MK UNIVERSITY

PATAN, GUJARAT

ESTABLISHED BY THE GUJARAT GOVT. **RECOGNIZED BY UGC UNDER SECTION 2(f) OF UGC ACT,1956**

MK University, Patan
Faculty of Engineering Technology,
Department of Information Technology

M. TECH (INFORMATION TECHNOLOGY) SEM-I									
SR NO.	COURSE TYPE	COURSE CODE	COURSE NAME	LECTURE (HRS.)/WEEK	PRACTICAL (HRS.)/WEEK	CREDITS	EXAMINATION		TOTAL MARKS
							INTERNAL	EXTERNAL	
1	MAJOR	MTIT101	ADVANCED MATHEMATICS FOR ENGINEERS	4	0	4	40	60	100
2	MAJOR	MTIT102	ADVANCED ALGORITHMS & DATA STRUCTURES	4	2	6	90	60	150
3	MAJOR	MTIT103	CLOUD COMPUTING & VIRTUALIZATION	4	2	6	90	60	150
4	MINOR	MTIT104	RESEARCH METHDOLOGY & TECHNICAL COMMUNICATION	4	0	4	40	60	100
5	SEC	MTIT105	ENTERPRENURSHIP DEVELOPMENT	4	0	4	40	60	100
TOTAL				20	4	24	300	300	600

M. TECH (INFORMATION TECHNOLOGY) SEM-II									
SR NO.	COURSE TYPE	COURSE CODE	COURSE NAME	LECTURE (HRS.)/WEEK	PRACTICAL (HRS.)/WEEK	CREDITS	EXAMINATION		TOTAL MARKS
							INTERNAL	EXTERNAL	
1	MAJOR	MTIT201	CYBERSECURITY & DIGITAL FORENSICS	4	0	4	40	60	100
2	MAJOR	MTIT202	BLOCK CHAIN	4	2	6	90	60	150
3	MAJOR	MTIT203	BIG DATA ANALYTICS & PROCESSING	4	2	6	90	60	150
4	MINOR	MTIT204	IOT SYSTEMS	4	2	6	90	60	150
5	VAC	MTIT205	BUSINESS COMMUNICATION-I	2	0	2	0	50	50
TOTAL				18	6	24	310	290	600

MK UNIVERSITY

PATAN, GUJARAT

ESTABLISHED BY THE GUJARAT GOVT. **RECOGNIZED BY UGC UNDER SECTION 2(f) OF UGC ACT,1956**

M. TECH (INFORMATION TECHNOLOGY) SEM-III									
SR NO.	COURSE TYPE	COURSE CODE	COURSE NAME	LECTURE (HRS.)/ WEEK	PRACTICAL (HRS.)/ WEEK	CREDITS	EXAMINATION		TOTAL MARKS
							INTERNAL	EXTERNAL	
1	MAJOR	MTIT301	NATURAL LANGUAGE PROCESSING	4	2	6	90	60	150
2	MAJOR	MTIT302	IT SERVICE MANAGEMENT	4	2	6	90	60	150
3	MINOR	MTIT303	MOOC/SWAYAM COURSE	3	0	3	100	00	100
4	VAC	MTIT304	DISSERTATION PHASE-I	0	8	8	100	100	200
TOTAL				11	12	23	380	220	600

M. TECH (INFORMATION TECHNOLOGY) SEM-IV									
SR NO.	COURSE TYPE	COURSE CODE	COURSE NAME	LECTURE (HRS.)/ WEEK	PRACTICAL (HRS.)/ WEEK	CREDITS	EXAMINATION		TOTAL MARKS
							INTERNAL	EXTERNAL	
1	MAJOR	MTIT401	INDUSTRY SEMINARS/WORKSHOPS/INTERNSHIP	0	2	2	50	00	50
2	MINOR	MTIT402	COMPREHENSIVE VIVA VOCE	0	2	2	50	00	50
3	MAJOR	MTIT403	DISEERTATION PHASE-II	0	16	16	200	200	400
4	VAC	MTIT404	BUSINESS COMMUNICATION-II	2	0	2	00	50	50
TOTAL				2	20	22	300	250	550

MK UNIVERSITY

PATAN, GUJARAT

ESTABLISHED BY THE GUJARAT GOVT. RECOGNIZED BY UGC UNDER SECTION 2(f) OF UGC ACT,1956

SEMESTER-I

SUBJECT CODE: MTIT101

SUBJECT NAME: ADVANCED MATHEMATICS FOR ENGINEERS

Course Objectives:

- To provide a rigorous mathematical foundation for advanced engineering modeling and analysis.
- To bridge theoretical mathematics with practical engineering applications.
- To develop problem-solving skills using analytical and computational tools.
- To prepare students for research and development in engineering domains requiring mathematical sophistication.

Course Outcomes: At the end of the course students shall be able to

CO1	Formulate and solve engineering problems using advanced techniques in linear algebra and tensor analysis.
CO2	Apply partial differential equations (PDEs) and transform methods to model dynamical systems and boundary value problems.
CO3	Use variational calculus and optimization methods for engineering design and control problems.
CO4	Analyze stochastic systems and uncertainty propagation using probability theory and statistical methods.

Unit	Content	Credit	Weightage
I	**Advanced Linear Algebra & Tensors for Engineers** ○ Review of vector spaces, eigenvalues, SVD, Jordan form ○ Matrix decompositions (LU, QR, Cholesky, Schur) ○ Tensor algebra: notation, operations, invariants ○ Tensor applications: stress-strain, inertia, constitutive models ○ Numerical linear algebra (conditioning, iterative solvers) • **Applications:** Structural analysis, continuum mechanics, control systems, data compression.	1	25%
II	**Partial Differential Equations & Transform Methods** ○ Classification of PDEs (elliptic, parabolic, hyperbolic) ○ Separation of variables, eigenfunction expansions ○ Green's functions for ODEs and PDEs ○ Integral transforms (Fourier, Laplace, Hankel) for PDEs ○ Introduction to finite element and finite volume concepts	1	25%

	• **Applications:** Heat transfer, wave propagation, fluid dynamics, signal processing.		
III	**Calculus of Variations & Optimization** ○ Functional derivatives, Euler–Lagrange equation ○ Constraints (Lagrange multipliers, isoperimetric problems) ○ Direct methods (Ritz, Galerkin) ○ Optimal control theory (Pontryagin's principle, Hamiltonian formulation) ○ Convex optimization basics (gradient descent, KKT conditions) • **Applications:** Optimal design, trajectory optimization, energy minimization, control systems.	1	25%
IV	**Stochastic Processes & Uncertainty Quantification** ○ Probability spaces, random variables, distributions ○ Stochastic processes (Brownian motion, Poisson process, Markov chains) ○ Itô calculus basics (stochastic differential equations) ○ Uncertainty quantification (Monte Carlo, polynomial chaos, sensitivity analysis) ○ Statistical estimation and regression for engineering data • **Applications:** Risk analysis, reliability engineering, random vibrations, financial engineering, signal noise modelling	1	25%

TEXT BOOKS:

- Kreyszig, E. – *Advanced Engineering Mathematics* (10th ed.) – Wiley.
- Strang, G. – *Linear Algebra and Its Applications* (5th ed.) – Cengage.
- Arfken, G.B., Weber, H.J., Harris, F.E. – *Mathematical Methods for Physicists* (7th ed.) – Academic Press.
- J.N. Reddy – *Applied Functional Analysis and Variational Methods in Engineering* – McGraw-Hill.
- Papoulis, A., & Pillai, S.U. – *Probability, Random Variables and Stochastic Processes* (4th ed.) – McGraw-Hill.

REFERENCE BOOKS:

- Riley, K.F., Hobson, M.P., Bence, S.J. – *Mathematical Methods for Physics and Engineering* (3rd ed.) – Cambridge.
- Gelfand, I.M., & Fomin, S.V. – *Calculus of Variations* – Dover.
- Oksendal, B. – *Stochastic Differential Equations: An Introduction with Applications* (6th ed.) – Springer.
- Holmes, M.H. – *Introduction to Numerical Methods in Differential Equations* – Springer.
- Gould, P. – *Introduction to Linear Elasticity* (for tensor applications) – Springer.

ONLINE RESOURCES:

- Coursera:
  - *Mathematics for Engineers Specialization* (The Hong Kong University of Science and Technology)
  - *Data Science Math Skills* (Duke University)

SUBJECT CODE: MTIT102

SUBJECT NAME: ADVANCED ALGORITHMS AND DATA STRUCTURES

Course Objectives:

- To develop a deep understanding of advanced algorithmic paradigms and their applications in solving complex computational problems.
- To master advanced data structures for efficient storage, retrieval, and manipulation of large-scale data.
- To analyze algorithmic complexity in terms of time, space, and approximation guarantees.
- To design and implement efficient algorithms for real-world problems in distributed systems, AI, and computational geometry.

Course Outcomes: At the end of the course students shall be able to

CO1	Understanding of advanced algorithmic paradigms and their applications in solving complex computational problems.
CO2	Advanced data structures for efficient storage, retrieval, and manipulation of large-scale data.
CO3	Analyze algorithmic complexity in terms of time, space, and approximation guarantees.
CO4	Design and implement efficient algorithms for real-world problems in distributed systems, AI, and computational geometry.

