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 (IT)

B. TECH (INFORMATION TECHNOLOGY) SEM-I										
SR NO.	COURSE TYPE	COURSE CODE	COURSE NAME	LECTURE (HRS.)/ WEEK	Tutorial	PRACTICAL (HRS.)/ WEEK	CREDITS	EXAMINATION		TOTAL MARKS
								INTERNAL	EXTERNAL	
1	MAJOR	BTIT101	Engineering Mathematics-I	4	0	0	4	40	60	100
2	MAJOR	BTIT102	Introduction to Computer Programming (Python/C)	4	0	0	4	40	60	100
3	MAJOR	BTIT103	Engineering Physics	4	0	0	4	40	60	100
4	MINOR	BTIT104	Data Science	4	0	0	4	40	60	100
5	MINOR	BTIT105	Electronics	4	0	0	4	40	60	100
6	VAC	BTIT106	Communication Skills in - I	2	0	0	2	0	50	50
TOTAL				22	0	0	22	200	350	550

B. TECH (INFORMATION TECHNOLOGY) SEM-II									
SR NO.	COURSE TYPE	COURSE CODE	CUORSE NAME	LECTURE (HRS.)/ WEEK	PRACTICAL (HRS.)/WEEK	CREDITS	EXAMINATION		TOTAL MARKS
							INTERNAL	EXTERNAL	
1	MAJOR	BTIT201	Engineering Mathematics-II	4	0	4	40	60	100
2	MAJOR	BTIT202	Data Structures & Algorithms	4	0	4	40	60	100
3	MAJOR	BTIT203	Digital Logic Design	4	1	5	40	60	100
4	MINOR	BTIT204	Engineering Graphics & Design	4	1	5	40	60	100
5	MINOR	BTIT205	Management	4	0	4	40	60	100
6	VAC	BTIT206	Environmental Science	2	0	2	0	50	50
TOTAL				22	2	24	200	350	550

MK UNIVERSITY

PATAN, GUJARAT

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

B. 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	BTIT301	Discrete Mathematics	4	0	4	40	60	100
2	MAJOR	BTIT302	Object-Oriented Programming (Java/C++)	4	1	5	40	60	100
3	MAJOR	BTIT303	Computer Organization & Architecture	4	0	4	40	60	100
4	MAJOR	BTIT304	Database Management Systems	4	1	5	40	60	100
5	MINOR	BTIT305	Economics for Engineers	4	0	4	40	60	100
6	SEC	BTIT306	Software Tool: Git	0	2	2	50	0	50
TOTAL				20	4	24	250	300	550

B. 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	BTIT401	Operating Systems	4	0	4	40	60	100
2	MAJOR	BTIT402	Theory of Computation	4	0	4	40	60	100
3	MAJOR	BTIT403	Microprocessors & Microcontrollers	4	1	5	40	60	100
4	MAJOR	BTIT404	Advanced Data Structures	4	1	5	40	60	100
5	MINOR	BTIT405	Introduction to IoT	4	0	4	40	60	100
6	I	BTIT406	Indian Constitution	2	0	2	0	50	50
TOTAL				22	2	24	200	350	550

MK UNIVERSITY

PATAN, GUJARAT

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

B. TECH (INFORMATION TECHNOLOGY) SEM-V									
SR NO.	COURSE TYPE	COURSE CODE	COURSE NAME	LECTURE (HRS.)/ WEEK	PRACTICAL (HRS.)/WEEK	CREDITS	EXAMINATION		TOTAL MARKS
							INTERNAL	EXTERNAL	
1	MAJOR	BTIT501	Computer Networks	4	1	5	40	60	100
2	MAJOR	BTIT502	Web Technologies	4	0	4	40	60	100
3	MAJOR	BTIT503	Information Security	4	1	5	40	60	100
4	MAJOR	BTIT504	Advanced Database Management System	4	0	4	40	60	100
5	MINOR	BTIT505	Cloud Computing Fundamentals	4	0	4	40	60	100
6	SEC	BTIT506	Mini-Project	0	2	2	50	0	50
TOTAL				20	4	24	250	300	350

