.
4. Development Environment: VS Code / PyCharm .

Detailed Syllabus

Unit	Description	Duration (H)
I	**Introduction:** Definition , Goals, Types of Distributed Systems, Architectures, Design Issues, Middleware, Overview of Distributed Systems, distributed multimedia systems. **Self Study:** A model of distributed computations, The Role of Virtualization in Distributed Systems.	7
II	**Communication:** Fundamentals, Remote Procedure Call, Message-Oriented Communication, Stream-Oriented Communication, P2P Messaging, WebRTC. **Self Study:** Names, Identifiers, and Addresses,Fault tolerance.	8

III	**Synchronization** Clock Synchronization, Logical Clocks, Lamport's Algorithm, Global State, Vector Algorithm, Election Algorithms, Mutual Exclusion Algorithms,beacon protocol . **Self Study :** Lodha and Kshemkalyani's fair mutual exclusion algorithm , Knapp's classification of distributed deadlock detection algorithms.	8
IV	**Emerging Distributed Paradigms** Distributed Object-Based Systems, Distributed Web-Based Systems, Distributed file system,Serverless Architectures, Case Studies:Megaport , Cloudflare, AWS, Apache Hadoop, Kubernetes. **Self Study:** Blockchain and distributed ledger technologies , understanding trade-offs in the design of modern distributed databases.	7
	Total	30

Suggested Lab Assignments : (60 hours)

1. **Client–Server Program using Socket/gRPC**
   Implement a simple client-server communication model using sockets or gRPC to exchange messages between distributed nodes.

2. **Coordinator Election (Bully/Ring Algorithm)**
   Simulate process coordination by implementing Bully or Ring algorithm for dynamic leader (coordinator) selection in a distributed system.

3. **Deadlock Detection Algorithm**
   Implement a deadlock detection mechanism using a wait-for graph or similar approach to identify and resolve deadlocks among processes.

4. **Clock Synchronization (Lamport or Variant)**
   Simulate logical clock synchronization among distributed processes using Lamport's algorithm or vector clock approach.

5. **Word Count using Hadoop**
   Implement a MapReduce program in Hadoop to count word frequencies from text data stored in HDFS.

6. **Mini Project**
   a. Design and deploy a small distributed application using Docker or Kubernetes or equivalent distributed systems demonstrating scalability and inter-service communication.

"Knowledge Brings Freedom"

Text Books:

- Distributed Systems –Principles and Paradigms" , Andrew S. Tanenbaum & Maarten van Steen, PHI. ISBN-978-81-203-2215 ,2nd edition.
- Distributed Operating Systems Concepts and Design" , Pradeep K. Sinha, PHI Publication 2007 .

Reference Books:

- George Coulouris et al. — Distributed Systems: Concepts and Design, Pearson,The 5th edition .
- Ajay D. Kshemkalyani & Mukesh Singhal — Distributed Computing: Principles, Algorithms, and Systems, Cambridge University Press.

E-resources:

- **https://webrtc.org/**
- **https://kubernetes.io/docs/tutorials/kubernetes-basics/**
- https://docs.megaport.com/overview/
- **https://hadoop.apache.org/docs/r1.2.1/hdfs_design.html**
- https://developers.cloudflare.com/fundamentals/concepts/how-cloudflare-works/
- https://onlinecourses.nptel.ac.in/

Program:	B. Tech. (Computer Engineering)						Semester: VII				
Course:	Object Oriented Modelling and System Design						Code: BCE27PC02				
Credit	Teaching Scheme (Hrs./Week)				Evaluation Scheme and Marks						
					FA		SA	TW	OR	PR	Total
L P T	Lecture	Practical	Tutorial	O*	FA1	FA2					
4 -	2	4	-	2	10	10	30	50	-	-	100

Prior knowledge of Object Oriented Concept, Software Engineering, Programming **is essential**

Course Objectives:

This course aims at enabling students:

1. To learn advanced object-oriented principles (SOLID, GRASP) and UML for modeling software systems concepts.
2. To understand database schema design and persistence layers.
3. To learn scalable system and service design approaches.
4. To understand the integration of object, data, service, and architectural designs into deployable system solutions.

Course Outcomes:

After learning the course, the students will be able to:

1. Apply advanced object-oriented principles and UML to model complex systems.
2. Design database schemas and persistence layers aligned with object models.
3. Design scalable and efficient system architectures using appropriate design principles, patterns, and UI design approaches..
4. Integrate system components to develop complete and deployable software solutions.

Guidelines for Laboratory/Term Work Assessment:

- Continuous assessment of laboratory work is done based on overall performance and lab assignments performance of the students. Each lab assignment assessment will assign marks based on parameters with appropriate weightage.
- Weightage of term work is based on suggested parameters for lab assignment includes- understanding and participation, timely completion, punctuality and neatness, overall professional behavior, internal oral, etc.

Detailed Syllabus

Unit	Description	Duration (H)
I	**Introduction Object-Oriented Modelling** Object-Oriented Thinking and Architectural Thinking, Advanced OOP Principles: SOLID Principles, GRASP Principles, Object Responsibilities, Cohesion, Coupling, UML Modelling, CRC Cards and Responsibility Assignment, Introduction to Architecture Decision Records (ADR), UML Modelling Tools: draw.io, StarUML, Mermaid.js, **Self Study:** Domain-Driven Design, ADR documentation templates	7
II	**Database and Persistence Layer Design** Translating Class Models into Database Schema, ER Model vs Class Model Mapping, Normalization Techniques, Object-Relational Mapping (ORM), DAO and Repository Patterns, NoSQL Database Design Concepts, Distributed Database Fundamentals, Case Study: Design persistence layer for E-Commerce / Library System **Self Study**: CAP Theorem, Database indexing strategies	7
III	**System and Service Design**	8

	System Design Fundamentals Software Architecture Patterns: Layered Architecture, Client-Server, Event-Driven, Microservices, REST API Design Principles: Resource modeling, HTTP methods, API versioning, API, Gateway, Scalability and Performance: Vertical and Horizontal Scaling, Load Balancing, Caching Strategies, Database, Scalability, Introduction to CDN, UI Design Approaches: Traditional HTML-based UI, React-based UI, Introduction to HTMx. **Self Study:** SOA principles and applications, Real-world API case studies	
IV	**Architectural Integration and Deployable System Design** Design Patterns: Factory, Singleton, Adapter, Observer, Strategy, MVC, Architecture Quality Attributes: Performance, Scalability, Maintainability, Security, Deployment Design: Deployment Diagram, Containerization Concepts, Integration of: Object Model, Database Design, API Layer, UI Layer, Deployment Architecture, Use of AI Tools for: Assignment validation, Design review, Documentation assistance, Writing Architecture Decision Records (ADR) **Self Study:** End-to-end System Design (Model to Deployable Architecture)	8
	Total	30

Suggested Lab Assignments:

1. Identify Objects, Responsibilities, and Relationships using GRASP Principles for an Online Course Management System. (Object List and Responsibility Table, CRC (Class–Responsibility–Collaboration) Cards, Preliminary Class Diagram)
2. Develop Use Case, Class, and Sequence Diagrams applying SOLID Principles for Smart Library Management. (Use Case Diagram,Class Diagram (demonstrating SRP, OCP, DIP), Sequence Diagram for "Borrow Book")
3. Design and Normalize a Database Schema from Object Model for an E-Commerce Order Processing System. (ER Diagram, Normalized Relational Schema, Mapping Table (Class -> Table))
4. Implement Object–Relational Mapping (ORM) for a Hospital Management System using Pseudocode or ORM Tools.(ORM Mapping File / Pseudocode, UML Diagram showing Entity Relationships, Short Report on Mapping Logic)
5. Design Service Interactions and API Architecture for an Online Food Delivery System. (Sequence Diagram,Component Diagram, API Specification Document (endpoint names, methods, request/response schema))
6. Apply Design Patterns to Enhance Flexibility in a Banking Transaction System. (UML Diagrams for Factory and Observer Patterns, Class Diagram showing pattern integration, Pseudocode demonstrating pattern logic)
7. Design Layered / Microservice Architecture and Deployment Diagram for a Smart Healthcare System (Architecture Diagram (Layered or Service-Oriented), Deployment Diagram, Brief architecture justification document).
8. Develop an Integrated System Design Document Combining Object, Database, Service, and Architecture Layers. (Complete Design Document (UML + DB + API + Architecture) Presentation Slides)

Text Books:

1. Grady Booch, James Rumbaugh, Ivar Jacobson, The Unified Modeling Language User Guide, 2nd Edition, Pearson Education, 2005. ISBN: 978-0321267979
2. Mark Richards & Neal Ford , Fundamentals of Software Architecture, O'Reilly, First Edition, 2020

Reference Books:

1. Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, Design Patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley, 1994.ISBN: 978-0201633610
2. Martin Fowler, Patterns of Enterprise Application Architecture, Addison-Wesley, 2003. ISBN: 978-0321127426
3. Robert C. Martin, Clean Architecture: A Craftsman's Guide to Software Structure and Design, Pearson

Education, 2017. ISBN: 978-0134494166
4. Martin Kleppmann, Designing Data-Intensive Applications, O'Reilly, first edition, 2017

E-resources:

1. UML Reference: www.uml.org
2. Design Patterns: refactoring.guru, sourcemaking.com
3. System Design: GitHub - System Design Primer

"Knowledge Brings Freedom"

Program:	B. Tech. (Computer Engineering)						Semester: VII				
Course:	DevOps						Code: BCE27PE01				
Credit	Teaching Scheme (Hrs./Week)				Evaluation Scheme and Marks						
					FA		SA	TW	OR	PR	Total
L P T	Lecture	Practical	Tutorial	O*	FA1	FA2					
4 -	2	4	-	2	10	10	30	50	-	-	100

Prior knowledge of Linux Bash Scripting and Python Programming **is essential**

Course Objectives:

This course aims at enabling students:

1. To understand the latest DevOps tools and technologies like DevOps tools like Git, Docker, Kubernet.
2. To develop the skills for implementing a project using various Infrastructure as Code methodologies.
3. To understand the selection process of zero-downtime CD strategies (Blue/Green, Canary) to meet application SLOs within a Kubernetes architecture.
4. To understand the SRE system by integrating an Observability Stack (Metrics/Logs/Alerts) and developing policies for incident response.

Course Outcomes:

After learning the course, the students will be able to:

1. Analyze DevOps cultural models and collaboration strategies effectively with version control workflows.
2. Design production-grade, immutable infrastructure by applying declarative Infrastructure as Code (IaC) tools.
3. Apply end-to-end Continuous Delivery (CD) pipelines using Pipeline as Code methodology.
4. Apply a comprehensive Site Reliability Engineering (SRE) system.

Guidelines for Laboratory/Term Work Assessment:

1. Continuous assessment of laboratory work is done based on overall performance and Laboratory performance of students.
2. Each Laboratory assignment assessment should assign grade/marks based on parameters with appropriate weightage.
3. Suggested parameters for overall assessment as well as each Laboratory assignment assessment include- timely completion, performance, innovation, efficiency, punctuality, and neatness.

Guidelines for Laboratory Conduction:

- 64-bit Linux (Ubuntu/Fedora) or Windows.
- Programming Tools: Python, Java, C/C++, Use of Anaconda or Jupyter Notebook is encouraged.
- Frameworks / Platforms: Git, GitHub, Docker, Kubernetes, Jenkins, Ansible.
- Development Environment: VS Code / PyCharm

Detailed Syllabus

Unit	Description	Duration (H)
I	**Foundations and Version Control** Introduction to DevOps, its history, and business benefits. CAMS Model , Three Ways of DevOps (Flow, Feedback, Continual Learning). Integration with Agile, Scrum, and Kanban methodologies. Git: GitHub, Git Features , 3-Tree Architecture, Various commands of Git. GitHub Actions. **Self Study:** Git Hub Tutorial	7

II	**Infrastructure as Code (IaC) and Containerization** Infrastructure as Code (IaC) Principles, concepts, benefits, Imperative and Declarative approaches. Introduction to cloud platforms (AWS, Azure, or GCP), AWS Aurora and AWS RDS, Ansible, Containerization – Docker, managing multi-container applications with Docker Compose. Do-It-Yourself Docker Module, Introduction to Microservices architecture. Terraform for provisioning cloud resources. **Self Study:** Docker Tutorial	8
III	**Continuous Integration and Continuous Delivery (CI/CD) :** CI/CD Pipeline Fundamentals, Understanding the Build, Test, Deploy loop, Setting up a CI server with Jenkins and GitLab CI, Pipeline as Code, Automating code compilation and unit testing, DevSecOps. Introduction to Continuous Deployment, Deployment strategies. Container Orchestration: - Kubernetes (K8s), architecture. Introduction to Build Tool- Maven **Self Study:** Kubernetes, Maven Tutorials	8
IV	**Monitoring, Observability, and Site Reliability Engineering (SRE):** Monitoring and Observability (O11y), Three Pillars of Observability (Metrics, Logs, Traces). Setting up centralized logging using the ELK Stack (Elasticsearch, Logstash, Kibana) or similar. Metrics and Visualization: Prometheus, Grafana. Site Reliability Engineering (SRE) Principles - Service Level Objectives /Agreements/Indicators), Incident Management and Post-mortems. **Self Study:** Prometheus Tutorial	7
	Total	30

Suggested List of Assignments: -

1. Explore the different git commands to working with local repository, remote repository and log operation (add, commit, status, log, show, branch, checkout, merge, clone, pull, reset, revert, rebase)
2. Explore the different services of AWS. (EC2, S3, AWS Lambda, RDS, ELB,ECS) and create EC2 instance in AWS.
3. Discover Infrastructure as a Code using Terraform and write a terraform script to create virtual machine EC2.
4. Explore the Docker architecture, Docker commands and Dockerfile and containerize a simple Python Application in Docker.
5. Discover Infrastructure as a Code using Terraform and write a terraform script to create virtual machine EC2
6. Study Jenkins Architecture and Plugins with Jenkins Installation and Setup.
7. Create a Freestyle Jenkins job: Configure a job to print "Hello, Jenkins!" in the console output and view logs.
8. Integrate Jenkins with a Git repository. Configure Jenkins to pull code from GitHub.
9. Explore the concept of Container orchestration using Kubernetes with Kubernetes architecture and the key components of a Helm chart (Chart.yaml, values.yaml, templates/)Helm commands: install, upgrade, uninstall, and template.
10. Discover kubernetes objects (Pod, Deployment, Service,Namespace, ConfigMap, Secret, Persistent Volume) and kubernetes services (ClusterIP, NodePort, LoadBalancer, ExternalName).
11. Prepare a Case Study on Ansible and demonstrate its application.