Unit	Content	Credit	Weightage

I	**ADVANCED ANALYSIS TECHNIQUES & DIVIDE-AND-CONQUER**
	• **Divide-and-Conquer Paradigms:**
	  ○ Strassen's matrix multiplication
	  ○ Fast Fourier Transform (FFT) and polynomial multiplication
	  ○ Closest pair of points algorithm
	  ○ Median finding algorithms (deterministic and randomized)
	• **Amortized Analysis:**
	  ○ Aggregate method, accounting method, potential method
	  ○ Dynamic table expansion analysis
	  ○ Fibonacci heaps (structure and amortized analysis)
	• **Competitive Analysis:**
	  ○ Online vs offline algorithms
	  ○ Paging problem and competitive ratios
	Applications:
		1	25%

	• Signal processing (FFT) • Computational geometry • Dynamic resource allocation		

II	**ADVANCED DATA STRUCTURES & PERSISTENCE**
	• **Augmented Data Structures:**
	  ○ Order-statistic trees
	  ○ Interval trees
	  ○ Segment trees with lazy propagation
	• **Persistent Data Structures:**
	  ○ Fully persistent and partially persistent structures
	  ○ Persistent segment trees
	  ○ Persistent hash arrays
	• **Spatial Data Structures:**
	  ○ K-D trees
	  ○ Quad trees and Octrees
	  ○ R-trees and B+ trees for spatial indexing
	• **Probabilistic Data Structures:**
	  ○ Bloom filters
	  ○ Count-min sketch
	  ○ Hyper Log for cardinality estimation
	• **Succinct Data Structures:**
	  ○ Bit vectors with rank and select operations
	  ○ Succinct trees
	Applications:
	• Database indexing
	• Geographic information systems
	• Streaming data algorithms
	• Compressed data storage
		1	25%

III	**ADVANCED GRAPH ALGORITHMS & NETWORK FLOWS**
	• **Advanced Graph Algorithms:**
	  ○ Maximum flow algorithms:
	  ▪ Edmonds-Karp algorithm
	  ▪ Dinic's algorithm
	  ▪ Push-relabel algorithm
	  ○ Minimum cost maximum flow
	  ○ Bipartite matching (Hopcroft-Karp algorithm)
	  ○ Strongly connected components (Kosaraju, Tarjan)
	• **Planar Graphs:**
	  ○ Planarity testing
	  ○ Planar separator theorem
		1	25%

	• **Approximation Algorithms:**
	  ○ Vertex cover (2-approximation)
	  ○ Traveling salesman problem (TSP) approximations
	  ○ Set cover (log n approximation)
	• **Randomized Graph Algorithms:**
	  ○ Minimum cut (Karger's algorithm)
	  ○ Randomized incremental construction
	• **Dynamic Graph Algorithms:**
	  ○ Connectivity queries in dynamic graphs
	  ○ Fully dynamic reachability
	Applications:
	• Network routing
	• Social network analysis
	• Resource allocation problems
	• VLSI design		

IV	**COMPUTATIONAL GEOMETRY & PARALLEL ALGORITHMS**
	• **Computational Geometry:**
	  ○ Convex hull algorithms (Graham scan, Quickhull)
	  ○ Line segment intersection (sweep line algorithm)
	  ○ Voronoi diagrams and Delaunay triangulation
	  ○ Range searching (range trees, kd-trees)
	• **Parallel Algorithm Design:**
	  ○ PRAM model (EREW, CREW, CRCW)
	  ○ Parallel prefix sum
	  ○ Parallel merge sort
	  ○ Parallel matrix multiplication
	• **Multithreaded Algorithms:**
	  ○ Work-span model
	  ○ Parallel dynamic programming
	  ○ Parallel graph algorithms (BFS, connected components)
	• **External Memory Algorithms:**
	  ○ I/O model
	  ○ External sorting and searching
	  ○ B-trees for external memory
	• **Approximation and Hardness:**
	  ○ NP-completeness proofs
	  ○ Polynomial-time approximation schemes (PTAS)
	  ○ Fixed-parameter tractability
		1	25%

	Applications: • Geographic mapping • Computer graphics • Big data processing • Multi-core and distributed computing		

TEXT BOOKS:

- Cormen, T.H., Leiserson, C.E., Rivest, R.L., & Stein, C. – *Introduction to Algorithms* (4th ed.) – MIT Press.
- Kleinberg, J., & Tardos, É. – *Algorithm Design* – Pearson.
- Sedgewick, R., & Wayne, K. – *Algorithms* (4th ed.) – Addison-Wesley.
- Skiena, S.S. – *The Algorithm Design Manual* (3rd ed.) – Springer.

REFERENCE BOOKS:

- Dasgupta, S., Papadimitriou, C., & Vazirani, U. – *Algorithms* – McGraw-Hill.
- Goodrich, M.T., & Tamassia, R. – *Algorithm Design and Applications* – Wiley.
- Mehlhorn, K., & Sanders, P. – *Algorithms and Data Structures: The Basic Toolbox* – Springer.
- de Berg, M., Cheong, O., van Kreveld, M., & Overmars, M. – *Computational Geometry: Algorithms and Applications* (3rd ed.) – Springer.
- Jájá, J. – *An Introduction to Parallel Algorithms* – Addison-Wesley.

ONLINE RESOURCES:

1. Coursera:
   - *Algorithms Specialization* (Stanford University)
   - *Data Structures and Algorithms* (UC San Diego)
2. edX:
   - *Algorithmic Design and Techniques* (UC San Diego)
   - *Advanced Algorithms and Complexity* (UC San Diego)

PRACTICAL LIST:

- FFT Implementation: Implement Fast Fourier Transform and inverse FFT for polynomial multiplication.
- Segment Tree with Lazy Propagation: Build a segment tree supporting range updates and queries.
- Bloom Filter: Implement a Bloom filter with configurable false positive rate.
- Dinic's Algorithm: Implement maximum flow using Dinic's algorithm.
- Hopcroft-Karp Algorithm: Solve bipartite matching for large graphs.
- Convex Hull: Implement Graham scan and compare with Quickhull.
- Parallel Merge Sort: Implement parallel merge sort using OpenMP or pthreads.
- Persistent Segment Tree: Build a persistent segment tree for range queries on historical data.
- K-D Tree: **Implement k-d tree for nearest neighbor search.**
- Karger's Algorithm: **Implement randomized minimum cut algorithm.**
- External Merge Sort: **Simulate external sorting for large datasets.**
- TSP Approximation: **Implement 2-approximation for Metric TSP.**

MK UNIVERSITY

PATAN, GUJARAT

ESTABLISHED BY THE GUJARAT GOVT. RECOGNIZED BY UGC UNDER SECTION 2(f) OF UGC ACT,1956

SUBJECT CODE: MTIT103

SUBJECT NAME: CLOUD COMPUTING AND VIRTUALIZATION

Course Objectives:

- To provide a comprehensive understanding of cloud computing architectures, service models, and deployment models.
- To master virtualization technologies at different layers (server, network, storage) and their role in cloud infrastructure.
- To design, implement, and manage cloud-native applications using containerization, orchestration, and serverless computing.
- To develop skills in cloud security, cost optimization, and performance management across major cloud platforms.
- To prepare students for industry certifications and roles in cloud architecture, DevOps, and cloud security.

Course Outcomes: At the end of the course students shall be able to

CO1	Analyze and compare different cloud service models (IaaS, PaaS, SaaS) and deployment models (public, private, hybrid).
CO2	Design and implement virtualization solutions using hypervisors, containers, and virtual networking.
CO3	Develop and deploy cloud-native applications using container orchestration platforms (Kubernetes) and serverless frameworks.
CO4	Implement cloud security best practices, identity management, and compliance frameworks.

