Fundamentals of Cloud Computing

Syllabus

Course Overview

- **Course:** CSC 410/510 – Fundamentals of Cloud Computing
- Overview of modern cloud computing concepts and practice
- Emphasis on both *foundations* and *hands-on implementation*
- Public cloud platforms used for coursework (AWS, Azure)
- Upper-division undergraduate / graduate-level course

What This Course Is About

- Cloud computing as computing delivered as a service
- Renting distributed resources instead of owning infrastructure
- How cloud platforms reshape software design and operations
- Why cloud skills are foundational for modern computing careers

Course Goals

- Understand the history and foundations of cloud computing
- Learn virtualization, containerization, and orchestration
- Design secure, scalable cloud architectures
- Programmatically interact with cloud services and APIs
- Work with cloud data systems and ML-oriented workloads

Learning Outcomes

- Explain cloud service and deployment models
- Deploy and manage VMs, containers, and cloud services
- Configure networking, storage, and identity securely
- Automate infrastructure and application deployment
- Design resilient, cost-aware cloud systems

Prerequisites and Assumed Background

- CSC 323: Introduction to Computer Networks
- CSC 363: Software Engineering
- Comfort with programming and basic systems concepts
- Willingness to learn new tools and platforms quickly

Course Format and Instruction Mode

- In-person, synchronous instruction
- Weekly lectures with technical walkthroughs
- Hands-on assignments and multi-week projects
- Readings posted online (no required textbook)
- Course materials and announcements via Brightspace

Assignments and Projects

- **Weekly assignments**
  - Small, focused exercises reinforcing lecture material
- **Major projects (3 total)**
  - Multi-week, applied cloud systems work
  - Emphasis on design, implementation, and trade-offs
- Projects may involve public cloud usage

Exams and Assessments

- **Midterm exam**
  - Scheduled: February 20, 2026
  - Covers foundational concepts and architectures
- **Final exam**
  - University-scheduled exam period
  - Cumulative with emphasis on later material

Grading Breakdown

- Quizzes: **20%**
- Assignments: **5%**
- Projects: **30%**
- Midterm exam: **20%**
- Final exam: **25%**
- Letter grades assigned by percentage thresholds

Attendance and Participation

- Attendance expected for all lectures
- Attendance required for exams
- Notify instructor in advance if issues arise
- Engagement is essential for success in technical material

Technology and Recording Policy

- Laptops permitted for note-taking
- Student recording of lectures not permitted
- Instructor may record audio and provide transcripts
- Use of course technologies must follow university policy

Late Work and Make-Up Exams

- Late assignments accepted with penalty
- 10% reduction per day late
- Make-up exams require advance approval
- Documentation required for exceptional circumstances

Use of AI Tools

- AI is an essential part of modern computing
- AI use permitted or prohibited per assignment
- All AI use must be documented by students
- Grading emphasizes *effective and responsible* AI use

Academic Integrity

- Academic dishonesty is not tolerated
- Students are responsible for their submitted work
- Proper attribution is required for all sources
- Violations handled according to university policy

Course Logistics and Communication

- Instructor: Richard Kelley, Ph.D.
- Office hours: Wednesday, 10am, Pangborn 324
- Questions:
  - Post to Brightspace for general discussion
  - Email instructor when appropriate
- Course website hosts schedule and readings

Contingency and Accessibility

- Course can shift online if required
- Assignments and exams would move electronically
- Students needing accommodations should contact DSS

Week 1 - Foundations of Cloud Computing

- What cloud computing is (and is not): core definitions
- Historical drivers: virtualization, networking, economies of scale
- Cloud service models: IaaS, PaaS, SaaS, FaaS
- Deployment models: public, private, hybrid, multi-cloud
- Economic and operational motivations for cloud adoption

Week 2 - Virtualization Fundamentals

- Hardware virtualization: CPU, memory, and device abstraction
- Hypervisors and virtual machine isolation
- VM lifecycle: images, provisioning, sizing, termination
- Performance considerations and overhead
- When virtual machines are the right abstraction

Week 3 - Containerization

- Motivation for containers vs virtual machines
- OS-level isolation: namespaces, cgroups, filesystem layering
- Building container images and managing dependencies
- Running and networking containers locally and in the cloud
- Common container use cases and pitfalls

Week 4 - Container Orchestration

- Why orchestration is needed at scale
- Core Kubernetes concepts: pods, services, deployments
- Scheduling, scaling, and self-healing workloads
- Configuration and secrets management
- Trade-offs of managed vs self-managed orchestration

Week 5 - Cloud Networking & Storage Systems

- Virtual networks, subnets, routing, and gateways
- Load balancing and traffic distribution
- Object, block, and file storage models
- Performance, durability, and access patterns
- Designing networked storage for cloud applications

Week 6 - Identity and Access Management

- Authentication vs authorization in cloud systems
- IAM users, roles, policies, and permissions
- Least-privilege design principles
- Federated identity and single sign-on
- IAM as a primary cloud security boundary

Week 7 - Cloud Architecture Patterns

- Scalability and elasticity patterns
- High availability and fault tolerance
- Stateless vs stateful service design
- Caching, messaging, and decoupling patterns
- Choosing architectures based on workload requirements

Week 8 - Midterm Exam

- In-class assessment covering Weeks 1-7
- Emphasis on conceptual understanding and system design
- Practical reasoning about trade-offs and architectures

Week 9 - Spring Break

- No class meeting

Week 10 - CI/CD and Deployment Automation

- Continuous integration vs continuous delivery/deployment
- Automated build, test, and deployment pipelines
- Infrastructure as Code fundamentals
- Integrating containers and IaC into pipelines
- Security and reliability concerns in CI/CD systems

Week 11 - Cloud Monitoring and Operations

- Observability: metrics, logs, and traces
- Centralized logging and alerting
- Diagnosing failures in distributed systems
- Reliability, incident response, and postmortems
- Cost monitoring and optimization strategies

Week 12 - Cloud Programming and APIs

- Programming cloud systems via SDKs and CLIs
- REST APIs and service integration patterns
- Event-driven and asynchronous programming models
- Serverless functions and managed execution
- Choosing between serverless, containers, and VMs

Week 13 - Databases and Data Management

- Relational vs NoSQL database models
- Managed cloud database services
- Replication, durability, and consistency
- Integrating databases into cloud applications
- Selecting databases based on access patterns

Week 14 - Data Pipelines and Distributed Processing

- Data pipeline architectures
- Batch vs streaming data processing
- Distributed compute services for data workloads
- Orchestration of data workflows
- Scaling and reliability in data pipelines

Week 15 - Machine Learning Training Architecture

- ML training workflows in the cloud
- Compute-intensive workloads and GPU utilization
- Distributed training strategies
- Data management for large-scale training
- Cost, performance, and reproducibility trade-offs

Week 16 - Machine Learning Inference Architecture

- Serving trained models at scale
- Latency, throughput, and availability requirements
- Batch vs real-time inference
- Model versioning and deployment strategies
- Monitoring and optimizing inference workloads