Text Books:

- PierluigiRiti, —Pro DevOps with Google Cloud Platform|, Apress, ISBN: 978-1-4842-3896-7.-2018
- The Phoenix Project: A Novel About IT, DevOps, and Helping Your Business Win by Gene Kim, Kevin Behr, George Spafford, IT Revolution Press (Latest Edition) - 2018

"Knowledge Brings Freedom"

Reference Books:

- Viktor Farcic. —The DevOps 2.0 Toolkit: Automating the Continuous Deployment Pipeline with Containerized Microservices‖ - 2016
- Sanjeev Sharma and Bernie Coyne. —DevOps for Dummies‖. John Wiley & Sons, Inc., 2nd IBM Limited Edition, ISBN: 978-1-119-04705-6 - 2015
- The DevOps Handbook: How to Create World-Class Agility, Reliability, & Security in Technology Organizations by Gene Kim, Patrick Debois, John Willis, Jez Humble, Nicole Forsgren, IT Revolution Press (Latest Edition). - 2016

E-resources:

5. https://kubernetes.io/docs/tutorials/kubernetes-basics/
6. https://www.udemy.com/course/ci-cd-devops/
7. https://www.edureka.co/devops-certification-training
8. https://www.coursera.org/learn/intro-to-devops

"Knowledge Brings Freedom"

Program:	B. Tech. (Computer Engineering)						Semester: VII				
Course:	Agentic AI Design						Code: BCE27PE02				
Credit	Teaching Scheme (Hrs./Week)				Evaluation Scheme and Marks						
					FA						
L P T	Lecture	Practical	Tutorial	O*	FA1	FA 2	SA	TW	OR	PR	Total
4 -	2	4	-	2	10	10	30	50	-	-	100

Prior knowledge of Python programming and Basics of AI/ML **is essential**

Course Objectives:

Students will be able to

1. Understand fundamental concepts, terminology, and interaction mechanisms of agentic artificial intelligence systems.
2. Explore and implement frameworks and programming tools for designing single-agent autonomous AI systems.
3. Analyze multi-agent workflows and evaluate agent performance, adaptability, and ethical considerations.
4. Design and develop a complete domain-specific agentic AI model integrating perception, reasoning, and action modules.

Course Outcomes:

After learning the course, the students will be able to:

1. Apply Fundamentals of AI agents and their interaction with environments.
2. Implement AI-agent systems using open-source frameworks.
3. Evaluate agentic performance, adaptability, and ethical constraints.
4. Design a domain-specific agentic AI project

Guidelines for Laboratory/Term Work Assessment:

1. Continuous assessment will be based on students' overall performance and individual lab work.
2. Each assignment will be graded on defined parameters with suitable weightage.
3. Suggested parameters: timely completion, technical performance, innovation, efficiency, punctuality, and documentation quality.
4. Final evaluation will reflect consistency, competence, and contribution across all lab activities.

"Knowledge Brings Freedom"

Guidelines for Laboratory Conduction:

System Requirements: 64-bit Linux (Ubuntu/Fedora) or Windows 10/11.

Programming Tools: Python (preferred), Java, C/C++; use of Anaconda or Jupyter Notebook encouraged.

Frameworks/Platforms: LangChain, LangGraph, AutoGen, LlamaIndex, Git, GitHub, Docker, Kubernetes, Jenkins.

Development Environment: VS Code / PyCharm / Jupyter Notebook.

Detailed Syllabus

Unit	Description	Duration (H)
I	**Foundations of Agentic AI** Introduction to agentic AI: definitions, distinctions between traditional AI, generative AI and agentic (autonomous) systems. (Copilot, AutoGPT,Claude Artifacts,n8n). Large-language-models (LLMs) and their role in agentic systems: tokenization, embeddings, prompts, retrieval. ReAct and Chain-of-Thought frameworks **Self Study:** Prompt Engineering Techniques.	8

II	**Architecture & Frameworks for Agents** Agent architectures: single-agent vs multi-agent systems. Design patterns: Tool-use pattern, Reflection pattern, Planning, ReAct (Reasoning+Acting) pattern, Multi-agent orchestration. Framework overview: popular libraries and platforms: LangChain, LangGraph and vector DBs (FAISS, Chroma). **Self Study:** AutoGen, Crew	8
III	**Benchmarks, Agent Workflow Implementation & Multi-Agent Systems** Multi-agent benchmarks, and Google's A2A/A2P protocols for assessing planning, coordination, and self-correction. Inter-agent communication; task delegation, collaboration, and role assignment; collective intelligence; workflow case studies: research automation, smart city monitoring. **Self Study:** ToolBench, AgentBench, Multi Agent Simulation.	7
IV	**Evaluation & Emerging Trends** Metrics for evaluating agents (accuracy, adaptability, safety); alignment and control; transparency and explainability; ethical and societal concerns; emerging AI governance standards. Emerging trends: multimodal agents, multi-agent ecosystems. **Self Study:** Generative/Agentic hybrid systems.	7
	Total	30

Suggested Lab Assignments: (60 hrs)

1. Installation and setup of Python, Jupyter Notebook/VS Code, Ollama/OpenAI API, LangChain/LangGraph, and development environment for Agentic AI.
2. Explore the architecture of an AI agent by implementing a simple ReAct (Reasoning and Acting) agent for solving question-answering tasks.
3. Develop a single-agent AI assistant using an LLM with prompt engineering techniques for task automation.
4. Design an autonomous agent capable of using external tools such as calculators, web search, APIs, or databases to complete complex tasks.
5. Multi Agent Simulation - Develop a multi-agent collaboration system where planner, researcher, and executor agents cooperate to solve a given problem.
6. Creating your own AI Assistant, Similar to Microsoft Copilot

Text Books:

5. Russell, S. J., & Norvig, P. . Artificial Intelligence: A Modern Approach 4th ed.. Pearson.. 2020.
6. Lanham, M. AI Agents in Action.1st ed. Manning Publications, 2025.

Reference Books:

1. Gulli, A. Agentic Design Patterns: A hands-on guide to building intelligent systems (1st ed.). Springer Cham. https://doi.org/10.1007/978-3-032-01402-3, 2025.
2. Bornet, P., & Wirtz, J. . Agentic artificial intelligence: Harnessing AI agents to reinvent business, work and life. Irreplaceable Publishing. ISBN 979-8-9928336-0-7, 2025.
3. Arsanjani, A., & Bustos, J. P. Agentic architectural patterns for building multi-agent systems: Proven design patterns & practices for GenAI, agents, RAG, LLMOps & enterprise-scale AI systems.2025.

E-resources:

1. https://docs.langchain.com/oss/python/langgraph/workflows-agents
2. https://academy.langchain.com/collections
3. https://microsoft.github.io/ai-agents-for-beginners/01-intro-to-ai-agents/

Program:	B. Tech. (Computer Engineering)						Semester: VII				
Course:	Research Methodology						Code: BCE27EL01				
Credit	Teaching Scheme (Hrs./Week)				Evaluation Scheme and Marks						
					FA		SA	TW	OR	PR	Total
L P T	Lecture	Practical	Tutorial	O*	FA1	FA2					
4	2	4	-	2	10	10	30	50	-	-	100

Prior knowledge of Mathematical Foundations, Programming **is essential**

Course Objectives:

This course aims at enabling students:

1. To learn fundamental concepts, types of research, and the overall research process.
2. To understand review of existing literature and analyze secondary data sources effectively.
3. To understand ethical concerns involved in research.