Unit	Content	Credit	Weightage

I	**CLOUD COMPUTING FOUNDATIONS & VIRTUALIZATION**
	• **Cloud Computing Essentials:**
	  ○ Evolution from distributed computing to cloud
	  ○ NIST cloud definition and characteristics
	  ○ Service models: IaaS, PaaS, SaaS, FaaS, DBaaS
	  ○ Deployment models: Public, Private, Hybrid, Community, Multi-cloud
	• **Virtualization Technologies:**
	  ○ Type 1 vs Type 2 hypervisors (Bare-metal vs Hosted)
	  ○ Full virtualization, paravirtualization, hardware-assisted virtualization
	  ○ Hypervisors: VMware ESXi, Microsoft Hyper-V, KVM, Xen
	  ○ Virtual machine migration: Live migration, storage migration
	• **Resource Virtualization:**
	  ○ CPU virtualization (VT-x, AMD-V)
		1	25%

	○ Memory virtualization (Shadow page tables, EPT) ○ I/O virtualization (SR-IOV, VFIO) ○ GPU virtualization (vGPU, GPU passthrough) • **Storage Virtualization:** ○ SAN vs NAS ○ Storage area networks virtualization ○ Software-defined storage (Ceph, vSAN) ○ Object storage vs block storage vs file storage		
II	**CONTAINERIZATION & ORCHESTRATION** • **Container Fundamentals:** ○ Container vs VM: Architecture comparison ○ Linux namespaces (pid, net, mnt, ipc, uts, user) ○ Control groups (cgroups) ○ Union filesystems (OverlayFS, AUFS) • **Docker Ecosystem:** ○ Docker architecture: Docker Engine, Docker Hub, Docker Desktop ○ Dockerfile best practices ○ Docker networking models (bridge, host, overlay, macvlan) ○ Docker storage drivers and volumes • **Container Orchestration:** ○ Kubernetes architecture: Master components, Node components ○ Pods, Services, Deployments, StatefulSets ○ ConfigMaps and Secrets ○ Persistent volumes and storage classes ○ Ingress controllers and service mesh (Istio basics) • **Advanced Container Topics:** ○ Container security: Image scanning, runtime security ○ Container registries (Docker Hub, Harbor, ECR) ○ GitOps with ArgoCD/Flux ○ Serverless containers (AWS Fargate, Google Cloud Run)	1	25%
III	**CLOUD PLATFORMS & SERVICES** • **AWS Core Services:** ○ Compute: EC2, Lambda, ECS, EKS ○ Storage: S3, EBS, EFS, Glacier ○ Database: RDS, DynamoDB, Redshift ○ Networking: VPC, CloudFront, Route 53	1	25%

	○ Security: IAM, KMS, Secrets Manager • **Azure Core Services:** ○ Compute: Virtual Machines, App Service, Azure Functions ○ Storage: Blob Storage, Azure Files, Managed Disks ○ Database: Azure SQL, Cosmos DB ○ Networking: Virtual Network, Azure CDN, Load Balancer • **Google Cloud Platform:** ○ Compute: Compute Engine, Cloud Functions, GKE ○ Storage: Cloud Storage, Persistent Disk ○ Database: Cloud SQL, Firestore, Bigtable ○ Networking: VPC, Cloud Load Balancing ○ AI/ML services: Vertex AI, AutoML • **Cloud Management:** ○ Infrastructure as Code: Terraform, CloudFormation, ARM Templates ○ Configuration management: Ansible, Chef, Puppet ○ Monitoring and logging: CloudWatch, Azure Monitor, Stackdriver ○ Cost management and optimization tools		
IV	**CLOUD SECURITY, MIGRATION & FUTURE TRENDS** • **Cloud Security:** ○ Shared responsibility model ○ Identity and Access Management (IAM) best practices ○ Network security: Security groups, NACLs, web application firewalls ○ Data encryption: At-rest, in-transit, client-side ○ Compliance frameworks: GDPR, HIPAA, PCI-DSS, SOC 2 ○ Security monitoring and incident response • **Cloud Migration Strategies:** ○ 6 R's of migration: Rehost, Refactor, Revise, Rebuild, Replace, Retain ○ Migration assessment tools (AWS MGN, Azure Migrate) ○ Database migration strategies • **Multi-cloud and Hybrid Cloud:** ○ Multi-cloud management challenges ○ Hybrid cloud connectivity (VPN, Direct Connect, ExpressRoute)	1	25%

	○ Cloud management platforms (vRealize, CloudHealth) ○ Kubernetes federation • **Emerging Trends:** ○ Serverless computing and Function as a Service ○ Edge computing and cloud-edge continuum ○ AI/ML in cloud operations (AIOps) ○ Quantum computing as a service ○ Green cloud computing and sustainability • **Cloud Economics:** ○ Total Cost of Ownership (TCO) analysis ○ Reserved instances vs spot instances ○ Cost optimization techniques ○ FinOps principles		

TEXT BOOKS:

- Buyya, R., Broberg, J., & Goscinski, A. – *Cloud Computing: Principles and Paradigms* – Wiley.
- Maurer, N., & O'Reilly, T. – *Cloud Native Patterns* – Manning Publications.
- Turnbull, J. – *The Docker Book* – Self-published.
- Hightower, K., Burns, B., & Beda, J. – *Kubernetes: Up and Running* (3rd ed.) – O'Reilly.
- Wittig, M., & Wittig, A. – *Amazon Web Services in Action* (2nd ed.) – Manning.

REFERENCE BOOKS:

- Fehling, C., et al. – *Cloud Computing Patterns* – Springer.
- Varia, J., & Mathew, S. – *Overview of Amazon Web Services* – AWS Whitepaper.
- Rhoton, J., & Haukioja, R. – *Cloud Computing Architected* – Recursive Press.
- Luszczek, P. – *High Performance Computing* – CRC Press.
- Srivastava, A., & Pal, R. – *Multi-Cloud Handbook for Developers* – Apress.

ONLINE RESOURCES:

- AWS Training & Certification (Free tier, Skill Builder)
- Microsoft Learn (Azure modules and learning paths)
- Google Cloud Skills Boost (Qwiklabs, Coursera)

PRACTICAL LIST:

- Implement secure VPC with public and private subnets
- Set up IAM policies with least privilege principle
- Perform lift-and-shift migration of VM to cloud
- Implement disaster recovery strategy across regions
- Conduct cloud cost analysis and optimization exercise

MK UNIVERSITY

PATAN, GUJARAT

ESTABLISHED BY THE GUJARAT GOVT. **RECOGNIZED BY UGC UNDER SECTION 2(f) OF UGC ACT,1956**

SUBJECT CODE: MTIT104

SUBJECT NAME: RESEARCH METHDOLOGY AND TECHNICAL COMMUNICATION

Course Objectives:

- To equip engineering graduates with a structured approach to scientific inquiry and problem-solving.
- To develop proficiency in selecting and applying appropriate research methods for engineering investigations.
- To enhance technical communication skills for academia and industry.
- To foster an understanding of research ethics, scholarly publishing, and lifelong learning in research.

Course Outcomes: At the end of the course students shall be able to

CO1	Formulate a research problem, conduct systematic literature reviews, and develop a viable research proposal.
CO2	Design and execute appropriate research methodologies (experimental, numerical, analytical) with consideration for ethics and data integrity.
CO3	Apply statistical tools and software for data analysis, interpretation, and validation of research findings.
CO4	Produce high-quality technical documents (research papers, proposals, theses) and deliver effective technical presentations.

Unit	Content	Credit	Weightage

I	**Foundations of Engineering Research & Problem Formulation**
	• **Topics:**
	  ○ Philosophy of research: inductive vs. deductive reasoning, scientific method in engineering.
	  ○ Types of engineering research: fundamental, applied, experimental, computational, empirical.
	  ○ Problem identification and formulation: research gap analysis.
	  ○ Literature review strategies: databases (Scopus, Web of Science, IEEE Xplore), citation management tools (Zotero, Mendeley), critical analysis of literature.
	  ○ Developing a research proposal: objectives, scope, significance, and work plan.
	• **Applications:** Thesis topic selection, grant proposal writing, project planning.
		1	25%

II	**Research Design, Methods & Ethics**
	• **Topics:**
	  ○ Research design: experimental, quasi-experimental, case study, modeling & simulation.
	  ○ Data collection methods: sensors, surveys,
		1	25%

	instrumentation, simulation outputs. ○ Design of Experiments (DoE): factorial design, Taguchi methods, response surface methodology. ○ Research ethics: plagiarism, fabrication/falsification, authorship, informed consent. ○ Ethical approval process and responsible conduct of research (RCR). • **Applications:** Planning a lab/field experiment, setting up a CFD/FEA study, survey design.		
III	**Data Analysis, Statistics & Software Tools** • **Topics:** ○ Data preprocessing: outlier detection, missing data, normalization. ○ Descriptive and inferential statistics: hypothesis testing (t-test, ANOVA), confidence intervals. ○ Regression analysis: linear, multiple, logistic. ○ Introduction to multivariate analysis and machine learning for engineering data. ○ Software tools: MATLAB/Python (NumPy, SciPy, pandas), R, MiniTab. ○ Data visualization principles: effective graphs, charts, and plots. • **Applications:** Analyzing experimental results, validating computational models, interpreting sensor data.	1	25%
IV	**Technical Communication & Research Dissemination** • **Topics:** ○ Structure of technical documents: research papers, theses, technical reports. ○ Writing strategies: clarity, conciseness, coherence, and argument development. ○ Graphical abstracts, data presentation, and table/figure design. ○ Oral presentations: conference talks, thesis defense, poster design. ○ Publication process: journal selection, peer review, responding to reviewers. ○ Intellectual Property Rights (IPR): patents, copyrights, licensing. ○ Research dissemination: repositories, academic social networks (ResearchGate, LinkedIn), and impact metrics (h-index, citations).	1	25%

	• **Applications:** Paper writing, thesis compilation, conference presentation, patent filing.		

TEXT BOOKS:

- Kothari, C.R. – *Research Methodology: Methods and Techniques* (4th ed.) – New Age International.
- Day, R.A., and Gastel, B. – *How to Write and Publish a Scientific Paper* (9th ed.) – Greenwood.
- Montgomery, D.C. – *Design and Analysis of Experiments* (10th ed.) – Wiley.
- Alley, M. – *The Craft of Scientific Writing* (4th ed.) – Springer.