B. TECH (INFORMATION TECHNOLOGY) SEM-VI									
SR NO.	COURSE TYPE	COURSE CODE	COURSE NAME	LECTURE (HRS.)/ WEEK	PRACTICAL (HRS.)/WEEK	CREDITS	EXAMINATION		TOTAL MARKS
							INTERNAL	EXTERNAL	
1	MAJOR	BTIT601	System Administration & Maintenance	4	0	4	40	60	100
2	MAJOR	BTIT602	Cryptography & Network Security	4	1	5	40	60	100
3	MAJOR	BTIT603	Advanced Computer Architecture	4	0	4	40	60	100
4	MAJOR	BTIT604	Advanced Web Technologies	4	1	5	40	60	100
5	MINOR	BTIT605	Enterprise Computing	4	0	4	40	60	100
6	SEC	BTIT606	Aptitude & Career Skills	0	2	2	50	0	50
TOTAL				20	4	24	250	300	550

MK UNIVERSITY

PATAN, GUJARAT

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

B. TECH (INFORMATION TECHNOLOGY) SEM-VII									
SR NO.	COURSE TYPE	COURSE CODE	COURSE NAME	LECTURE (HRS.)/ WEEK	PRACTICAL (HRS.)/ WEEK	CREDITS	EXAMINATION		TOTAL MARKS
							INTERNAL	EXTERNAL	
1	MAJOR	BTIT701	IT Project Management	4	0	4	40	60	100
2	MINOR	BTIT702	Cyber Forensics	4	0	4	40	60	100
3	MAJOR	BTIT703	Wireless Network	4	1	5	40	60	100
4	MINOR	BTIT704	IT Service Management	4	0	4	40	60	100
5	Multidisciplinary (MD)	BTIT705	Block Chain Management	4	1	5	40	60	100
6	SEC	BTIT706	Project Phase-I	0	2	2	50	00	50
TOTAL				20	4	24	250	300	550

B. TECH (INFORMATION TECHNOLOGY) SEM-VIII									
SR NO.	COURSE TYPE	COURSE CODE	COURSE NAME	LECTURE (HRS.)/ WEEK	PRACTICAL (HRS.)/ WEEK	CREDITS	EXAMINATION		TOTAL MARKS
							INTERN AL	EXTERN AL	
1	MAJOR	BTIT801	Enterprise Solutions	4	1	5	40	60	100
2	MAJOR	BTIT802	Network Implementation	4	1	5	40	60	100
3	MAJOR	BTIT803	IT Infrastructure	4	0	4	40	60	100
4	SEC	BTIT804	Project Phase-II	0	10	10	100	100	200
TOTAL				12	12	24	220	280	500

MK UNIVERSITY

PATAN, GUJARAT

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

SUBJECT CODE: BTIT101

SUBJECT NAME: ENGINEERING MATHEMATICS-I

Course Objective:

- The concept of rank of a matrix which is used to know the consistency of system of linear equations and also to find the eigen vectors of a given matrix.
- Finding maxima and minima of functions of several variables.
- Applications of first order ordinary differential equations. (Newton's law of cooling, Natural growth and decay)
- How to solve first order linear, nonlinear partial differential equations and also method of separation of variables technique to solve typical second order partial differential equations.
- Solving differential equations using Laplace Transforms.

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

CO1	The concept of rank of a matrix which is used to know the consistency of system of linear equations and also to find the eigen vectors of a given matrix
CO2	Finding maxima and minima of functions of several variables
CO3	Applications of first order ordinary differential equations
CO4	How to solve first order linear, nonlinear partial differential equations and also method of separation of variables technique to solve typical second order partial differential equations

Unit	Content	Credit	Weightage
I	Matrices Introduction, types of matrices-symmetric, skew-symmetric, Hermitian, skew-Hermitian, orthogonal, unitary matrices. Rank of a matrix - echelon form, normal form, consistency of system of linear equations (Homogeneous and Non-Homogeneous). Eigen values and Eigen vectors and their properties (without proof), Cayley-Hamilton theorem (without proof), Diagonalization.	1	25%
II	Functions of Several Variables Limit continuity, partial derivatives and total derivative. Jacobian-Functional dependence and independence. Maxima and minima and saddle points, method of Lagrange multipliers, Taylor's theorem for two variables.	1	25%
III	Ordinary Differential Equations First order ordinary differential equations: Exact, equations reducible to exact form. Applications of first order differential equations - Newton's law of cooling, law of natural growth and decay. Linear differential equations of second and higher order with constant coefficients: Non-homogeneous term of the type f(x) = eax, sinax, cosax, xn , eax V and xn V. Method of variation of parameters.	1	25%

IV	Partial Differential Equations Introduction, formation of partial differential equation by elimination of arbitrary constants and arbitrary functions, solutions of first order Lagrange's linear equation and non-linear equations, Charpit's method, Method of separation of variables for second order equations and applications of PDE to one dimensional (Heat equation).	1	25%

TEXT BOOKS:

1. Higher Engineering Mathematics by B V Ramana ., Tata McGraw Hill.
2. Higher Engineering Mathematics by B.S. Grewal, Khanna Publishers.
3. Advanced Engineering Mathematics by Kreyszig, John Wiley & Sons.