Course Outcomes:

After learning the course, the students will be able to:

1. Explore research and describe the research process and research methods.
2. Model and visualize the processes and requirements for conducting successful literature review and write a review paper by applying academic integrity and ethics
3. Investigate and apply the basic aspects of different statistical techniques to a research problem identified.
4. Analyze different optimization techniques and apply to a research problem identified

Guidelines for Laboratory/Term Work Assessment:

Faculty should form groups of students and assign tasks in groups. Students should use the central library facilities to access the e-journals for the Lab assignments. Each group should select a domain in which they should apply Research Methodology techniques which they learn progressively in each Unit. Since the course is TCP based, faculty can correlate the teaching with practical.

For TW assessment FA1 & FA2 activities will be conducted in the form of reviews and presentations, minimum 3 reviews will be conducted. Rubric will be framed for each review and during final submission the average of all three reviews will be considered. Outcome of the final submission will be a review paper for the international conference as well as a sample research proposal for the funding agencies.

Detailed Syllabus

Unit	Description	Duration (H)
I	**Introduction:** **Evolution of Research Methodology:** Meaning, nature, scope, and significance of research; Research paradigm; The purpose and outcomes of Research; Objectives of research; Motivation for research; Postulates underlying scientific investigations; Types of research; Research process and work flow. **Engineering Research-**Why? Research Questions, Engineering Ethics, conclusive proof-what constitutes, A research project-Why take on? **Self Study:** Code of Ethics, IEEE Code of Ethics, ACM Software Engineering Code of Ethics	8

II	**Literature Search and Review, Developing Research Plan:** Archival Literature. Why should engineers be ethical? Types of publications- Journal papers, conference papers, books, standards, patents, theses, trade magazine, newspaper article, infomercials, advertisement, Wikipedia & websites. Measures of research impact, Literature review, publication cost. **Case Study:** Engineering dictionary, Shodhganga, The Library of Congress, Research gate, Google Scholar, Bibliometrics, Citations, Impact Factor, h-index, I-index, plagiarism, copyright infringement. **Developing Research Plan:** Research Proposals, Finding a suitable research questions, The elements of research proposals-title, details, budget, Design for outcomes-1D data, 2D data, 3D data, N-D data, The research tools- Experimental measurements, numerical modeling, theoretical derivations & Calculations, curve matching. **Self Study:** Various Research grants and funding resources	8
III	**Statistical Analysis:** Introduction, Sources of error and uncertainty, One-Dimensional Statistics: combining errors and uncertainties, t-test, ANOVA statistics, example, Two-Dimensional Statistics: example, Multi-Dimensional Statistics: partial correlation coefficients, example, Null hypothesis testing. **Self Study:** GNU PSPP Tool, SOFA, NOST-Dataplot	7
IV	**Optimization Techniques:** Introduction, Two-parameter optimization methods: sequential uniform sampling, Monte Carlo optimization, Simplex Optimization method, Gradient Optimization method, Multi-parameter optimization methods, The cost function. **Self Study:** Google Optimization Tools, OpenMDAO	7
	Total	30

Suggested Lab Assignments: (60Hrs)

1. **Literature Review:** Students are often required to conduct a systematic literature review on a specific, current topic in computer engineering (e.g., AI, cyber security, big data, human-computer interaction, deep learning). The assignment involves summarizing published research papers, identifying research gaps, and synthesizing existing knowledge in a well-structured report.
2. **Experimental Design and Execution:** Students might design and run a small-scale experiment or simulation to test a hypothesis related to a computer system or algorithm. This includes data collection, analysis (often using statistical methods), and interpretation of results.
3. **Data Analysis Exercise:** Using a given dataset (from an experiment, survey, or publicly available repository), students perform data análysis to draw conclusions, often involving programming to process and analyze the data
4. **Technical Report Writing/Paper Summary:** Assignments may involve summarizing a complex technical paper or writing a section of a research report (e.g., a methodology section or results presentation) to develop technical writing and communication skills. By using a standard IEEE template, write a paper having paper title and keywords, writing an abstract, introduction, proposed approach, result analysis, conclusion and future scope.
5. **Research Proposal Writing:** A key assignment is developing a formal research proposal. This includes:
    a. **Problem Identification:** Clearly defining a specific research problem or question.
    b. **Objectives and Hypotheses:** Stating the research objectives and formulating testable hypotheses.
    c. **Methodology Design:** Describing the chosen research methods (experimental, simulation, case study, survey, etc.), data collection methods, and data analysis plan.
    d. **Significance:** Explaining the potential impact and contribution of the proposed research.

6. **Research Presentation:** : Prepare a presentation by using standard terms, Standard research methods and experimental techniques. , Paper presentation and review, Conference presentations, Poster presentations.
7. **Ethics and Plagiarism Module:** Assignments that specifically cover ethical considerations in research, proper citation, and avoiding plagiarism.

Text Books:

1. C. R. Kothari, "Research Methodology: Methods and Techniques" fourth Edition, New Age International Publishers, New Delhi, 2019
2. David V Thiel, "Research Methods- for Engineers", Cambridge University Press, ISBN:978-1-107-61019-4

Reference Books:

1. Caroline Whitbeck, "Ethics in Engineering Practice and Research", 2nd Ed., Cambridge University Press; ISBN :978-1-107-66847-8
2. Gordana DODIG-CRNKOVIC, "Scientific Methods in Computer Science", Department of Computer Science Malardalen University, Vasteas, Sweden; ISBN: 91-26-97860-1

E-resources:

ProQuest, OPAC, IEEE, Elsevier's Science Direct, E-Library, DELNET, NDLI, N- LIST

"Knowledge Brings Freedom"

Course Syllabus

BTech Semester - VIII

"Knowledge Brings Freedom"

Program:			B. Tech. (Computer Engineering)						Semester: VIII	
Course:			PEC 5 (MOOC) / PEC 6 (MOOC)						Code: BCE28PE01-02	
Credit			Teaching Scheme (Hrs./Week)				Evaluation Scheme and Marks			
			Lecture	Practical	Tutorial	O*	FA		SA	Total
L	P	T					FA1	FA2		
2	-	-	2	-	-	2	-	50	-	50*

Prior knowledge of as mentioned in MOOC course **is essential**

Course Objectives:

This course aims at enabling students:

1. To develop self-directed learning abilities
2. To familiarize students with modern tools, technologies, and emerging techniques
3. To enhance analytical and problem-solving skills
4. To enable the application of acquired technical knowledge in multidisciplinary environments.

Course Outcomes:

After learning the course, the students will be able to:

1. Inculcate Self learning.
2. Explore modern tools and techniques.
3. Analyze case studies and real-time problems
4. Apply the technical knowledge in multi-disciplinary projects.

Guidelines for Students:

1. Students should select MOOC from the NPTEL online platform or any other platform with prior approval from Project Guide and MOOC coordinator.
2. Selected MOOC courses should be of minimum of 30 hrs or Eight Weeks contact hours
3. Individual students should check that the selected course is not similar to the courses studied in earlier semesters.
4. Students must take prior approval from Mentor(Project Guide) and MOOC coordinator through the prescribed form given in Annexure-2 with proof of course registration done in NPTEL along with Annexure-2.
5. Students must complete all assessments related to the selected course as prescribed by the course in NPTEL.
6. After the completion of the course, students shall be required to submit the photocopy of the completion certificate and hardcopy of MOOC report.
7. MOOC courses should be completed in Semester VII or Semester VIII.
8. Evaluation will be based on Assignments and Certification (70% Assignments and 30% Certification).

E-resources:

1. https://onlinecourses.nptel.ac.in

Annexure 2: MOOC Course Approval Form

Pimpri Chinchwad Education Trust's

Pimpri Chinchwad College of Engineering

Annexure 2: MOOC Course Approval Form

Department: Academic Year: 20 -20 Date:

1. Student Name:		PRN No:	Year &Div :
2. Overall CGPA:		Phone :	
3. Student email address:			
4. Semester in which NPTEL course opted:			

NPTEL Course Details :

Sr. No.	Name of Course	Course Duration	Offering Institute	Substituting Course

The NPTEL courses opted by the student have been checked and found **OK/ Not OK**.

---------- ----------

Mentor MOOC Coordinator

Undertaking by the student:

I, _____, ROLL NO._____ will abide by the guidelines of the department for undertaking NPTEL courses and report the marks/grade/credits obtained would be duly communicated to my Mentor and MOOC Co-ordinator. The approval of the Mentor and MOOC Co-ordinatorwould be binding on me.

Signature of the student with Date:

Attachment : Proof of Course Registration

Sr. No.	Rubrics	Marks
1	Review-I a.Average scores of assignments(1-6 weeks)- 5 marks b.Presentation of the topics covered (1-6 weeks)-10 marks	15
2	Review-II a.Average scores of assignments(7-12 weeks)- 5 marks b.Presentation of the topics covered (7-12 weeks)-10 marks	15
3	MOOCs report Submission a.Plagiarism of report must be below acceptable level of standard-2 marks b.Content of the report -3 marks	5
	Total (A)	35
4	Final Examination Certificate a. Appeared for Exam - 2 marks b. Successful completion of the course with certificate: • Successfully Completed -2 marks • Elite -5 marks • Elite + Silver - 10 marks • Elite + Gold -12 marks • Topper Certificate (Top 5% / Topper) -15 marks	15
	Total(B)	15
	Total (A+B)	50

"Knowledge Brings Freedom"

Program:			B. Tech. (Computer Engineering)					Semester: VIII	
Course:			Internship					Code: BCE28EL02	
Credit			Teaching Scheme (Hrs./Week)			Evaluation Scheme and Marks			
L	P	T	Lecture	Practical	Tutorial	TW	OR	PR	Total
-	12	-	-	24	-	150	100	-	250

Prior knowledge of Programming Languages, Domain related subjects, Engineering Mathematics **is essential.**

Course Objectives:

1. To gain progressive exposure to industry/research environments and understand real-world professional practices.
2. To apply theoretical and conceptual knowledge from computer engineering courses to practical and interdisciplinary problems.
3. To develop technical competency in software engineering, system development, data analysis, or research projects.
4. To strengthen professional communication, documentation, and presentation skills.
5. To nurture teamwork, leadership, and collaborative skills in diverse and multidisciplinary environments.
6. To instill professional ethics, responsibility, and an understanding of socio-economic and environmental impacts of engineering work.

Course Outcomes:

After learning the course, the students will be able to:

1. Demonstrate understanding of industrial/research organization structure, workflows, and professional practices.
2. Apply computer engineering principles and tools to analyze and solve real-life technical problems.
3. Design and develop software/hardware systems or research prototypes meeting performance, scalability, and reliability standards.
4. Communicate effectively through reports, technical documentation, and professional presentations.
5. Work effectively in teams and demonstrate leadership, project management, and interpersonal skills.
6. Exhibit professional ethics, responsibility, and awareness of societal, environmental, and sustainability issues in engineering practice.

Internship program Spread from Semester III to Semester VIII and credits will be earned in semester VIII. Student can complete internship in following domain with their interest

- Software/Application Developement
- Research Internship
- Management (Domain)Skills

Internship Guidelines:

1. The internship must be at a recognized industry, government lab, R&D institute, academic institution, NGO, or startup.
2. Projects should relate to the student's discipline—software development, hardware, data analysis, computational projects, technical documentation, quality assurance, innovation, etc..

3. Students may also opt for research projects under faculty guidance within the institute.

Activities:

1. Active participation in assigned work, technical projects, coding, simulation, documentation, presentations.
2. Exposure to multidisciplinary teams, industry tools, project planning, and professional reporting.

Evaluation Criteria:

1. Submission of a comprehensive internship report (including objectives, methodology, results, learning outcomes, reflection, and technical content).
2. Maintenance of daily/weekly work diary signed by external and internal supervisors.
3. Seminar or oral presentation before a department evaluation committee.
4. Evaluation based on industry mentor's feedback and viva voce.
5. Grades reflected in the semester mark sheet.

Steps for Completion:

3. Internship application through formal process (department approval, bonafide certificate).
4. Internship and project execution at host organization.
5. Submission of signed and stamped report and work diary to department.
6. Oral presentation and viva.
7. Grading by committee (Pass/Fail/Letter grade).

Tentative Internship Credit Distribution:

Total Credits: 12

Duration: Spread from Semester III to Semester VIII

Objective: Progressive experiential learning through industry exposure, project work, research, and innovation.

Semester	Credits	Internship Component	Duration	Description	Deliverables
IV (2nd Year)	2	Mini Project / Industrial Exposure/Summer school	4 weeks (part-time/summer)	Short industry visit or mini-project assigned by faculty to introduce real-world engineering problems.	Mini project report, presentation to faculty
V (3rd Year)	2	Industrial Internship Phase I/ Winter School	6 weeks (winter break)	Internship at software companies/startups, R&D labs; tasks include coding, testing, and documentation.	Internship report, supervisor feedback
VI (3rd Year)	2	Industrial Internship Phase II	6 weeks (summer break)	Advanced internship focusing on software/system development or research under industry/faculty partnership.	Detailed project report, presentation, viva

VII (4th Year)	3	Research and Innovation Internship	4 weeks (part-time/semester)	In-house research or innovation projects; engagement with departmental labs or allied industries.	Research write-up, seminars
VIII (4th Year)	3	Major Project and Final Internship	8–10 weeks (semester-long)	Capstone project or industrial internship with deep engagement, leading to significant deliverables.	Final project report, publications (if any), seminar, viva

"Knowledge Brings Freedom"

Program:			B. Tech. (Computer Engineering)					Semester: VIII	
Course:			Project					Code: BCE28EL01	
Credit			Teaching Scheme (Hrs./Week)			Evaluation Scheme and Marks			
L	P	T	Lecture	Practical	Tutorial	TW	OR	PR	Total
-	4	-	-	8	-	100	50	-	150

Prior knowledge of

Software Engineering, Domain related subjects, Engineering Mathematics

is essential.

Course Objectives:

1. To understand the Project Development Process.
2. To develop problem solving ability by following Software Development Life Cycle meticulously.
3. To review literature for project work from appropriate sources such as books, manuals, research journals and from other sources, and in turn increase analytical skills.
4. To design and implement real world applications using available platforms.
5. To test and validate the work undertaken.
6. To evaluate the work undertaken.
7. To work in a team with individual contribution for the project development
8. To prepare good quality technical reports based on the selected project statement and present them effectively.
9. To develop solutions that ensure ethical use of technology and sustainability in project execution

Course Outcomes:

After learning the course, the students will be able to:

1. Analyze existing literature on various techniques and applications to identify research gaps and propose feasible solutions.
2. Design a real-world application considering actual requirements integrating ethical and sustainable practices.
3. Develop an application considering actual requirements and social/environmental/ethical/legal issues
4. Evaluate the results of the developed solution.
5. Work effectively in a team by contributing individually to project development.
6. Use report writing and originality verification tools to prepare and present quality technical reports.

Guidelines:

The intention of Project work is to conceive an idea and to implement it systematically by using knowledge derived during the course of education mainly to innovate or facilitate. A group of Under Graduate students at Final Year will undertake project work. Work involves study of feasibility of the project, planning of project, studying existing systems, tools available to implement the project and state of art software testing procedures and technology with use of case tools, design is to be implemented into a working model (software or hardware or both) with necessary software interface as an executable package. Projects may be in-house research, sponsored or multidisciplinary. Projects

can be carried out inside or outside the institute, in any relevant industry/organization or research institution or labs/organization.

1. Project Team

1. 3 - 4 students can form a team within the same or different discipline and their area of interest is to be registered with the project Coordinator.
2. It is necessary to explore the domain of interest / research/ thrust area/ society needs.
3. Students shall identify the area or topics in recent trends and developments in consultation with institute guide or industry or any research organization.
4. Each student of the team has to work collaboratively and contribute significantly for the project development.
5. Students shall meet their assigned project guide regularly (at least once in a week) and report the progress of the project work.
6. Students shall maintain the record of all the meetings, remarks given by guide / reviewers and progress of the work in the project diary. The project diary must be presented during each review presentation to the reviewers.
7. For final examination, students shall complete the project report in all aspects including formatting. Each Student shall prepare the report duly signed by project guide, Head of the department, Director and the external examiner (If applicable). Students should prepare three copies of the project report.
8. Students shall submit all the data related to project work in soft copy to their guides which shall include project report, Presentation, paper etc.
9. Students shall write a research article/paper, funding proposal, patent and copyright on their respective project work.
10. Students shall ensure the ethical use of technology and sustainability in project execution.
11. Students are encouraged to use modern tools that aid project management, collaboration, and documentation (e.g., GitHub, JIRA, Overleaf, Mendeley).

2. Project Activities

Students are expected to perform the following activities–

1. Review of Recent Literature and Gap Identification
2. Requirement Analysis and Feasibility Study
3. Defining the Problem Statement and Objectives
4. Identifying the Project Implementation Requirements
5. Formulation of Methodology and Mathematical Modeling
6. Project Implementation
7. Testing and Deployment
8. Observations & Results
9. Results Analysis and Validation
10. Conclusions
11. Research Paper Publication/IPR Filing if any/Participation in events
12. Report Writing

3. Project guide

- Each project activity must be supervised by a faculty member of the concerned department. This faculty member is termed as a Project Guide.
- In case of an interdisciplinary project, there can be two project guides; one from the parent department and second from the other department.
- Project guide shall help students to finalize (or identify) the project statement and suggest the objectives / methodology through brainstorming. The project team is required to regularly appraise the project guide about the progress and seek his/her guidance.

- Guide will motivate students to select interdisciplinary project / project facilitated and coordinated by Indian Knowledge System (IKS).
- Project guides must monitor the weekly progress being carried out by the project groups.
- In case of industry sponsored projects, guides are expected to visit the industry with students.
- The project guide shall ensure the completion of all the project related activities as per the requirement of review.
- Project guide shall motivate and facilitate the students to write patent, copyright, research funding proposal and paper publications for the overall development of the student.

Project Guide Allocation

- Considering registered teams area of interest/domain and expertise of guide, the project coordinator in consultation with a panel of experts allots Project guides.
- Guide should be allotted from the same program.
- In case of an interdisciplinary project, along with the guide from same program, guide or co-guide should be allotted from the other program.
- In case of a sponsored project (with reputed industry or any research organization) external guide should be from sponsored company/Industry along with the internal guide from the program.

4. Project Synopsis

Teams in consultation with guide will prepare project synopsis. The group should submit the synopsis in the following form.

1. Title of Project
2. Names of Students
3. Name of Guide
4. Background
5. Literature Review
6. Problem Definition
7. Objectives
8. Brief Methodology
9. Hardware/Software Requirements
10. References

The synopsis shall be signed by each student in the group, approved by the guide (along with external guide in case of sponsored projects) and endorsed by the Head of the Department.

5. Project Monitoring and Reviews

1. The project coordinator with the Head of the department shall constitute a review committee comprising of domain experts and senior faculty members.
2. The review committee will approve the project group and title. Discussion/ presentation may be arranged covering topics listed in the synopsis.
3. The project guide and review committee will evaluate the timely progress of the projects.
4. Students with group members are expected to appear for minimum two reviews as per the project calendar.
5. It is mandatory for students to remain present for all the reviews and examinations well before scheduled time.
6. Assessment criteria for each review in the form of rubrics should be notified to the students in advance by the project coordinator. Students have to read these carefully and accordingly to be prepared for reviews.

7. During reviews, students are required to demonstrate the progress done after the last review.
8. The suggestions or corrections given by the review panel committee should be recorded in the project dairy, incorporated and demonstrated in the consecutive reviews.
9. Final term work will be calculated based on the performance in reviews.

6. Project Report

- Report should be prepared using report writing tools such as MS Word, Latex as per the template provided by the department and verified using originality check softwares.
- In case of a Sponsored project, students must submit completion certificate with signature of external guide from the sponsored company.
- In case of an Interdisciplinary project, students must submit completion certificate with signature of both the guides.
- Project Report should begin with cover pages (Front Page, Certificate, and Certificate from industry if industry sponsored project, Acknowledgement, Abstract, Table of Contents, List of Figures, and List of tables).
- Then project report should be presented in a number of chapters, starting with Introduction and ending with Summary and Conclusions.
- Each of the other chapters will have a precise title reflecting the contents of the chapter.
- A chapter can be subdivided into sections, subsections and sub subsection so as to present the content discretely and with due emphasis.

7. Plagiarism

A student has to ensure that the Synopsis, Project Report and Research Publications are checked for plagiarism by using plagiarism software such as Grammerly, iThenticate / Turnitin/good rating software. The maximum similarity allowed is 10%. The plagiarism verification and AI writing reports must be attached in the project report. AI writing percentage should be 0%.

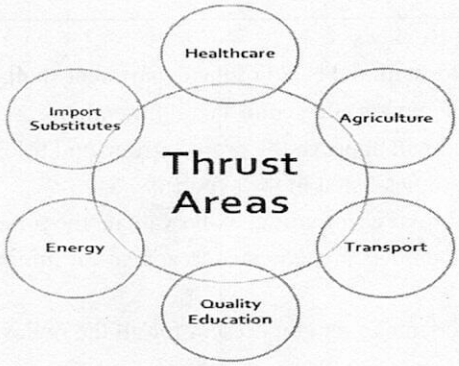
8. Research Outcomes

Based on the project results and conclusions, students are recommended to generate the research outcomes in terms of Research Publication, Patents, and Copyrights. This has to be done in consultation with project guides. Guides will decide the appropriateness of the results and converting those into research outcomes.

9. Thrust Areas and SDGs for Research

Students shall select a problem statement from the given thrust areas that effectively contributes to achieving the outlined Sustainable Development Goals (SDGs).

Thrust Areas:



Sustainable Development Goals:

List of the 17 Sustainable Development Goals (SDGs) adopted by the United Nations in the 2030 Agenda for Sustainable Development:

1. No Poverty – End poverty in all its forms everywhere.
2. Zero Hunger – End hunger, achieve food security and improved nutrition, and promote sustainable agriculture.
3. Good Health and Well-being – Ensure healthy lives and promote well-being for all at all ages.
4. Quality Education – Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.
5. Gender Equality – Achieve gender equality and empower all women and girls.
6. Clean Water and Sanitation – Ensure availability and sustainable management of water and sanitation for all.
7. Affordable and Clean Energy – Ensure access to affordable, reliable, sustainable, and modern energy for all.
8. Decent Work and Economic Growth – Promote sustained, inclusive, and sustainable economic growth, full and productive employment, and decent work for all.
9. Industry, Innovation, and Infrastructure – Build resilient infrastructure, promote inclusive and sustainable industrialization, and foster innovation.
10. Reduced Inequalities – Reduce inequality within and among countries.
11. Sustainable Cities and Communities – Make cities and human settlements inclusive, safe, resilient, and sustainable.
12. Responsible Consumption and Production – Ensure sustainable consumption and production patterns.
13. Climate Action – Take urgent action to combat climate change and its impacts.
14. Life Below Water – Conserve and sustainably use the oceans, seas, and marine resources for sustainable development.
15. Life on Land – Protect, restores, and promotes sustainable use of terrestrial ecosystems, manage forests sustainably, combat desertification, and halt biodiversity loss.
16. Peace, Justice, and Strong Institutions – Promote peaceful and inclusive societies, provide access to justice for all, and build effective, accountable institutions.
17. Partnerships for the Goals – Strengthen the means of implementation and revitalize the global partnership for sustainable development.

Agriculture and Rural Development

- Mobile App for plant/crop diseases identification and prediction using Machine Learning Techniques.
- Stored grain insect identification: Mobile App for grain sack analysis/scanning for identification of bugs.
- Women friendly improved farm tools for small operations.
- Colour Sorter: Image Processing based affordable grain colour sorting system.
- Micro-climate identification and prediction of local climate of a usually small site or habitat.
- Soil Moisture Monitoring: Wireless system for alerting farmers on the smart phone about how much, when, and where to water their plants or crops.
- Soil nutrients Analyzer Automated Soil micro/ macro-nutrient analyzer
- Food Grain Analysis: Automated system for classification and quality analysis of food grains.
- Cotton picking Automated system for intelligent cotton harvesting machine.
- Seed Sowing Robot Automatic seeding sowing and ploughing machine.
- Automatic weeders for row crops
- Post harvest trash management system (eg. Sugarcane, Maize) (Rural)
- Application for Agro Product & services
- Application for management & Marketing of agro foods & artifacts (rural)

- Replacement to traditional fuel by biodegradable fuel (rural)
- Technique for early detection of pest in cotton
- Perishable crop wastage, storage management
- Application for farming as a service (rural)
- Affordable drone technology for spraying in Indian scenario
- Affordable solution for food processing in Indian Scenario(rural)
- Supply chain management for agro product and services in Indian scenario(rural)
- Telemedicine for rural health care management
- Build an online system for monitoring water quality, leaks, contamination, and managing pipeline networks.
- Systems like Smart Garbage systems , Smart education system
- Development of Low Cost Solar Dryer for Hygienic drying.
- Design and Development of Integrated curing and storage structure for onions.
- Cold Storage Facility for Post-harvest Preservation of Fruits and Vegetables using Solar and Bio methane Heat Based Refrigeration.
- Enhancement of Shelf-life of Perishable Agro Produce using Evaporative Cooling Technology.
- Cost effective mechanism to treat waste water in small villages
- Artificial intelligence enabled robotic trash boat to drive& harvest floating trash from urban drain.
- Priority Road List for Maintenance
- Automatic Assessment of Pavement condition based on road photographs

- **Healthcare Engineering**

• **Medical Imaging**	
• Computed tomography (CT) • Diagnostic radiology • Fluoroscopy • Magnetic resonance imaging (MRI)	• Mammography • Medical imaging • Positron emission tomography (PET) • Ultrasound • X-ray

"Knowledge Brings Freedom"

• **Artificial Organs**	
• 3D printing of organs • Artificial organs • Bionics • Bone tissue engineering	• Cartilage tissue engineering • Prostheses • Regenerative medicine • Tissue engineering
• **Biomaterials**	

• Biomaterial surface characterization • Biomaterial surface modification • Biomaterials for spine • Breast implants • Cardiovascular materials • Cell therapies • Composite resin fillings • Craniofacial materials • Dental amalgam • Dental materials • Denture adhesives • Dermal fillers • Drug delivery materials • Hydrogel for drug delivery	• Hydrogel for healthcare • Hydrogel for bone • Implant materials • Medical ceramics • Medical metals • Nanotechnology • Ophthalmic materials • Organ therapy • Orthopedic materials • Medical polymers • Protein and cells at interfaces • Stem cells • Tissue engineering
• AI based Disease Diagnosis	
• Alzheimer's disease/anesthesiology/ arthritis/ asthma/ attention deficit / hyperactivity disorder (ADHD)/ autism/ brain diseases/ cancer/ cardiovascular medicine/ Chronic Fatigue / chronic obstructive pulmonary disease (COPD)/ tuberculosis/ coronary artery disease/ dementia/ dentistry/ diabetes/ diagnosis Ebola/ epilepsy/ flu/ gastroenterology/ healthcare/ heart disease/hematology/ hepatitis/ kidney disease/ obesity/ ophthalmology/ orthopaedic/ osteoporosis/ pathology/ precision medicine/ stroke/ women's health/ aging	
• Internet of Things (IoT) for healthcare	
• IoT for patient monitoring/IoT for surgery/Medical IoT data security/Medical IoT device integration/Elderly care/Biomedical Device Manufacturing/Wearable Devices	
• Surgery & Robots	
• 3D printing for surgery • Computer-Assisted (Robotic) Surgery • Engineering for neurosurgery • Image-guided surgery • Minimally invasive surgery	• Minimally invasive surgery devices • Robot for heart surgery • Surgical robot • Surgical robot for cardiac surgery
• Transports and Safety	
• Intelligent Transportation System • Alternate Fuel Based Transportation • Advanced Powertrain Technologies • Affordable Energy Storage And Infrastructure For Fast Charging • Active Aerodynamics • Heat Recovery Systems • Intelligent Roads • Long Life, Low Maintenance Roads And Structures • Self Healing Roads • Fog Vision System For Road And Rail	• Active And Passive Safety Technology • Magnetic Levitation Technology • Tilting Train Technology • Autonomous Vehicles • Novel Modes Of Transport (E.G.Evacuated Tube Transport, Hyperloop) • Amphibian And Flying Vehicles • Biomimetics Design For Ship