REFERENCE BOOKS:

- Bordens, K.S., and Abbott, B.B. – *Research Design and Methods: A Process Approach* (11th ed.) – McGraw-Hill.
- Wallwork, A. – *English for Writing Research Papers* (2nd ed.) – Springer.
- Box, G.E.P., Hunter, J.S., and Hunter, W.G. – *Statistics for Experimenters* (2nd ed.) – Wiley.
- IEEE Author Center Guides – *IEEE Publication Services and Products Board*.
- Laplante, P.A. – *Technical Writing: A Practical Guide for Engineers and Scientists* – CRC Press.

ONLINE RESOURCES:

- edX Courses:
  1. *"Principles of Statistical Analysis"* (Microsoft)
  2. *"How to Write and Publish a Scientific Paper"* (KU Leuven)

SUBJECT CODE: MTIT105

SUBJECT NAME: ENTERPRENURSHIP DEVELOPMENT

Course Objectives:

- To cultivate an entrepreneurial mindset among engineering graduates.
- To provide practical tools for transforming technical ideas into viable business ventures.
- To develop skills in business modeling, financial planning, and venture funding.
- To prepare students for startup creation, intrapreneurship, or technology commercialization roles.

Course Outcomes: At the end of the course students shall be able to

CO1	Identify and evaluate entrepreneurial opportunities emerging from technological trends and market gaps.
CO2	Develop a comprehensive business model and validate it using lean startup methodologies and customer discovery.
CO3	Create financial projections, evaluate startup costs, and understand funding mechanisms for technology ventures.
CO4	Formulate a complete investor-ready business plan and deliver an effective pitch to potential stakeholders.

Unit	Content	Credit	Weightage
I	**Entrepreneurial Mindset & Opportunity Identification** • **Topics:** ○ The Entrepreneurial Engineer: Mindset, traits, and role in economic development.	1	25%

	○ Sources of Innovation: Technology push vs. market pull, disruptive vs. sustaining innovation. ○ Opportunity Recognition: Identifying problems worth solving, trend analysis (STEEP), blue ocean strategy. ○ Idea Validation: Lean canvas, hypothesis testing, conducting problem-solution interviews. ○ Intellectual Property Strategy for Startups: Patents, trademarks, trade secrets, and licensing basics. • **Applications:** Spotting opportunities in cleantech, Industry 4.0, medtech, and digital transformation.		
II	**Business Model Design & Customer Development** • **Topics:** ○ Business Model Innovation: Business Model Canvas (Osterwalder), Value Proposition Canvas. ○ Customer Discovery & Validation: The "Get Out of the Building" approach, creating MVP (Minimum Viable Product). ○ Market Analysis: TAM, SAM, SOM, competitive analysis, positioning. ○ Pricing Strategies for Tech Products: Cost-plus, value-based, subscription, freemium models. ○ Go-to-Market Strategy: Sales channels, partnerships, digital marketing fundamentals. • **Applications:** Designing scalable models for SaaS, hardware-as-a-service, platform businesses.	1	25%
III	**Startup Finance, Funding & Legal Foundations** • **Topics:** ○ Startup Financials: Building financial models, unit economics, burn rate, runway. ○ Pro Forma Statements: Income statement, cash flow, balance sheet projections. ○ Funding Sources: Bootstrapping, angels, venture capital, crowdfunding, government grants (SBIR, DST). ○ Valuation Methods for Early-Stage Startups: Scorecard, Berkus, risk factor summation.	1	25%

	○ Legal Structures & Compliance: Company registration (LLP, Pvt Ltd), shareholder agreements, ESOPs, compliance essentials. ○ Term Sheet Fundamentals: Key clauses, negotiation basics. • **Applications:** Preparing for seed funding, managing cash flow, cap table management.		
IV	**Business Planning, Pitching & Scaling Ventures** • **Topics:** ○ The Business Plan: Executive summary, company description, product/service, market analysis, marketing plan, management team, financial projections. ○ The Art of Pitching: Investor pitch deck structure, storytelling, demo preparation. ○ Building the Team: Co-founder selection, hiring early employees, advisory boards. ○ Operational Planning: Supply chain, quality, scaling production. ○ Growth Strategies: Scaling challenges, pivoting, exit strategies (acquisition, IPO). ○ Social Entrepreneurship & Ethics: Creating social impact, ethical leadership. • **Applications:** Crafting investor pitches, developing operational roadmaps, planning for scale.	1	25%

TEXT BOOKS:

- Osterwalder, A., and Pigneur, Y. – *Business Model Generation* – Wiley.
- Ries, E. – *The Lean Startup* – Penguin.
- Blank, S., and Dorf, B. – *The Startup Owner's Manual* – K & S Ranch.
- Barrow, C., Barrow, P., and Brown, R. – *The Business Plan Workbook* (10th ed.) – Kogan Page.

REFERENCE BOOKS:

- Aulet, B. – *Disciplined Entrepreneurship* – Wiley.
- Mullins, J. – *The New Business Road Test* (5th ed.) – FT Publishing.
- Kawasaki, G. – *The Art of the Start 2.0* – Portfolio Penguin.
- Thiel, P. – *Zero to One* – Crown Business.
- Maurya, A. – *Running Lean* (2nd ed.) – O'Reilly.

ONLINE RESOURCES:

- Coursera: *"Entrepreneurship Specialization"* (Wharton), *"Startup Entrepreneurship"* (Technion).
- edX: *"Entrepreneurship in Emerging Economies"* (HarvardX), *"Innovation and Entrepreneurship"* (DelftX).

Udemy: Courses on Business Plan Writing, Startup Funding, and Digital Marketing.

MK UNIVERSITY

PATAN, GUJARAT

ESTABLISHED BY THE GUJARAT GOVT. RECOGNIZED BY UGC UNDER SECTION 2(f) OF UGC ACT,1956

SEMESTER-II

SUBJECT CODE: MTIT201

SUBJECT NAME: CYBER SECURITY AND DIGITAL FORENSICS

Course Objectives:

- To provide a comprehensive understanding of cybersecurity principles, threat landscapes, and defense mechanisms.
- To develop expertise in network security, cryptography, and secure system design methodologies.
- To master offensive security techniques, penetration testing methodologies, and vulnerability assessment.
- To explore digital forensics, incident response, and security operations center (SOC) management.
- To prepare students for cybersecurity certifications and roles in threat intelligence, security architecture, and forensic analysis.

Course Outcomes: At the end of the course students shall be able to

CO1	Analyze cybersecurity threats, vulnerabilities, and attack vectors across different environments.
CO2	Design and implement secure network architectures, cryptographic solutions, and access control systems.
CO3	Conduct penetration testing, vulnerability assessment, and security auditing using industry-standard tools.
CO4	Perform digital forensics investigations, incident response, and malware analysis.

Unit	Content	Credit	Weightage

I	**ADVANCED CRYPTOGRAPHY & NETWORK SECURITY**
	• **Cryptographic Foundations:**
	  ○ Advanced symmetric encryption: AES modes (GCM, CCM), ChaCha20
	  ○ Public-key cryptography: RSA, ECC, post-quantum cryptography basics
	  ○ Hash functions: SHA-3, BLAKE2, Argon2 for password hashing
	  ○ Digital signatures: ECDSA, EdDSA, RSA-PSS
	  ○ Key management and PKI implementation
	• **Network Security Protocols:**
	  ○ TLS 1.3 deep dive: handshake, cipher suites, session resumption
	  ○ IPsec: ESP, AH, IKEv2
	  ○ DNSSEC implementation and deployment
	  ○ Secure email protocols: S/MIME, PGP
	  ○ Wireless security: WPA3, 802.1X/EAP
	• **Secure Network Architecture:**
	  ○ Zero Trust Architecture (ZTA) implementation
		1	25%

	○ Software-Defined Perimeter (SDP) ○ Micro-segmentation strategies ○ Network deception technologies (honeypots, honeynets) • **Cloud Network Security:** ○ Cloud security groups and NACLs ○ Web Application Firewalls (WAF) configuration ○ API security and management ○ DDoS protection mechanisms		
II	**PENETRATION TESTING & VULNERABILITY ASSESSMENT** • **Methodologies & Frameworks:** ○ PTES, OWASP Testing Guide, NIST SP 800-115 ○ Red team vs blue team vs purple team operations ○ Threat modeling: STRIDE, DREAD, Attack trees • **Reconnaissance & Scanning:** ○ Passive and active reconnaissance techniques ○ Network scanning: Nmap advanced scripting ○ Vulnerability scanning: Nessus, OpenVAS, Qualys ○ Web application scanning: Burp Suite, OWASP ZAP • **Exploitation Techniques:** ○ Buffer overflows: Stack-based, heap-based, ROP chains ○ Web exploitation: SQLi, XSS, CSRF, SSRF, XXE ○ Binary exploitation: Format string, integer overflow ○ Post-exploitation: Privilege escalation, lateral movement • **Advanced Persistent Threats (APT):** ○ APT lifecycle and tactics ○ Living-off-the-land techniques ○ Command and control (C2) infrastructure ○ Defense evasion techniques • **Mobile & IoT Security:** ○ Android/iOS application security testing ○ IoT protocol vulnerabilities (MQTT, CoAP) ○ Firmware analysis and reverse engineering	1	25%
III	**DIGITAL FORENSICS & INCIDENT RESPONSE**	1	25%

	• **Forensic Foundations:**
	  ○ Digital evidence handling and chain of custody
	  ○ Legal considerations: CFAA, GDPR, CCPA
	  ○ Forensic acquisition: Disk imaging, memory acquisition
	  ○ Timeline analysis and event reconstruction
	• **Disk & File System Forensics:**
	  ○ File system analysis: NTFS, EXT4, APFS, exFAT
	  ○ File carving and data recovery techniques
	  ○ Anti-forensic detection and countermeasures
	  ○ Cloud storage forensics
	• **Memory & Malware Forensics:**
	  ○ Volatility framework advanced usage
	  ○ Malware analysis: Static, dynamic, behavioral
	  ○ Reverse engineering malicious binaries
	  ○ Ransomware analysis and decryption techniques
	• **Network Forensics:**
	  ○ Packet analysis with Wireshark and tcpdump
	  ○ NetFlow analysis for anomaly detection
	  ○ Intrusion detection system (IDS) log analysis
	  ○ Encrypted traffic analysis
	• **Incident Response:**
	  ○ NIST SP 800-61 incident handling guide
	  ○ IR team roles and responsibilities
	  ○ Threat intelligence integration
	  ○ SOC operations and SIEM management		

IV	**SECURITY OPERATIONS & EMERGING TECHNOLOGIES**
	• **Security Operations Center (SOC):**
	  ○ SOC architecture and tooling
	  ○ Security Information and Event Management (SIEM)
	  ○ Security Orchestration, Automation, and Response (SOAR)
	  ○ Threat intelligence platforms (TIP)
	• **Cloud & Container Security:**
	  ○ Cloud Security Posture Management (CSPM)
	  ○ Container security: Image scanning, runtime protection
		1	25%

	○ Kubernetes security: Pod security policies, network policies ○ Serverless security considerations • **IoT & OT Security:** ○ Industrial Control System (ICS) security ○ SCADA system protection ○ Automotive and medical device security ○ Smart city security challenges • **AI in Cybersecurity:** ○ ML for threat detection and anomaly detection ○ Adversarial machine learning ○ AI-powered security analytics ○ Automated vulnerability discovery • **Compliance & Governance:** ○ ISO 27001, NIST CSF, GDPR compliance ○ Risk assessment methodologies ○ Security awareness and training programs ○ Third-party risk management • **Emerging Threats & Future Trends:** ○ Quantum computing impact on cryptography ○ 5G security challenges ○ Space system cybersecurity ○ Bio-digital convergence security		

TEXT BOOKS:

- Stallings, W., & Brown, L. – *Computer Security: Principles and Practice* (4th ed.) – Pearson.
- Skoudis, E., & Liston, T. – *Counter Hack Reloaded* (2nd ed.) – Prentice Hall.
- Casey, E. – *Digital Evidence and Computer Crime* (3rd ed.) – Academic Press.
- Ligh, M., et al. – *The Art of Memory Forensics* – Wiley.
- Scarfone, K., & Mell, P. – *Guide to Intrusion Detection and Prevention Systems* – NIST SP 800-94.

REFERENCE BOOKS:

- Bejtlich, R. – *The Practice of Network Security Monitoring* – No Starch Press.
- Clark, M. – *Network Forensics* – Prentice Hall.
- Howard, M., & Lipner, S. – *The Security Development Lifecycle* – Microsoft Press.
- Engebretson, P. – *The Basics of Hacking and Penetration Testing* (3rd ed.) – Syngress.
- Chapple, M., et al. – *CISSP Official Study Guide* (9th ed.) – Sybex.

ONLINE RESOURCES:

- Cybrary: Free cybersecurity courses
- TryHackMe / HackTheBox: Hands-on hacking labs
- SANS Cyber Aces: Free security training
- PentesterLab: Web penetration testing exercises

PRACTICAL LIST:

- Build mini-SOC with ELK Stack and Wazuh
- Implement automated threat hunting pipeline
- Develop cloud security monitoring dashboard
- Create AI-based anomaly detection system
- Conduct risk assessment and compliance audit

SUBJECT CODE: MTIT202

SUBJECT NAME: BLOCK CHAIN

Course Objectives:

- To provide a comprehensive understanding of blockchain fundamentals, architectures, and consensus mechanisms.
- To explore cryptocurrency systems, smart contracts, and decentralized applications (dApps) development.
- To analyze blockchain security, privacy mechanisms, and cryptographic foundations.
- To examine enterprise blockchain platforms, interoperability solutions, and real-world applications.
- To prepare students for roles in blockchain development, architecture, and research in Web3 ecosystems.

Course Outcomes: At the end of the course students shall be able to

CO1	Analyze and compare different blockchain architectures, consensus algorithms, and cryptographic foundations.
CO2	Design, develop, and deploy smart contracts and decentralized applications on major blockchain platforms.
CO3	Implement blockchain solutions with appropriate security measures, privacy protocols, and scalability techniques.
CO4	Evaluate enterprise blockchain platforms and design solutions for specific industry use cases.

Unit	Content	Credit	Weightage

I	**BLOCKCHAIN FUNDAMENTALS & CRYPTOGRAPHIC FOUNDATIONS**
	• **Introduction to Distributed Ledger Technology:**
	  ○ Evolution from distributed databases to blockchain
	  ○ Key characteristics: Decentralization, Immutability, Transparency
	  ○ Types of blockchains: Public, Private, Consortium, Hybrid
	  ○ Blockchain vs traditional databases
	• **Cryptographic Foundations:**
	  ○ Hash functions: SHA-256, Keccak (SHA-3), RIPEMD-160
	  ○ Digital signatures: ECDSA, EdDSA, Schnorr signatures
	  ○ Public Key Infrastructure (PKI) in blockchain
		1	25%

	○ Merkle Trees and Patricia Merkle Trees • **Blockchain Architecture:** ○ Block structure: Header, Transactions, Nonce ○ Blockchain data structure ○ Network architecture: P2P networks, gossip protocols ○ Wallets: Hot vs Cold, HD wallets, mnemonic phrases • **Consensus Mechanisms:** ○ Proof of Work (Bitcoin, Ethereum 1.0) ○ Proof of Stake (Ethereum 2.0, Cardano) ○ Delegated Proof of Stake (EOS, TRON) ○ Practical Byzantine Fault Tolerance (PBFT, Hyperledger) ○ Alternative consensus: Proof of Authority, Proof of Space/Time		
II	**CRYPTOCURRENCIES & SMART CONTRACTS** • **Bitcoin Ecosystem:** ○ Bitcoin protocol: UTXO model, Script language ○ Bitcoin transactions: Inputs, Outputs, Fees ○ Mining process and difficulty adjustment ○ SegWit, Taproot, and Bitcoin improvements ○ Lightning Network for scalability • **Ethereum Virtual Machine (EVM):** ○ Account-based model vs UTXO ○ Gas mechanism and fee market (EIP-1559) ○ EVM architecture and opcodes • **Smart Contract Development:** ○ Solidity programming language ○ Smart contract architecture: Functions, Modifiers, Events ○ Security patterns and common vulnerabilities (SWC Registry) ○ Testing and debugging smart contracts ○ Upgradeability patterns (Proxy, Diamond) • **Decentralized Applications (dApps):** ○ Web3.js and Ethers.js libraries ○ Frontend integration with blockchain ○ IPFS for decentralized storage ○ Oracle integration (Chainlink)	1	25%
III	**ENTERPRISE BLOCKCHAIN & SCALABILITY SOLUTIONS** • **Enterprise Blockchain Platforms:** ○ Hyperledger Fabric: Architecture, Channels, Private Data	1	25%

	○ Hyperledger Besu: Ethereum for enterprises ○ Corda: Financial-grade blockchain ○ Quorum: Enterprise Ethereum with privacy • **Blockchain Interoperability:** ○ Cross-chain communication protocols ○ Bridges: Trusted vs Trustless ○ Polkadot and Cosmos ecosystems ○ Atomic swaps and hashed timelock contracts • **Scalability Solutions:** ○ Layer 1 solutions: Sharding, DAG-based blockchains ○ Layer 2 solutions: Rollups (Optimistic, ZK-Rollups) ○ State channels and payment channels ○ Sidechains and plasma chains • **Privacy-Enhancing Technologies:** ○ Confidential transactions ○ Zero-knowledge proofs applications ○ zk-SNARKs vs zk-STARKs ○ Privacy coins: Monero, Zcash • **Blockchain as a Service (BaaS):** ○ AWS Managed Blockchain ○ Azure Blockchain Service ○ IBM Blockchain Platform		
IV	**ADVANCED APPLICATIONS & FUTURE TRENDS** • **Decentralized Finance (DeFi):** ○ Automated Market Makers (Uniswap, Curve) ○ Lending protocols (Aave, Compound) ○ Yield farming and liquidity mining ○ Stablecoins: Algorithmic vs Collateralized ○ Risks and security in DeFi • **NFTs and Digital Assets:** ○ NFT standards and metadata ○ Gaming and metaverse applications ○ Royalty mechanisms and IP management ○ Fractionalized NFTs • **DAO Governance:** ○ Decentralized Autonomous Organizations ○ Governance tokens and voting mechanisms ○ Treasury management ○ Legal implications of DAOs • **Blockchain in Enterprise:** ○ Supply chain traceability ○ Digital identity and verifiable credentials ○ Healthcare data management	1	25%

	○ Real estate tokenization • **Regulatory and Legal Aspects:** ○ Global regulatory landscape ○ Anti-Money Laundering (AML) compliance ○ Know Your Customer (KYC) solutions ○ Tax implications of cryptocurrencies • **Future Trends:** ○ Web3 and decentralized internet ○ Central Bank Digital Currencies (CBDCs) ○ Blockchain and IoT convergence ○ Quantum-resistant cryptography ○ Sustainability and green blockchain initiatives		

TEXT BOOKS:

- Antonopoulos, A.M. – *Mastering Bitcoin* (2nd ed.) – O'Reilly.
- Antonopoulos, A.M., & Wood, G. – *Mastering Ethereum* – O'Reilly.
- Narayanan, A., et al. – *Bitcoin and Cryptocurrency Technologies* – Princeton University Press.
- Voshmgir, S. – *Token Economy* (2nd ed.) – BlockchainHub.
- Bashir, I. – *Mastering Blockchain* (4th ed.) – Packt.

REFERENCE BOOKS:

- Swan, M. – *Blockchain: Blueprint for a New Economy* – O'Reilly.
- Tapscott, D., & Tapscott, A. – *Blockchain Revolution* – Portfolio.
- Andreas, M. – *The Internet of Money* (Volumes 1-3) – Merkle Bloom.
- Harz, D., & Gudgeon, L. – *Advances in Financial Machine Learning* – Wiley.
- Hyperledger Foundation – *Hyperledger Fabric Documentation* – Linux Foundation.

ONLINE RESOURCES:

- Coursera: *Blockchain Specialization* (University at Buffalo)
- edX: *Blockchain Fundamentals* (Berkeley)

PRACTICAL LIST:

- Build complete DeFi application with staking and farming
- Create DAO with governance token and voting system
- Implement supply chain traceability solution
- Develop blockchain-based digital identity system
- Analyze and audit smart contract for security vulnerabilities

MK UNIVERSITY

PATAN, GUJARAT

ESTABLISHED BY THE GUJARAT GOVT. RECOGNIZED BY UGC UNDER SECTION 2(f) OF UGC ACT,1956

SUBJECT CODE: MTIT203

SUBJECT NAME: BIG DATA ANALYTICS AND PROCESSING

Course Objectives:

- To provide a comprehensive understanding of big data technologies, distributed computing frameworks, and cloud-based AI/ML services.
- To develop expertise in processing, analyzing, and deriving insights from massive-scale datasets using scalable cloud platforms.
- To implement end-to-end machine learning pipelines on cloud infrastructure for production-level AI systems.
- To prepare students for roles in data engineering, cloud AI solution architecture, and large-scale AI system deployment.

Course Outcomes: At the end of the course students shall be able to

CO1	Design and implement scalable data processing pipelines using big data technologies.
CO2	Develop and deploy machine learning models on cloud platforms using managed AI services.
CO3	Optimize distributed ML training and inference for large-scale datasets.
CO4	Architect and implement production-grade cloud-native AI systems with MLOps principles.

Unit	Content	Credit	Weightage
I	**Big Data Foundations and Distributed Processing** **Topics:** • Introduction to Big Data: 5 Vs (Volume, Velocity, Variety, Veracity, Value) • Hadoop ecosystem: HDFS, MapReduce, YARN • Apache Spark: RDDs, DataFrames, Spark SQL, Spark MLlib • Distributed data processing patterns and optimizations • NoSQL databases: MongoDB, Cassandra, Redis • Data lakes vs. data warehouses • Real-time stream processing: Apache Kafka, Apache Flink • **Applications:** Log analysis, clickstream processing, IoT data aggregation	1	25%
II	**Cloud Computing for AI** • Cloud computing fundamentals: IaaS, PaaS, SaaS, FaaS • Major cloud platforms: AWS, Azure, Google Cloud Platform • Cloud storage solutions: S3, Azure Blob, Google Cloud Storage	1	25%

	• Containerization and orchestration: Docker, Kubernetes for AI workloads • Serverless computing for AI: AWS Lambda, Azure Functions • Cloud-based ML platforms: SageMaker, Azure ML, Vertex AI • Cost optimization and resource management in cloud AI • **Applications:** Scalable model training, automated ML pipelines, batch inference		
III	**Distributed Machine Learning and MLOps** **Topics:** • Distributed training frameworks: Horovod, PyTorch DDP, TensorFlow Distributed • Model parallelism and data parallelism • Hyperparameter tuning at scale: Ray Tune, Optuna on cloud • Feature stores: Feast, Tecton • ML pipeline orchestration: Apache Airflow, Kubeflow Pipelines • Model versioning and registry: MLflow, DVC • Monitoring and logging for ML systems: Prometheus, Grafana • CI/CD for machine learning (MLOps) • **Applications:** Large language model training, recommendation systems at scale	1	25%
IV	**Advanced Analytics and Cloud AI Services** **Topics:** • Big data analytics tools: Apache Hive, Presto, Apache Druid • Data visualization at scale: Tableau, Power BI, Apache Superset • Managed AI services: Computer Vision APIs, NLP services, speech recognition • Vector databases and similarity search: Pinecone, Weaviate, Milvus • Graph analytics and processing: Neo4j, Amazon Neptune • Edge-cloud AI integration • Security, compliance, and governance in cloud AI • Ethical considerations: Bias in large-scale AI, data privacy • **Applications:** Real-time recommendation engines, fraud detection, predictive maintenance	1	25%

TEXT BOOKS:

• Chambers, B., & Zaharia, M. – *Spark: The Definitive Guide* – O'Reilly.

- Lakshmanan, V., Robinson, S., & Munn, M. – *Machine Learning Design Patterns* – O'Reilly.
- Kimball, R., & Ross, M. – *The Data Warehouse Toolkit* (3rd ed.) – Wiley.
- Huyen, C. – *Designing Machine Learning Systems* – O'Reilly.

REFERENCE BOOKS:

- White, T. – *Hadoop: The Definitive Guide* (4th ed.) – O'Reilly.
- Guller, M. – *Big Data Analytics with Spark* – Apress.
- Kleppmann, M. – *Designing Data-Intensive Applications* – O'Reilly.
- AWS/Azure/GCP – *Official certification guides and whitepapers*.

ONLINE RESOURCES:

- Databricks Community Edition: Free Spark cluster
- Google Colab Pro: GPU access for distributed ML experiments
- AWS Educate/Azure for Students: Free cloud credits
- Coursera: *Big Data Specialization* (University of California San Diego)
- edX: *Data Science and Machine Learning in the Cloud* (Microsoft)

PRACTICAL LIST:

1. **Big Data Processing with Apache Spark**

Task: Process a multi-terabyte dataset (e.g., Wikipedia dump or Twitter stream) using PySpark on Databricks. Perform data cleaning, aggregation, and analysis. Compare performance between RDD and Data Frame APIs.

2**. End-to-End Cloud ML Pipeline**

Task: Build a complete ML pipeline on AWS SageMaker or Azure ML including data ingestion, feature engineering, model training (XGBoost/Deep Learning), hyperparameter tuning, and deployment. Implement automated retraining with Airflow.

3, **Distributed Deep Learning Training**

Task: Train a large vision or language model (ResNet50 or BERT) using distributed training across multiple GPUs/instances on cloud. Implement with Horovod or PyTorch DDP. Compare training time and cost vs. single instance.

4. **Real-time Analytics and AI Serving**

Task: Create a real-time fraud detection system using Kafka for streaming, Spark Streaming/Flink for processing, and a pre-trained model served via TensorFlow Serving on Kubernetes. Build a monitoring dashboard with Grafana.

MK UNIVERSITY

PATAN, GUJARAT

ESTABLISHED BY THE GUJARAT GOVT. RECOGNIZED BY UGC UNDER SECTION 2(f) OF UGC ACT,1956

SUBJECT CODE: MTIT204

SUBJECT NAME: IOT SYSTEMS

Course Objectives:

- To provide comprehensive understanding of IoT architectures, protocols, and ecosystems.
- To develop expertise in IoT device design, sensor networks, and edge computing.
- To implement IoT solutions using cloud platforms, data analytics, and visualization tools.
- To explore IoT security, privacy challenges, and ethical considerations.
- To prepare students for designing and deploying IoT solutions in smart cities, healthcare, industry, and agriculture.

Course Outcomes: At the end of the course students shall be able to

CO1	Analyze and design IoT system architectures considering scalability, interoperability, and performance requirements.
CO2	Develop IoT solutions using appropriate sensors, microcontrollers, communication protocols, and edge computing devices.
CO3	Implement IoT data pipelines with cloud integration, real-time analytics, and visualization dashboards.
CO4	Evaluate IoT security threats and implement security measures at device, network, and application levels.

Unit	Content	Credit	Weightage

I	**IoT FUNDAMENTALS & ARCHITECTURE**
	• **Introduction to IoT Ecosystem:**
	  ○ Evolution and components of IoT systems
	  ○ IoT architecture models: Three-layer, Five-layer, SOA-based
	  ○ IoT vs M2M vs WSN
	  ○ IoT reference architecture (ISO/IEC 30141)
	• **Sensors & Actuators:**
	  ○ Sensor types: Temperature, humidity, motion, proximity, gas
	  ○ Sensor characteristics: Accuracy, precision, resolution, sensitivity
	  ○ Actuator types: Motors, relays, servos, solenoids
	  ○ Signal conditioning and interfacing
	• **IoT Hardware Platforms:**
	  ○ Microcontrollers: Arduino, ESP32, STM32
	  ○ Single-board computers: Raspberry Pi, BeagleBone
	  ○ System-on-Chip (SoC) and System-on-Module (SoM)
	  ○ Development boards comparison and selection criteria
		1	25%

	• **Communication Protocols:**
	  ○ Short-range: Bluetooth Low Energy (BLE), Zigbee, Z-Wave, NFC
	  ○ Medium-range: Wi-Fi (802.11ah), Thread
	  ○ Long-range: LoRaWAN, Sigfox, NB-IoT, LTE-M
	  ○ Protocol selection matrix for different applications
	• **Edge Computing in IoT:**
	  ○ Edge vs Fog vs Cloud computing
	  ○ Edge device architectures
	  ○ Real-time processing requirements
	  ○ Resource-constrained computing challenges		

II	**IoT NETWORKING & PROTOCOL STACK**
	• **IoT Protocol Stack:**
	  ○ Physical and MAC layer protocols
	  ○ Network layer: 6LoWPAN, RPL
	  ○ Transport layer: TCP/UDP adaptations for IoT
	  ○ Application layer protocols: MQTT, CoAP, HTTP/2, AMQP
	• **MQTT Deep Dive:**
	  ○ Publish-subscribe architecture
	  ○ Quality of Service levels (0,1,2)
	  ○ MQTT-SN for constrained networks
	  ○ Broker implementations: Mosquitto, HiveMQ, AWS IoT Core
	• **CoAP Protocol:**
	  ○ RESTful architecture for constrained devices
	  ○ Observe pattern for resource monitoring
	  ○ Block-wise transfers
	  ○ CoAP security with DTLS
	• **IoT Network Design:**
	  ○ Mesh networking topologies
	  ○ Gateway design and implementation
	  ○ Network scalability considerations
	  ○ Quality of Service in IoT networks
	• **IoT Data Formats:**
	  ○ JSON, CBOR, Protocol Buffers
	  ○ Semantic interoperability with JSON-LD
	  ○ Efficient data serialization techniques
	  ○ Binary vs text-based protocols comparison
		1	25%

III	**IoT CLOUD PLATFORMS & DATA ANALYTICS** • **IoT Cloud Platforms:** ○ AWS IoT Core: Device Shadow, Rules Engine, Thing Registry	1	25%

	○ Azure IoT Hub: Device Twins, IoT Edge, DPS ○ Google Cloud IoT Core: Device Manager, Pub/Sub integration ○ Open-source platforms: ThingsBoard, Kaa, Mainflux • **Device Management:** ○ Over-the-air (OTA) updates ○ Device provisioning and authentication ○ Device lifecycle management ○ Remote monitoring and control • **IoT Data Processing:** ○ Stream processing vs batch processing ○ Time-series databases: InfluxDB, TimescaleDB ○ Real-time analytics with Apache Kafka, Apache Flink ○ Data aggregation and compression techniques • **Data Visualization:** ○ Dashboard design principles ○ Real-time monitoring interfaces ○ Alert and notification systems ○ Business intelligence integration • **AI/ML in IoT (AIoT):** ○ Edge AI: TensorFlow Lite, PyTorch Mobile ○ Predictive maintenance algorithms ○ Anomaly detection in sensor data ○ Federated learning for privacy preservation		
IV	**IoT SECURITY, APPLICATIONS & FUTURE TRENDS** • **IoT Security Challenges:** ○ Threat landscape: Device tampering, data breaches, DDoS attacks ○ Security architecture: Defense in depth ○ Device identity and authentication ○ Secure boot and firmware updates • **Cryptographic Solutions:** ○ Lightweight cryptography ○ Hardware Security Modules (HSM) and TPM ○ Key management for constrained devices ○ Secure communication protocols • **Privacy & Compliance:** ○ Data privacy regulations (GDPR, CCPA) ○ Privacy by design principles	1	25%

	○ Anonymization and pseudonymization techniques ○ Ethical considerations in IoT deployment • **Industry Applications:** ○ **Smart Cities:** Traffic management, waste management, smart lighting ○ **Industrial IoT (IIoT):** Predictive maintenance, asset tracking, quality control ○ **Healthcare IoT:** Remote patient monitoring, wearable devices, telemedicine ○ **Agriculture IoT:** Precision farming, soil monitoring, automated irrigation ○ **Smart Home:** Energy management, security systems, home automation • **Emerging Trends:** ○ Digital Twins for simulation and monitoring ○ 5G and IoT convergence ○ Sustainable IoT and energy harvesting ○ Quantum-safe cryptography for IoT ○ Swarm intelligence and collaborative IoT		

TEXT BOOKS:

- Bahga, A., & Madisetti, V. – *Internet of Things: A Hands-On Approach* – VPT Publishers.
- Holler, J., et al. – *From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence* – Academic Press.
- Morrow, R. – *IoT and Edge Computing for Architects* (2nd ed.) – Packt Publishing.
- Ashton, K. – *How to Fly a Horse: The Secret History of Creation, Invention, and Discovery* – Anchor Books.
- Minerva, R., Biru, A., & Rotondi, D. – *Towards a Definition of the Internet of Things (IoT)* – IEEE IoT Initiative.

REFERENCE BOOKS:

- Rosenthal, D., & Varshney, K. – *IoT Security: Advances in Authentication* – Wiley.
- Raj, P., & Raman, A. – *The Internet of Things: Enabling Technologies, Platforms, and Use Cases* – CRC Press.
- Guth, J., et al. – *A Reference Architecture for the Internet of Things* – WSO2.
- Botta, A., de Donato, W., Persico, V., & Pescapé, A. – *Integration of Cloud Computing and Internet of Things: A Survey* – Future Generation Computer Systems.
- Zanella, A., Bui, N., Castellani, A., Vangelista, L., & Zorzi, M. – *Internet of Things for Smart Cities* – IEEE IoT Journal.

ONLINE RESOURCES:

1. **Coursera:**
   - *IoT (Internet of Things) Specialization* (University of California, Irvine)
   - *An Introduction to Programming the Internet of Things (IoT)* (University of California, Irvine)
   - *AWS IoT: Developing and Deploying an Internet of Things* (AWS)

MK UNIVERSITY

PATAN, GUJARAT

ESTABLISHED BY THE GUJARAT GOVT. RECOGNIZED BY UGC UNDER SECTION 2(f) OF UGC ACT,1956

SEMESTER-III

SUBJECT CODE: MTIT301

SUBJECT NAME: NATURAL LANGUAGE PROCESSING

Course Objectives:

- To provide a comprehensive understanding of linguistic foundations and computational techniques for natural language understanding and generation.
- To develop expertise in modern NLP techniques including statistical models, neural approaches, and transformer architectures.
- To implement and evaluate NLP systems for real-world applications such as text classification, machine translation, and conversational AI.
- To prepare students for research and development in cutting-edge NLP technologies and emerging language AI applications.

Course Outcomes: At the end of the course students shall be able to

CO1	Understand linguistic structures and apply preprocessing techniques for text data.
CO2	Design and implement statistical and neural NLP models for various language tasks.
CO3	Build and fine-tune transformer-based models for advanced NLP applications.
CO4	Develop end-to-end NLP systems and evaluate their performance with appropriate metrics.