REFERENCE BOOKS:

i)Advanced Engineering Mathematics by R.K Jain & S R K Iyenger, Narosa Publishers. ii)Advanced Engineering Mathematics by Michael Green Berg, Pearson Publishers . iii)Engineering Mathematics by N.P Bali and Manish Goyal.

SUBJECT CODE: BTIT102

SUBJECT NAME: INTRODUCTION TO COMPUTER PROGRAMMING - C

Course Objective:

1. To express algorithms and draw flowcharts in a language independent manner.
2. To teach how to write modular, efficient and readable C programs
3. To impart knowledge in creating and using Arrays of the C data types.
4. To describe the techniques for creating program modules in C using functions and recursive functions.
5. To demonstrate creation of derived data types and perform operations on files.
6. To familiarize pointers and dynamic memory allocation functions to efficiently solve problems

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

CO1	Write, compile and debug programs in C language
CO2	Use different data types in a computer program
CO3	Design programs involving decision structures, loops, arrays and functions
CO4	Identify the difference between call by value and call by reference

Unit	Content	Credit	Weightage
I	Introduction to the C Language – Algorithm, Pseudo code, Flow chart, Background, C Programs, Identifiers, Data Types, Variables, Constants, Input / Output, Operators (Arithmetic, relational, logical, bitwise etc.), Expressions, Precedence and Associatively, Expression Evaluation, Type conversions.	1	25%
II	Statements- Selection Statements(making decisions) – if and switch statements, Repetition statements (loops)-while, for, do-while statements, Loop examples, other statements related to looping – break, continue, go to, Simple C Program examples.	1	25%

III	Functions- Introduction to Structured Programming, Functions-basics, user defined functions, inter function communication(call by value, call by reference), Standard functions. Storage classes-auto, register, static, extern, scope rules, arrays to functions, recursive functions, example C programs.	1	25%
IV	Arrays– Basic concepts, one-dimensional arrays, two – dimensional arrays, multidimensional arrays, C programming examples Pointers – Introduction (Basic Concepts), pointers to pointers, compatibility, Pointer Applications, Arrays and Pointers, Pointer Arithmetic, memory allocation functions, array of pointers, pointers to void, pointers to functions, command –line arguments, Introduction to structures and unions.	1	25%

TEXT BOOKS:

1. Computer Science: A Structured Programming Approach Using C, B.A.Forouzan and R.F. Gilberg, Third Edition, Cengage Learning.

2. The C Programming Language by Brian Kernighan and Dennis Ritchie 2nd edition

REFERENCE BOOKS:

1. Let Us C Yashavant kanetkar BPB.

2. Absolute beginner's guide to C, Greg M. Perry, Edition 2, Publisher: Sams Pub., 1994.

3. Computer Programming and Data Structures by E Balagurusamy, Tata McGraw Hill.

SUBJECT CODE: BTIT103

SUBJECT NAME: ENGINEERING PHYSICS

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

CO1	Gain the knowledge on the basic concepts of oscillations exhibited by various systems in nature
CO2	Study the basic concepts of light through interference and diffraction
CO3	Explore band structure of the solids and classification of materials
CO4	Compare dielectric and magnetic properties of the materials and enable them to design and apply in different fields

Unit	Content	Credit	Weightage
I	HARMONIC OSCILLATIONS : Introduction to harmonic oscillators, simple harmonic oscillator: equation of motion and its solution (complex exponential method), damped harmonic oscillator: equation of motion and its solution, over, critical and lightly-damped oscillators; energy decay in damped harmonic oscillator, Quality factor (qualitative), forced damped harmonic oscillator: equation of motion and its solution.	1	25%
II	WAVEOPTICS: Interference- Introduction, Superposition of waves, interference of light by division of wave front-interference of reflected light in thin films, interference of light by division of amplitude Newton’s rings, Diffraction- difference between Fresnel	1	25%

	and Fraunhofer diffraction, Fraunhofer diffraction due to single slit, Diffraction grating- Grating spectrum and resolving power.		
III	INTRODUCTION TO SOLIDS: Free electron theory (Classical & Quantum): Assumptions, Merits and drawbacks, Fermi level, Density of states, Periodic potential, Bloch's theorem, Kronig – Penny modal, E – K diagram, Effective mass, Origin of energy bands in solids, Classification of materials: Metals, semi-conductors and insulators.	1	25%
IV	Dielectrics: Introduction, Types of polarizations (Electronic and Ionic) and calculation of their polarizabilities, internal fields in a solid, Clausius-Mossotti relation. Magnetism: Introduction, Bohr magneton, classification of dia, para and ferro magnetic materials on the basis of magnetic moment, Properties of anti-ferro and ferri magnetic materials, Hysteresis curve based on Domain theory of ferro magnetism, Soft and hard magnetic materials.	1	25%

