Cloud Computing

Course Name:	**CourseCode**	**L-T-P**	**Credits**
Cloud Computing	ETMCCC273	3-0-2	4
Type of Course:	Major		
Pre-requisite(s), if any:	Knowledge of computer networks, operating systems, and basic programming concepts.		

Course Perspective. This course introduces students to the fundamentals and applications of cloud computing, covering service models (IaaS, PaaS, SaaS), cloud deployment strategies, virtualization, containerization, and cloud-native development. It emphasizes practical use of public cloud platforms such as AWS, Azure, and Google Cloud, enabling learners to build scalable, reliable, and cost-effective cloud solutions.

The Course Outcomes (COs). On completion of the course the participantswill be able to:

COs	Statements
CO 1	Explaining the key concepts of cloud computing, service models, and cloud infrastructure.
CO 2	Deploying and managing cloud resources using virtualization, containers, and automation tools.
CO 3	Applying cloud-native design principles using public cloud services for scalable application deployment.

CO 4	Evaluating and implementing secure, cost-optimized, and resilient cloud architectures with real-time monitoring.

CO = Course outcomes. A student is expected to have learnt conceptsand demonstrated/developed abilities or skills related to strategic management at the end of the course.

Course Outline:

Unit Number:1	**Introduction to Cloud Computing and Service Models**	**No. of hours: 12**
Topics Covered: • Definition, characteristics, benefits of cloud computing • Evolution from traditional computing to cloud paradigms • Service models: IaaS, PaaS, SaaS • Deployment models: Public, Private, Hybrid, Community clouds • **Real-World Use Case:** Hosting a startup's website and database using cloud infrastructure		
Unit Number: 2	**Virtualization and Containerization**	**No. of hours: 11**
Topics Covered: • Virtual machines vs containers • Hypervisors: Type 1 and Type 2 • Introduction to Docker and container lifecycle • Kubernetes basics: pods, services, deployments		

- **Real-World Use Case:** Deploying a containerized microservice architecture using Docker & Kubernetes

Unit Number:3	Cloud Platforms and DevOps Integration	No. of hours: 11

Topic Covered:

- Overview of AWS, Azure, and GCP services
- Cloud storage (S3, Blob, Cloud Storage), compute (EC2, Lambda, App Engine)
- CI/CD pipelines and Infrastructure as Code (Terraform, CloudFormation)
- Monitoring and logging (CloudWatch, Stackdriver, Azure Monitor)
- **Real-World Use Case:** Automating deployment and scaling of a web app using AWS CI/CD tools

Unit Number:4	Cloud Security, Cost Management, and Design Patterns	No. of hours: 11

Topic Covered

- Identity and Access Management (IAM)
- Cloud encryption, security groups, firewalls
- Cost optimization, pay-as-you-go, billing alerts
- Cloud design patterns: Auto-scaling, fault tolerance, redundancy
- **Real-World Use Case:** Designing a secure, multi-tiered cloud system for an e-commerce portal

Text & Reference Books

- Rajkumar Buyya – *Cloud Computing: Principles and Paradigms*
- Thomas Erl – *Cloud Computing: Concepts, Technology & Architecture*
- Michael Hausenblas – *Containers & Kubernetes for Dummies*
- AWS, Azure, and Google Cloud Documentation
- A. Velte – *Cloud Computing: A Practical Approach*

Learning Outcomes

Inside the Classroom

1. **In-depth Understanding of Machine Learning Algorithms:**
    - Analyze and implement advanced algorithms such as ensemble methods, kernel-based learning, and boosting techniques.
2. **Mathematical Foundations and Optimization Techniques:**
    - Understand the role of convex optimization, gradient-based methods, and regularization techniques in improving algorithm performance.
3. **Algorithmic Design and Efficiency:**
    - Design scalable and efficient algorithms for classification, regression, and clustering with a focus on time and space complexity.
4. **Advanced Topics in Model Evaluation:**
    - Apply cross-validation, ROC analysis, and performance metrics tailored to imbalanced and noisy datasets.
5. **Graph-based and Probabilistic Learning:**

- Explore semi-supervised learning, Markov Random Fields, and Graph Neural Networks.

6. **Interpretability and Explainability:**
   - Understand techniques such as SHAP, LIME, and model-agnostic interpretability for black-box models.
7. **Ethical and Fair ML Practices:**
   - Recognize algorithmic bias, fairness, and transparency issues in deploying ML models.

Outside the Classroom

1. **Independent Exploration of Research Papers:**
   - Critically read and summarize recent research on novel ML algorithms from conferences like NeurIPS, ICML, and CVPR.
2. **Capstone and Mini Projects:**
   - Design, implement, and present real-world applications using advanced ML techniques such as fraud detection, recommendation engines, or anomaly detection.
3. **Use of ML Libraries and Frameworks:**
   - Gain practical experience with libraries like Scikit-learn, XGBoost, LightGBM, CatBoost, and TensorFlow/PyTorch for algorithm development.
4. **Kaggle and Competitive Learning:**
   - Participate in data science competitions to apply advanced ML algorithms on structured and unstructured datasets.
5. **Collaborative Learning and Code Sharing:**
   - Collaborate on GitHub, engage in peer code reviews, and contribute to open-source ML toolkits.
6. **End-to-End Model Deployment Skills:**

- Explore tools like MLflow, ONNX, and Docker for tracking, deploying, and scaling machine learning models in production.

7. **Ethics and Responsible AI Practice:**
   - Evaluate real-world case studies to understand the societal impact and limitations of algorithmic decision-making systems.

Lab Assignment

S.No	Lab Tasks
1.	**LAB TASK 1: Cloud Setup and Resource Provisioning on AWS** **Objective:** To create and manage virtual machines, storage buckets, and VPCs using AWS Console. **Activities:** • Launch EC2 instances and configure security groups • Create S3 buckets and upload/download files • Set up VPC and subnets with internet gateway **Learning Focus:** Resource provisioning, compute & storage services **Tools:** AWS Free Tier, EC2, S3, VPC
2	**LAB TASK 2: Docker Containerization and Kubernetes Deployment** **Objective:** To package applications using Docker and deploy them on Kubernetes. **Activities:** • Write Dockerfiles and build container images

	• Push/pull from Docker Hub • Deploy services using Kubernetes manifests • Scale and update deployments using kubectl **Learning Focus:** Container orchestration, deployment automation **Tools:** Docker, Kubernetes (Minikube/k3s), Docker Hub
3	**LAB TASK 3: Serverless Function and CI/CD Integration** **Objective:** To create a serverless function and automate its deployment using CI/CD tools. **Activities:** • Write and deploy a Lambda function or Azure Function • Connect GitHub repository to a CI/CD pipeline (GitHub Actions) • Deploy updated function on commit **Learning Focus:** Serverless computing, automation, GitOps **Tools:** AWS Lambda / Azure Functions, GitHub Actions
4	**LAB TASK 4: Cloud Monitoring and Security Configuration** **Objective:** To configure IAM, monitor resources, and manage logs. **Activities:** • Create IAM users with specific policies • Enable billing alerts and security audits • Monitor resource usage via CloudWatch or Azure Monitor **Learning Focus:** Identity management, billing control, observability **Tools:** IAM, AWS CloudWatch, Azure Monitor
5	**Capstone Project: Cloud-Based Scalable Web Application Deployment**

	Objective: To build, containerize, deploy, and monitor a full-stack web application on a public cloud using best practices in architecture, automation, and security. **Project Tasks:** • Deploy application with Docker and Kubernetes on AWS/GCP • Implement CI/CD using GitHub Actions or Jenkins • Integrate with cloud-native storage and database • Configure logging, alerts, and IAM policies for access control **Learning Focus:** Full-stack deployment, cost-effective scaling, security, and monitoring **Tools:** Docker, Kubernetes, AWS/GCP, GitHub, Terraform (optional)