Unit	Content	Credit	Weightage
I	**Foundations of NLP and Text Processing** **Topics:** • Introduction to linguistics: Morphology, syntax, semantics, pragmatics • Text preprocessing: Tokenization, stemming, lemmatization, stop-word removal • Regular expressions and pattern matching • N-gram language models and smoothing techniques • Word representations: One-hot encoding, TF-IDF, co-occurrence matrices • Collocations and statistical measures (PMI, chi-square) • Named Entity Recognition (NER) using rule-based and statistical methods • **Applications:** Text normalization, spell checking, basic information extraction	1	25%
II	**Statistical NLP and Classical Approaches** **Topics:** • Part-of-Speech tagging: Hidden Markov Models, Viterbi algorithm • Context-Free Grammars and parsing algorithms (CKY, Earley)	1	25%

	• Dependency parsing and constituency parsing • Semantic analysis: Word sense disambiguation, semantic roles • Sentiment analysis using lexicon-based and machine learning approaches • Topic modeling: Latent Dirichlet Allocation (LDA) • Text classification using Naïve Bayes, SVM, and logistic regression • **Applications:** Document classification, opinion mining, topic extraction		
III	**Neural NLP and Word Embeddings** **Topics:** • Distributed representations: Word2Vec (CBOW, Skip-gram), GloVe • Contextual embeddings: ELMo, ULMFiT • Neural sequence models for NLP tasks • Sequence-to-sequence models with attention mechanism • Neural machine translation • Text generation with RNNs and LSTMs • Neural text classification and sentiment analysis • **Applications:** Semantic similarity, document clustering, basic chatbots	1	25%
IV	**Transformer Models and Advanced NLP** **Topics:** • Transformer architecture: Self-attention, multi-head attention, positional encoding • BERT and its variants (RoBERTa, DistilBERT, ALBERT) • GPT models and autoregressive language modeling • Fine-tuning pre-trained language models • Question Answering systems (SQuAD) • Text summarization: Extractive and abstractive approaches • Dialogue systems and conversational AI • Ethical considerations: Bias in language models, fairness, interpretability • **Applications:** Chatbots, document summarization, intelligent search systems	1	25%

TEXT BOOKS:

- Jurafsky, D., & Martin, J.H. – *Speech and Language Processing* (3rd ed. draft) – Pearson.
- Eisenstein, J. – *Introduction to Natural Language Processing* – MIT Press.
- Goldberg, Y. – *Neural Network Methods for Natural Language Processing* – Morgan & Claypool.
- Clark, K., Khandelwal, U., Levy, O., & Manning, C.D. – *What Does BERT Look At?* – ACL.

REFERENCE BOOKS:

- Manning, C.D., & Schütze, H. – *Foundations of Statistical Natural Language Processing* – MIT Press.
- Bird, S., Klein, E., & Loper, E. – *Natural Language Processing with Python* – O'Reilly.
- Sarkar, D. – *Text Analytics with Python* – Apress.
- Howard, J., & Ruder, S. – *Universal Language Model Fine-tuning for Text Classification* – ACL.

ONLINE RESOURCES:

- Coursera: *Natural Language Processing Specialization*
- Stanford Online: *CS224N: Natural Language Processing with Deep Learning*
- Kaggle: NLP competitions and datasets
- Google Colab: GPU access for training NLP models

SUBJECT CODE: MTIT302

SUBJECT NAME: IT SERVICE MANAGEMENT

Course Objectives:

- To provide comprehensive understanding of IT Service Management frameworks, principles, and best practices.
- To develop expertise in implementing and managing ITIL 4 practices for service delivery and value creation.
- To explore modern ITSM approaches including DevOps integration, Agile Service Management, and SIAM.
- To analyze IT governance, risk management, compliance, and continuous improvement methodologies.
- To prepare students for ITSM certifications and roles in service design, delivery, and continual improvement.

Course Outcomes: At the end of the course students shall be able to

CO1	Analyze and apply ITIL 4 framework components including Service Value System and Four Dimensions.
CO2	Design and implement IT service management processes aligned with organizational objectives.
CO3	Evaluate and improve service quality using metrics, KPIs, and continual improvement methodologies.
CO4	Integrate ITSM with modern practices including DevOps, Agile, and Cloud Service Management.

Unit	Content	Credit	Weightage

I	**ITSM FOUNDATIONS & ITIL 4 FRAMEWORK**
	• **Introduction to Service Management:**
	  ○ Evolution of IT Service Management
	  ○ Key concepts: Service, Utility, Warranty, Value co-creation
	  ○ Service relationship model: Service provider, consumer, sponsor
	  ○ Characteristics of services vs products
	• **ITIL 4 Framework:**
	  ○ Service Value System (SVS) components
		1	25%

	○ Guiding Principles: Focus on value, Start where you are, etc. ○ Four Dimensions of Service Management: ▪ Organizations and People ▪ Information and Technology • **Service Management Practices:** ○ General Management Practices ○ Service Management Practices ○ Technical Management Practices ○ Practice integration and relationships • **Service Strategy & Design:** ○ Service portfolio management ○ Business relationship management ○ Service design principles ○ Service catalog management		
II	**SERVICE OPERATIONS & CONTINUAL IMPROVEMENT** • **Service Operation Processes:** ○ Incident Management: Priority, categorization, escalation ○ Problem Management: Reactive vs Proactive, Root Cause Analysis ○ Service Request Management ○ Monitoring and Event Management ○ Access Management • **Service Desk Management:** ○ Service desk structures: Local, centralized, virtual, follow-the-sun ○ Technologies and tools for service desk ○ Metrics: First contact resolution, CSAT, SLA compliance ○ Knowledge Management integration • **Continual Improvement:** ○ Continual Improvement Model ○ Measurement and metrics: CSFs, KPIs, metrics ○ Reporting: Balanced scorecard, service reports ○ Service reviews and improvement register • **Service Level Management:** ○ Service Level Agreements (SLAs) ○ Operational Level Agreements (OLAs) ○ Underpinning Contracts (UCs) ○ Service Level Requirements (SLRs)	1	25%
III	**MODERN ITSM & INTEGRATION PRACTICES** • **DevOps & ITSM Integration:** ○ DevOps principles and practices	1	25%

	○ CALMS model: Culture, Automation, Lean, Measurement, Sharing ○ Integrating ITIL with DevOps ○ Site Reliability Engineering (SRE) practices • **Agile Service Management:** ○ Agile principles in ITSM ○ Scrum and Kanban for service management ○ Agile release management ○ DevOps and Agile toolchain integration • **Cloud Service Management:** ○ Cloud service models (IaaS, PaaS, SaaS) management ○ Cloud financial management (FinOps) ○ Multi-cloud service management ○ Cloud migration impact on ITSM • **SIAM (Service Integration and Management):** ○ Multi-sourcing environment challenges ○ SIAM roles: Service Integrator, Service Provider ○ SIAM processes and functions ○ SIAM implementation roadmap • **Automation in ITSM:** ○ Robotic Process Automation (RPA) ○ AI and ML in service management ○ Chatbots and virtual agents ○ Self-service portals and automation		
IV	**IT GOVERNANCE, RISK & COMPLIANCE** • **IT Governance Frameworks:** ○ COBIT 2019 framework and components ○ ISO/IEC 20000 standards ○ IT Governance Institute (ITGI) framework ○ Aligning IT with business strategy • **Risk Management:** ○ IT risk identification and assessment ○ Risk treatment strategies ○ Business continuity and disaster recovery ○ IT resilience management • **Compliance Management:** ○ Regulatory compliance: GDPR, HIPAA, SOX ○ Industry standards compliance ○ Audit processes and controls ○ Compliance automation tools • **Financial Management for IT Services:** ○ IT budgeting and accounting ○ Chargeback and show back models ○ Total Cost of Ownership (TCO)	1	25%

	○ Return on Investment (ROI) calculations • **Organizational Change Management:** ○ Change enablement practices ○ Organizational change models ○ Stakeholder management ○ Communication strategies • **Emerging Trends:** ○ AIOps for intelligent operations ○ Value Stream Management (VSM) ○ Everything as a Service (XaaS) management ○ Sustainable ITSM practices ○ Remote workforce service management		

TEXT BOOKS:

- Axelos – *ITIL 4 Foundation: ITIL 4 Edition* – The Stationery Office.
- Axelos – *ITIL 4 Direct, Plan and Improve* – The Stationery Office.
- Axelos – *ITIL 4 Drive Stakeholder Value* – The Stationery Office.
- ISACA – *COBIT 2019 Framework: Introduction and Methodology* – ISACA.
- ISO/IEC – *ISO/IEC 20000-1:2018 Service Management System Requirements* – ISO.

REFERENCE BOOKS:

- Brooks, P. – *Metrics for IT Service Management* – Van Haren Publishing.
- Steinberg, R.A. – *Implementing ITIL Configuration Management* – IBM Press.
- Spalding, G. – *ITIL® Continual Service Improvement* – The Stationery Office.
- Lloyd, V. – *ITIL 4 Essentials* – IT Governance Publishing.
- Humble, J., & Farley, D. – *Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation* – Addison-Wesley.
- Kim, G., et al. – *The DevOps Handbook: How to Create World-Class Agility, Reliability, & Security in Technology Organizations* – IT Revolution Press.

ONLINE RESOURCES:

1. Coursera:
   - *IT Infrastructure and Emerging Trends* (University of Minnesota)
   - *ServiceNow ITSM Professional Certificate*
2. edX:
   - *ITIL 4 Foundation* (AXELOS)
   - *IT Project Management* (University of Washington)

PRACTICAL LIST:

- Develop IT governance framework using COBIT
- Conduct IT risk assessment and create risk register
- Design compliance checklist for GDPR requirements
- Create IT service financial model with chargeback
- Develop organizational change plan for ITSM implementation
- Implement AIOps use case for predictive incident management
- Design remote workforce ITSM support model