TEXT BOOKS:

1. Engineering Physics by Arumugam, Anuradha publications.
2. Engineering Physics- B.K. Pandey, S. Chaturvedi, Cengage Learning.

REFERENCES:

1. Engineering Physics – R.K. Gaur and S.L. Gupta, DhanpatRai Publishers.
2. Engineering Physics, S Mani Naidu- Pearson Publishers.
3. Engineering physics 2nd edition –H.K. Malik and A.K. Singh.
4. Engineering Physics – P.K. Palaniswamy, Scitech publications.
5. Physics by Resnick and Haliday.

SUBJECT CODE: BTIT104

SUBJECT NAME: DATA SCIENCE

Course Objective:

To Impart knowledge about basic computer fundamentals and programming languages for data science.
To Impart knowledge about mathematical and statistical methods for data analysis.
To Empower students with data visualization techniques and tools.
To impart knowledge about the basics of data management and Business Theory.
To impart knowledge about various machine learning techniques used for data analysis.
To enable students to develop data-based machine learning models for solving real-word applications.
To enable students to gain practical experience in programming languages and statistical and machine learning tools for data sciences.

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

CO1	Apply basic data cleaning techniques to prepare data for analysis
CO2	Demonstrate proficiency in using appropriate tools and technology to collect, process, transform, summarize, and visualize data
CO3	Apply various machine learning algorithms in data-based decision-making applications, and draw accurate and useful conclusions through data analysis
CO4	Demonstrate some skills in data retrieval using Structured Query language (SQL)

Unit	Content	Credit	Weightage

I	History of computer, Basic Computer hardware, input and output devices, Basic computer architecture, input output devices, memory and CPU, networking of machines (overview of LAN, MAN, WAN, Internet, Wifi etc), types of computer (workstation, desktop, Smartphone, embedded system, etc.), Overview of Software (system software and application software with examples (mention names only)), Definition of Operating System and functions (mention names of some popular operating systems like Windows, Linux, Android, etc).	1	25%
II	Bit, Byte and Word, Number System (Base, Binary, Decimal, Octal, Hexadecimal), Conversion of number systems, Boolean logic (Boolean Gates), Boolean operators (OR, AND and NOT), ASCII code, Concept of Algorithm and Flowchart.	1	25%
III	Basics of Python programming (with a simple 'hello world' program, process of writing a program, running it, and print statement), Concept of class and object, Data-types (integer, float, string), notion of a variable, Operators (assignment, logical, arithmetic etc.), accepting input from console, conditional statements (If else and Nested If else), Collections (List, Tuple, Sets and Dictionary), Loops (For Loop, While Loop & Nested Loops), iterator, string and fundamental string operations (compare, concatenation, sub-string etc.), Function, recursion.	1	25%
IV	Overview of linear and nonlinear data structure (definition, schematic view and difference), array (1D, 2D and its relation with matrix, basic operations: access elements using index, insert, delete, search), stack (concept of LIFO, basic operations: Push, Pop, peek, size), queue (concept of FIFO, basic operations: Enqueue, Dequeue, peek, size), use of List methods in Python for basic operations on array, stack and queue, overview of NumPy library and basic array operations (arrange, shape(), ndim(), dtype() etc.), binary tree (definition and schematic view only) .	1	25%

SUBJECT CODE: BTIT105

SUBJECT NAME: ELECTRONICS

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

CO1	The role of key electronic components and subsystems in the data acquisition pipeline.
CO2	Analyse simple analog and digital electronic circuits relevant to sensor interfacing and signal conditioning.
CO3	Design and implement basic data acquisition systems using microcontroller boards and common sensors
CO4	Evaluate the trade-offs in resolution, sampling rate, and power consumption in electronic data systems.

Unit	Content	Credit	Weightage

I	**Introduction to Electronics in the Data Pipeline** • The role of electronics in the data science workflow: **Sensing → Acquisition → Processing → Analysis**. • Overview of system components: Sensors, signal conditioners, Analog-to-Digital Converters (ADCs), microcontrollers/embedded processors. • Core concepts: Voltage, current, power, energy (crucial for battery-powered IoT/edge devices).	1	25%
II	**Semiconductor Devices for Computing & Sensing** • Diodes: Basic operation, LEDs (output indicators). • **Transistors (BJT & MOSFET):** As a switch (fundamental building block of digital logic and memory). Introduction to amplification. • Operational Amplifiers (Op-amps): Basic configurations (inverting, non-inverting) for signal conditioning (amplification, filtering).	1	25%
III	**Digital Electronics Foundations** • Logic Gates (AND, OR, NOT, NAND, NOR) and Boolean algebra. • Combinational circuits: Multiplexers, Decoders, Adders. • Sequential circuits: Flip-flops (SR, D), Registers, and basic counter concepts. • **Microcontroller/System-on-Chip (SoC) Introduction:** Block diagram (CPU, Memory, I/O ports, ADC, DAC). Contrast with general-purpose CPUs.	1	25%
IV	**Data Acquisition and Interface Electronics** • **Sensors & Transducers:** Types (Temperature, Light, Motion, Image), analog output. • **Signal Conditioning:** Noise, filtering (passive RC filters), amplification for ADC compatibility. • **Analog-to-Digital Conversion (ADC):** Sampling, quantization, resolution, sampling rate (Nyquist Theorem). • **Interfacing Protocols:** Serial Communication (UART, I2C, SPI) - the "language" for sensor data transfer.	1	25%

Suggested Textbooks & Resources

1. *The Art of Electronics* by Horowitz & Hill (Reference).
2. *Arduino Project Handbook* by Mark Geddes (Practical Guide).
3. *Data Acquisition Systems* by M. G. K. Rao.
4. Online Resources: Arduino Documentation, Analog Devices Tutorials, Spoken Tutorial on Circuit Simulation.

MK UNIVERSITY

PATAN, GUJARAT

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

SEMESTER-II

COURSE CODE: BTIT201

COURSE NAME: ENGINEERING MATHEMATICS-II

Course Objective

To equip computer science students with advanced mathematical tools in linear algebra, probability, and discrete mathematics that form the theoretical foundation for algorithms, machine learning, cryptography, and computational complexity. The course emphasizes computational thinking and algorithmic applications of mathematical concepts.

Course Outcomes (COs)

Upon successful completion, students will be able to:

CO1	**Apply** linear algebra concepts (eigenvalues, SVD) to problems in computer graphics, data compression, and machine learning.
CO2	**Formulate** and **solve** probability models for analyzing randomized algorithms, network reliability, and performance evaluation.
CO3	**Model** computational problems using graph theory and solve basic graph algorithms relevant to networks and data structures.
CO4	**Implement** numerical methods for solving mathematical problems computationally with error analysis awareness.

Unit	Content	Credit	Weightage
I	**Advanced Linear Algebra** • **Vector Spaces:** Definition, subspaces, linear independence, basis, dimension • **Linear Transformations:** Matrix representation, kernel, image, rank-nullity theorem • **Eigenvalues and Eigenvectors:** Computation, properties, diagonalization • **Singular Value Decomposition (SVD):** Geometric interpretation, applications in data science • **Matrix Factorization:** LU, QR decompositions (algorithmic perspective)	1	25%

II	**Probability Theory for Computer Science**
	• **Probability Basics:** Axioms, conditional probability, Bayes' theorem
	• **Random Variables:** Discrete and continuous, PMF/PDF, CDF
	• **Important Distributions:**
	  ○ Discrete: Bernoulli, Binomial, Poisson, Geometric
	  ○ Continuous: Uniform, Normal, Exponential
	• **Expectation and Variance:** Properties, moments
	• **Joint Distributions:** Covariance, correlation, independence
	• **Law of Large Numbers & Central Limit Theorem:** Conceptual understanding
		1	25%

III	**Discrete Mathematics** • **Graph Theory:** Basic terminology, types of graphs, connectivity • **Special Graphs:** Trees, bipartite graphs, planar graphs • **Graph Algorithms:** Shortest path (conceptual), graph coloring • **Combinatorics:** Counting principles, permutations, combinations • **Recurrence Relations:** Formulation, solving using characteristic equations	1	25%
IV	**Numerical Methods for CS Applications** • **Numerical Linear Algebra:** Solving linear systems (Gaussian elimination, iterative methods) • **Numerical Integration:** Trapezoidal rule, Simpson's rule • **Root Finding:** