S.N.	Program Code- MC307	Course Title DEVOPS PROCESS AUTOMATION	L	T	P	C	CH	Course Type
1	Course Code- 21CAT-741		3	0	0	3	3	Program Core
Pre-requisite	--							
Co-requisite	21CAP-746							
Anti-requisite	--							

a. Course Description

DevOps is a set of practices that combines software development (Dev) and information-technology operations (Ops) which aims to shorten the systems development life cycle and provide continuous delivery with high software quality.

b. Course Objectives

1. Understand importance of DevOps in Today's world, Scope of DevOps in following years, Learn Concepts of DevOps and its relation with Software development process
2. Master Concept of Containerization and its implementation using docker, Hands on Implementation and Use of Different tools in Real Life Examples.
3. Learn Basics, Different Open Source Tools such as Ansible, Chef, Puppet. Jenkins, Vagrant etc. used for development and operations, Concepts Version control with hands on GIT, Monitoring and management operations using Nagios.

c. Course Outcomes

CO1	Understand, Apply and demonstrate DevOps and Software development methods.
CO2	Articulate and monitoring DevOps with the use of Different Open Source Tools for problem solving in IT industry
CO3	Construct software integration and build process through various automation tools
CO4	Dramatize DevOps and Software development tools implement various software applications
CO5	System configuration, Monitoring and troubleshooting using different tools

d. Syllabus

Unit-1	Basics of Software Engineering Agile Methodology and DevOps Process	Contact Hours: 15
Chapter 1.1	**Introduction to DevOps:** The Basics of testing in DevOps, Integration of testing in DevOps, Importance of Continuous testing in DevOps, Tips for Developing DevOps testing strategy, DevOps Testing Tools. *Self Study:* How QA fits in DevOps	
Chapter 1.2	**Version Control in DevOps:** Distributed Version control system: Git, Install Git on Ubuntu, Install Git on Windows. Building the Code, Need for Building the Code.	

Unit-2	Managing Source Code	Contact Hours: 15
Chapter 2.1	**Tools Used for Build Process:** Jenkins. Managing the Build Process: Jenkins Build Server, Managing Build Dependencies, The Final Artefact, Managing the build Process using Jenkins, how to trigger a build from external links, how to Chain Jobs in Jenkins, how to use Command line interface for Jenkins.	
Chapter 2.2	**Continuous Integration and Its Tools:** Introduction to Continuous Integration: Continuous Delivery Pipeline, Setting Up Delivery Pipelines in Jenkins, Security aspects in the build process. Continuous Integrated Tools: Team city: Installing team City, Configuring TeamCity	
Chapter 2.3	**Managing Configuration in DevOps:** Configuration Management, Software Configuration Management, Configuration Management in DevOps, *Self Study:* Configuration Management Tools	
Unit-3	Dockers in DevOps	Contact Hours:15
Chapter 3.1	**Dockers in DevOps:** Introduction to Docker, Virtualization, Virtual Machine Manager (VMM), Types of Virtualization, Docker containers and their purpose, Purpose of using Docker. Docker Architecture: Advantages of Docker's Containers, Underlying Technology, Using Docker Commands, working with Docker, working with a Docker Container, Pushing Docker Images to Docker Repository.	
Chapter 3.2	**Installing Docker for Windows:** Working with Docker Toolbox, Kitematic.	
Chapter 3.3	**PUPPET and CHEF for DevOps:** Introduction to PUPPET: Puppet Architecture, Puppet Installation, Real Time Manifest, CHEF. Saltstack for DevOps: Introduction to Saltstack: Working of Saltstack, Salt-Key. Installing Saltstack: Installation of Salt on Linux, Installation of Salt on Windows. Configuring Saltstack: Configuring Firewall, Configuring Salt Minion, Starting Master and Minion, Accepting Minion Key, Running Commands on Minion. *Self Study:* , Need of Saltstack	

e. **Textbooks / Reference Books**

TEXT BOOKS

T1 Sanjeev Sharma, The DevOps Adoption Playbook: A Guide to Adopting DevOps in a Multi-Speed IT Enterprise, Wiley IBM Press.

T2 Jennifer Davis & Katherine Daniels, Effective DevOps: Building a Culture of collaboration, Affinity and Tooling at Scale, O'Relly Media, Inc.

T3 Paul Swartout, Continuous Delivery and DevOps, Packt Publishing

REFERENCE BOOKS

R1 ArunaRavichandaran, DevOps for Digital Leaders, CA Press Apress.

R2 Nathaniel Felsen, Effective DevOps with AWS, Packt Publishing Ltd.

R3 Michael Duffy, DevOps Automation Cookbook, Packt Publishing Ltd.

f. **Assessment Pattern - Internal and External**

The performance of students is evaluated as follows:

Components	Theory			Practical		
	Internal Assessment	Mid Term Assessment	End Term Assessment	Continuous Assessment	Mid Term Assessment	End Term Assessment
Marks	20	20	40	-	-	-
Total Marks	100			100		

g. **Internal Evaluation Component**

S N o.	Type of Assessment	Weightage of actual conduct	Frequency of Task	Final Weightage in Internal Assessment	Remarks (Graded/Non-Graded)
1	Short Term Paper/ Research Paper in form of assignment with Rubrics.	20 marks of each assignment	One per semester	12 marks	As applicable to course types depicted above.
2	Exam	20 marks for one MST	2 per semester	20 marks	As applicable to course types depicted above.
3	Quiz/Test	6 marks of each quiz	2 per Unit	6 marks	As applicable to course types depicted above.
4	Homework	NA	One per lecture topic (of 2 questions)	Non-Graded: Engagement Task	As applicable to course types depicted above.
5	Discussion Forum	NA	One per Chapter	Non-Graded: Engagement Task	As applicable to course types depicted above.
6	Presentation			Non-Graded: Engagement Task	Only for Self-Study MNG Courses.
7	Attendance and Engagement Score on BB	NA	NA	2 marks	

h. CO-PO Mapping

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

S.N.	Program Code-MC307	Course Title Cloud Security	L	T	P	C	CH	Course Type
2	Course Code-21CAT-742		3	0	0	3	3	Program Core
Pre-requisite	--							
Co-requisite								
Anti-requisite	--							

i. Course Description

DevOps is a set of practices that combines software development (Dev) and information-technology operations (Ops) which aims to shorten the systems development life cycle and provide continuous delivery with high software quality.

j. Course Objectives

4. Understand importance of DevOps in Today's world, Scope of DevOps in following years, Learn Concepts of DevOps and its relation with Software development process
5. Master Concept of Containerization and its implementation using docker, Hands on Implementation and Use of Different tools in Real Life Examples.
6. Learn Basics, Different Open Source Tools such as Ansible, Chef, Puppet. Jenkins, Vagrant etc. used for development and operations, Concepts Version control with hands on GIT, Monitoring and management operations using Nagios.

k. Course Outcomes

CO1	Understand, Apply and demonstrate DevOps and Software development methods.
CO2	Articulate and monitoring DevOps with the use of Different Open Source Tools for problem solving in IT industry
CO3	Construct software integration and build process through various automation tools
CO4	Dramatize DevOps and Software development tools implement various software applications
CO5	System configuration, Monitoring and troubleshooting using different tools

l. Syllabus

Unit-1	Cloud Security	Contact Hours: 15
Chapter 1.1	Introduction to cloud security, internet service security, vulnerability and attacks, cloud security mechanism, privacy and security in cloud storage services, privacy and security in multi clouds.	
Security Threats	Cybercrime, Introduction to threats, classification of cloud threats and countermeasures, Service providers threats, Generic Threat, Threat assessment,	

	facing cloud insiders, trusting the clouds, GDPR requirements for cloud providers Azure security.	
Unit-2	**Azure Security**	**Contact Hours: 15**
Chapter 2.1	Introduction to Azure security center and architecture, requirements before adoption cloud security, legacy azure security center security policy, next generation azure security center security, Azure policy, Azure security center RBAC and permissions.	
Security issues and incident response	Computer recommendations, networking recommendations, storage and data. Understanding security alerts, detection scenarios, accessing security alerts, investing a security issues, responding to security alerts.	
Cloud Defense and Splunk	Threat prevention and threat detection, methods of threat detection, cyber kill chain and fusion alerts, Just-In-Time VM access, Integration security incident and event management solutions, Splunk integration with Azure Security Center.	
Unit-3	Monitor and Identification	Contact Hours:15
Chapter 3.1	Monitoring identity related activities, Integrating security center with Azure Active Directory identity protection, customizing your search.	
Chapter 3.2	Threat intelligence and its use, threat intelligence report in security center, threat intelligence dashboard in security center, Hunting security issues in security center, virtual analyst.	

m.   **Textbooks / Reference Books**

TEXT BOOKS

T1 Sanjeev Sharma, The DevOps Adoption Playbook: A Guide to Adopting DevOps in a Multi-Speed IT Enterprise, Wiley IBM Press.

T2 Jennifer Davis & Katherine Daniels, Effective DevOps: Building a Culture of collaboration, Affinity and Tooling at Scale, O'Relly Media, Inc.

T3 Paul Swartout, Continuous Delivery and DevOps, Packt Publishing

REFERENCE BOOKS

R1 ArunaRavichandaran, DevOps for Digital Leaders, CA Press Apress.

R2 Nathaniel Felsen, Effective DevOps with AWS, Packt Publishing Ltd.

R3 Michael Duffy, DevOps Automation Cookbook, Packt Publishing Ltd.

n.   **Assessment Pattern – Internal and External**

The performance of students is evaluated as follows:

	Theory			Practical		
Components	**Internal Assessment**	**Mid Term Assessment**	**End Term Assessment**	**Continuous Assessment**	**Mid Term**	**End Term**

					Assessment	Assessment
Marks	20	20	40	-	-	-
Total Marks	100			100		

o. **Internal Evaluation Component**

S N o.	Type of Assessment	Weightage of actual conduct	Frequency of Task	Final Weightage in Internal Assessment	Remarks (Graded/Non-Graded)
1	Short Term Paper/ Research Paper in form of assignment with Rubrics.	20 marks of each assignment	One per semester	12 marks	As applicable to course types depicted above.
2	Exam	20 marks for one MST	2 per semester	20 marks	As applicable to course types depicted above.
3	Quiz/Test	6 marks of each quiz	2 per Unit	6 marks	As applicable to course types depicted above.
4	Homework	NA	One per lecture topic (of 2 questions)	Non-Graded: Engagement Task	As applicable to course types depicted above.
5	Discussion Forum	NA	One per Chapter	Non-Graded: Engagement Task	As applicable to course types depicted above.
6	Presentation			Non-Graded: Engagement Task	Only for Self-Study MNG Courses.
7	Attendance and Engagement Score on BB	NA	NA	2 marks	

p. **CO-PO Mapping**

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

S.N.	Program Code-MC307	Course Title	L	T	P	C	CH	Course Type
3	Course Code-21CAH-743	CONTAINERIZATION WITH DOCKER	0	2	4	4	6	Program Elective

Pre-requisite	--
Co-requisite	--
Anti-requisite	--

Course Description

Docker is an open platform for developing, shipping, and running applications. Docker enables you to separate your applications from your infrastructure so you can deliver software quickly. With Docker, you can manage your infrastructure in the same ways you manage your applications. By taking advantage of Docker's methodologies for shipping, testing, and deploying code quickly, you can significantly reduce the delay between writing code and running it in production.

Course Objectives
In this course you'll learn how to use Docker through actual examples. You'll see how to create your own custom images, deploy those images as containers on any machine, and have them communicate to each other or the host machine. By the end of the course you'll be familiar enough with Docker to apply your knowledge towards your own projects.

Course Outcomes

CO 1	Remember the importance of DevOps tools used in software development life cycle
CO 2	Will be able to learn installation of Docker on Windows.
CO 3	Analyze & Illustrate the Containerization of OS images and deployment of applications over Docker
CO 4	Summarize the importance of Software Configuration Management in DevOps
CO 5	Students will be able to interact with containers from other containers or the host machine

Syllabus

Unit-1	Introduction	Contact Hours: 15
Chapter 1.1	**Introduction** Introduction to containers, requirements of containers, Containers vs VMs, Introduction to Dockers, Docker architecture, components of Docker, Docker compose, advantages of Docker.	
Experiment no. 1.1	**i.** Install Docker on linux or windows **ii.** Using docker CLI with commands.	

Chapter 1.2	**Installation of Docker** Docker hub dashboard, creating an automatic build, setting up code and docker hub, docker certified images and verified publisher, docker registry, deploying your own registry, docker trusted registry, reviewing third party registries.	
Chapter 1.3	Docker container commands, docker networking and volumes, Docker Desktop dashboard. Docker compose requirements, Docker compose application, Docker compose commands, using Docker app. Introduction to Docker machine, deployment of local docker host, launch docker host in cloud machine,	
Experiment no. 1.2	You are given a list of 10000 integers and you need to compute the average, create docker image and upload to docker.	
Experiment no. 1.3	Craft a Dockerfile to containerize your static website. Run your Static Website on Docker.	
Experiment no. 1.4	Managing Containers with the Docker CLI	
Unit-2	**Docker components**	**Contact Hours: 15**
Chapter 2.1	**Windows containers, setting up docker host to window containers, Docker swarm, creating and managing swarm, managing cluster, Docker swarm services and stacks, Load balancing, overlays and scheduling. Introduction to Portainer, Portainer and Docker Swarm. Running Docker in public cloud: AWS, Azure, Google Cloud.**	
Experiment no. 2.1	Deploy the portfolio website on Microsoft Azure Docker.	
Chapter 2.2	**Docker and Kubernetes: Introduction to Kubernetes, using kubernetes and Docker Desktop, Kubernetes and other Docker tools, Deploying Kubernetes using Minikube, kind, MicroK8s, K3s.**	
Experiment no. 2.1	Building a Custom Docker Image for a Web Application	
Experiment no. 2.3	Maintaining State with Docker Volumes	
Unit-3	**Docker Storage, Networking and Kubernetes**	**Contact Hours:15**
Chapter 3.1	**Microsoft Azure Kubernetes. Docker security:container consideration, docker commands, Docker bench security application, third party security services. Technical requirements of docker deployment, monitoring Docker and Kubernetes.**	
Experiment no. 3.1	Working with Docker Compose	
Experiment no. 3.2	Creating a Private Docker Image Repository	
Experiment no. 3.2	Cleaning Up Old Containers and Docker Images	

Textbooks

- Karl Matthias & Sean P. Kane, Docker: Up and Running, O'Reilly Publication.
- Len Bass,Ingo Weber,Liming Zhu,DevOps, A Software Architects Perspective, Addison-Wesley-Pearson Publication.
- John Ferguson Smart,Jenkins, The Definitive Guide, O'Reilly Publication.
- Learn to Master DevOps by Star EduSolutions.

Reference Books

- Sanjeev Sharma and Bernie Coyne,DevOps for Dummies, Wiley Publication
- Httermann, Michael, DevOps for Developers,Apress Publication.

● Joakim Verona, Practical DevOps,Pack publication

Assessment Pattern - Internal and External

		Theory			Practical		
Components	Internal Assessment	Mid Term Assessment	End Term Assessment	Continuous Assessment	Mid Term Assessment	End Term Assessment
Marks	10	20	40	30	-	-
Total Marks	100					

Internal Evaluation Component

S No.	Type of Assessment	Weightage of actual conduct	Frequency of Task	Final Weightage in Internal Assessment (Prorated Marks)	Remarks
1	Short Term Paper	10 marks		8 marks	
2	Mid-Semester Test**	20 marks for one MST	2 per semester	20 marks	
3	Presentation***			Non-Graded: Engagement Task	Only for Self Study MNG Courses.
4	Homework	NA	One per lecture topic (of 2 questions)	Non-Graded: Engagement Task	
5	Discussion Forum	NA	One per chapter	Non-Graded: Engagement Task	
6	Attendance and Engagement Score on BB	NA	NA	2 marks	
7	Continuous Assessment of practical's	20 Marks for each practical		30 Marks	

CO-PO Mapping

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

S.N.	Program Code- MC307	Course Title	L	T	P	C	CH	Course Type
4	Course Code- 21CAP-747	DEVOPS PROCESS AUTOMATION LAB	0	0	4	2	4	PC
Pre-requisite	--							
Co-requisite	21CAT-741							
Anti-requisite	--							

a. **Course Description**
   DevOps is a set of practices that combines software development (Dev) and information-technology operations (Ops) which aims to shorten the systems development life cycle and provide continuous delivery with high software quality.

b. **Course Objectives**
   ● Master Concept of Containerization and its implementation using docker
   ● Learn Different Open-Source Tools such as Chef, Puppet. Jenkins, etc. used for development and operations.
   ● Concepts Version control with hands on GIT, Monitoring and management operations using Nagios.

c. **Course Outcomes**

CO1	Understand, Apply and demonstrate DevOps and Software development methods.
CO2	Articulate and monitoring DevOps with the use of Different Open-Source Tools for problem solving in IT industry
CO3	Construct software integration and build process through various automation tools.
CO4	Dramatize DevOps and Software development tools implement various software applications.
CO5	System configuration, Monitoring and troubleshooting using different tools.

d. **Syllabus**

Unit-1	Implementing Version control tool (Git, Github) Implementing branches and tags Implementing build automation tool (Maven)	Contact Hours: 20
Experiment no. 1.1	● Installation of Git ● Common Git Commands ● Configuring Git ● Creating Repositories ● Creating a Commit	
Experiment no. 1.2	● Visualizing Branches ● Branch Naming Conventions ● Creating a new Branch ● Handling Merge Conflicts ● Handling tags ● Force push	
Experiment no. 1.3	● Install and configure Apache Maven ● Create sample project by implementing maven	

	● Dependency resolution using POM.xml ● Implementing maven repository	
Experiment no. 1.4	● Install and configure Chef Workstation, Chef Server and Chef Node. ● Creating admin chef user and organization.	
Unit-2	**Implementing Configuration management using Chef.** **Installation and configuration of Puppet** **Implementing Configuration management.**	**Contact Hours: 20**
Experiment no. 2.1	● Creating Cookbooks and Recipes ● Installing Management Console for Chef Server ● Configuration of Starter Kit for WorkStation ● Validating the Connection b/w Server and Workstation ● Uploading the Cookbook ● Adding a Node ● Managing Node Run List	
Experiment no. 2.2	● Installation of puppet ● Configuration of master node ● Configuration of agent node	
Experiment no. 2.3	● Creating puppet manifest file ● Implementing configuration management using master node	
Experiment no. 2.4	● Install and configure Teamcity ● Configuration of database	
Unit-3	**Implementing CI/CD using Teamcity** **Installation and configuration of Nagios**	**Contact Hours: 20**
Experiment no. 3.1	● Configure and Run Your First Build	
Experiment no. 3.2	● Install and configure Nagios Server ● Creating hostgroups.cfg file ● Create Nagios config check script ● Adding users to Nagios ● Streamlining setup with templates.cfg ● Adding servers in Nagios to be monitored	

e. **Textbooks:**

● Sanjeev Sharma, The DevOps Adoption Playbook: A Guide to Adopting DevOps in a Multi-Speed IT Enterprise, Wiley IBM Press.
● Jennifer Davis & Katherine Daniels, Effective DevOps: Building a Culture of collaboration, Affinity and Tooling at Scale, O'Relly Media, Inc.
● Paul Swartout, Continuous Delivery and DevOps, Packt Publishing.

Reference Books: –

· Aruna Ravichandaran, DevOps for Digital Leaders, CA Press Apress.
· Nathaniel Felsen, Effective DevOps with AWS, Packt Publishing Ltd.
· Michael Duffy, DevOps Automation Cookbook, Packt Publishing Ltd

f. **Assessment Pattern - Internal and External**

	Theory	Practical

Components	Internal Assessment	Mid Term Assessment	End Term Assessment	Continuous Assessment	Mid Term Assessment	End Term Assessment
Marks	20	20	60	-	-	-
Total Marks	100					

g. Internal Evaluation Component

Sr. No.	Type of Assessment Task	Weightage of actual conduct	Frequency of task	Final Weightage in Internal Assessment (Prorated Marks)	Remarks
1.	Practical Worksheet (In Assignment with rubrics Category) and Class-room Learning	20 marks for each experiment	8-10 experiments	40 marks	Depending upon no. of experiments
2.	Mid-Term Test	20 marks	1 per semester	12 marks	At-least after the completion of 5 experiments.
3.	Discussion Forum/Short Digital Assignment/ Assignment with Rubrics to submit design/Portfolio	4 marks for each task	1 per semester	4 marks	
4.	Presentation*	------		Non Graded: Engagement Task	
5.	Attendance and BB Engagement Score	-----		4 marks	End Semester

h. CO-PO Mapping

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

S. N.	Program Code- MC307	Course Title	L	T	P	C	CH	Course Type
5	Course Code- 21CAY-749	INSTITUTIONAL/ INDUSTRIAL TRAINING	0	0	0	3*	0	PC
Pre-requisite	--							
Co-requisite	--							
Anti-requisite	--							

Course Description

The course learn how to work with various technologies.

Course Objectives

1. To have hands on experience in the student's relevant field so that they can relate and strengthen what has been taught at the department.
2. Student will develop and promote collaboration between industry and university in promoting knowledgeable society.
3. Student will increase self-confidence of students and helps in finding their own proficiency.

Course Outcomes

CO1	Apply interpersonal skills and knowledge to communicate professionally
CO2	Analyze objectives in collaboration with other members clearly for carrying project or task in hand.
CO3	Acquire communication skill with group of co-workers and learn proper behaviour of corporate life in industrial sector.
CO4	Attain good moral values related to responsibility, sincerity, dedication to work and trustworthiness.
CO5	Create the outcome based mini-project as a solution as per current industry standards and tools.

Assessment Pattern - Internal and External

Components	Theory			Practical		
	Internal Assessment	Mid Term Assessment	End Term Assessment	Continuous Assessment	Mid Term Assessment	End Term Assessment
Marks	-	-	-	60	-	40
Total Marks	100					

Internal Evaluation Component

Sr No.	Type of Assessment	Weightage of actual conduct	Frequency of Task	Final Weightage in Internal Assessment	Remarks (Graded/Non-Graded)
1	Short Term Paper/Research Paper in form of Assignment with Rubrics.	20 marks of each term paper	One per semester	12 marks	As applicable to course types depicted above.
2	Quiz.	6 marks of each quiz	2 per Unit	6 marks	As applicable to course types depicted above
3	. Mid-Semester Test*	20 marks for one MST	2 per semester	20 marks	As applicable to course types depicted above.
4	Presentation**	NA	NA	Non Graded: Engagement Task	Only for Self Study MNG Courses.
5	Homework	NA	One per lecture topic (of 2 questions)	Non-Graded: Engagement Task	As applicable to course types depicted above.
6	Discussion Forum	NA	One per Chapter	Non Graded: Engagement Task	As applicable to course types depicted above.
7	Attendance and BB Engagement Score	NA	NA	2 marks	End Semester

CO-PO Mapping

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

S. N.	Program Code-MC307	Course Title INSTITUTIONAL/ INDUSTRIAL TRAINING	L	T	P	C	CH	Course Type
6	Course Code-21CAR-751		0	0	6	3	6	PC

Pre-requisite	--
Co-requisite	--
Anti-requisite	--

Course Description

The Minor Project work segment is exclusively planned to help students to work and implement new ideas. Adopting standard procedure of software development and Hands on experience, you have to develop your project under supervision of our expert developers

Course Objectives
1. This course is designed to help students prepare minor project based on website application or mobile application development.
2. Student will gain hands on experience on innovative technology project
3. Student will learn to solve/work on the real world/practical/theoretical problems involving issues in computer science and engineering.

Course Outcomes

CO1	Apply fundamental and disciplinary concepts and methods in ways appropriate to their principal areas of study.
CO2	Analyze latest advancement in the field of IT and will implement to develop the software.
CO3	Demonstrate skill and knowledge of current information and technological tools and techniques specific to the professional field of study.
CO4	Demonstrate an awareness and application of appropriate personal, societal, and professional ethical standards.
CO5	Able to work in collaboration as a member of a team in developing a solution of a problem.

Syllabus

- Web Site Development/ Mobile Application Development/Business Application
- Front End: Dot Net Framework/ Android Framework/Java
- Backend: Sql Server/ SQLite/Oracle

Assessment Pattern - Internal and External

	Theory			Practical		
Components	Internal Assessment	Mid Term Assessment	End Term Assessment	Continuous Assessment	Mid Term Assessment	End Term Assessment
Marks	-	-	-	60	-	40

Total Marks	100

CO-PO Mapping

Course Outcome	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PO1 2	PSO 1	PSO 2
CO1	3												3	
CO2	2	3											2	
CO3			3											3
CO4	3			3									3	
CO5					3								3	