• **Quality Education, Livelihood and Creative Opportunities**	
• Massively Online Open Courseware (Moocs) • Gaming/ Gamification • Interactive Remotely Controlled Laboratories • Personalised Virtual Teachers • 4g And 5g Communication • Immersive Virtual Reality • Brain Computer Interface And • Machine Augmented Cognition	• Wearable Devices • Digital Identity And Learning Analytics • Automated Evaluation And Assessment Systems • Digital Holography, 3d Imaging And Volumetric/3d Display • 3d Printing • Real Time Translation For Indian Languages ○

• **Energy**	
• Solar PV • Algal Energy • Nuclear Fusion • Fusion Fission Hybrid Reactor • Fast Breeder Reactors For Thorium • Supercritical Coal • Advanced Coal Cycles • Advanced Fossil Fuels Extraction Technologies • Shale Gas • Tight Gas • Gas Hydrate • Hydrogen Energy • Biorefineries	• Hybrid Storage • Fuel Cell • Microbial Fuel Cell • Dc Grids • Smart Grids • Ict Based Smart Monitoring Systems • Wireless Power Transmission • Green And Net Zero Energy Buildings • Smart Windows • Zero Energy Artificial Lighting (E.G. Bioluminescence) • Micro-GasifierCookstove • Brushless Dc (Bldc) Motors

Import Substitutes

Following list 45 items is provided by the Director General of Commercial Intelligence and Statistics (DGCIS), Govt. Of India to encourage the MSME to manufacture indigenously under Atma Nirbhar Bharat Abhiyan.

1. Hand presses 2. Inverter Domestic type up to 5 KVA 3. Film Polythene-Including Wide Width Film 4. Toggle Switches 5. Valves metallic 6. Anklets web Khaki 7. Plaster of paris 8. Stoneware jars 9. Centrifugal Pumps and suction and Delivery 10. Air/ Room cooler manufacturing 11. Domestic House wiring with PVC insulated Aluminium 12. Corrugated Paper Board and Boxes 13. Pressure Die Casting up to 0.75kg 14. Rubber cord 15. Distribution Board upto 15 amps 16. Artistic wooden furniture 17. Squirrel cage Induction Motor 18. Spiked Boots, Skiboots and shoes 19. Steel cross bars, cross arms, clamps, arching horn, brackets 20. Dust Shield leather 21. Domestic electric appliances – Food mixer, wet grinder and food processor 22. M S Tie bars 23. Naphthalene Balls	24. Cotton Wool (Non-Absorbent) 1. Tent poles 2. Augur (carpenter) 3. Chrome tanned leather 4. Nuts & Bolts or Hand Tools of all Types or Distribution of Board up to 15 amps 5. Office furniture (Wooden chairs/Tables) 6. Pulley wires 7. Paper tapes (Gummed) 8. Insecticides Dust and Sprayers (Manual Only) 9. Street light fittings 10. Windshield wipers (arms and blade only) 11. Transistorized Insulation tester 12. Battery Eliminator, Voltage stabilizer 13. Transformer type welding set confirming to IS 1291[1] 14. Hinges, Hasps and staples 15. Garments (excluding supply from Indian Ordnance Factory 16. Hand tools-Mechanical 17. Cane furniture handloom 18. Sluice valves 19. Wooden boards 20. Teak fabricated round blocks 21. Lubricators
Additional Import Substitutes	

"Knowledge Brings Freedom"

1. Blood Bags - Manufacturing Plant, Detailed Project Report, Profile, Business Plan, Industry Trends, Market Research, Survey, Manufacturing Process, Machinery, Raw Materials, Feasibility Study, Investment Opportunities, Cost and Revenue, Plant Economics.
2. Liquid Organic Fertilizer (Biofertiliser)
3. Selenium Coated Aluminium Drum Used In Plain Paper Copier - Manufacturing Plant, Detailed Project Report, Profile, Business Plan, Industry Trends, Market Research, Survey, Manufacturing Process, Machinery, Raw Materials, Feasibility Study
4. Soyabean Cultivation and Processing For Soy Nuggets (Nutrela), Paneer and Milk - Manufacturing Plant, Detailed Project Report, Profile, Business Plan, Industry Trends, Market Research, Survey, Manufacturing Process, Machinery, Raw Materials, Plant Layout
5. Natural Food Colours - Manufacturing Plant, Detailed Project Report, Profile, Business Plan, Industry Trends, Market Research, Survey, Manufacturing Process, Machinery, Raw Materials, Feasibility Study, Investment Opportunities, Cost and Revenue
6. IMFL, INDIAN MADE FOREIGN LIQUOR (WHISKEY, RUM, GIN, VODKA AND BRANDY) - Manufacturing Plant, Detailed Project Report, Profile, Business Plan, Industry Trends, Market Research, Survey, Manufacturing Process, Machinery, Raw Materials, Feasibility Study
7. SANITARY NAPKINS - Manufacturing Plant, Detailed Project Report, Profile, Business Plan, Industry Trends, Market Research, Survey, Manufacturing Process, Machinery, Raw Materials, Feasibility Study, Investment Opportunities, Cost and Revenue, Plant Layout
8. PHOTOCOPIER CLEANING WEB (Non Woven) - Manufacturing Plant, Detailed Project Report, Profile, Business Plan, Industry Trends, Market Research, Survey, Manufacturing Process, Machinery, Raw Materials, Feasibility Study, Investment Opportunities.
9. MEDICAL DISPOSABLES: Disposable Syringes (Self Destructive) with Needles, Catheters and Mask - Manufacturing Plant, Detailed Project Report, Profile, Business Plan, Industry Trends, Market Research, Survey, Manufacturing Process, Machinery, Raw Materials

"Knowledge Brings Freedom"

10. Technology Domains

	Computer Engineering
1.	Data Science and Artificial Intelligence
2.	Block chain Technology
3.	Image Processing, Pattern Recognition and Computer Vision
4.	Natural Language Processing
5.	Cloud Computing/Edge Computing/Fog Computing
6.	Virtual Reality/Augmented Reality
7.	Gaming/Multimedia/Animation/ Computer Graphics
8.	Computational Biology
9.	Wireless Sensor Networks
10.	Machine Learning & Deep Learning
11.	Multi core Computing
12.	Quantum Computing
13.	Green Computing
14.	Human Computer Interaction
15.	ICT and E-learning
16.	Cybernetics, Virtualization and Parallel Computing
17.	Cryptography, Network Security and Cognitive networks
18.	Cognitive Learning
19.	Big Data Analytics
20.	Bioinformatics
21.	Soft Computing
22.	IOT & Sensor data mining
23.	High Performance Computing & Parallelization
24.	Robotic and Automation

"Knowledge Brings Freedom"

Vision and Mission of Computer Department

Department Vision

To be a premier Computer Engineering Department by achieving excellence in Academics and Research for creating globally competent and ethical professionals.

Department Mission

M1: To develop technologically competent and self-sustained professionals through contemporary curriculum.
M2: To nurture innovative thinking and collaborative research, making a positive impact on society.
M3: To provide a state-of-the art computing environment and learning opportunities through the Center of Excellence.
M4: To foster leadership skills and ethics with holistic development.

"Knowledge Brings Freedom"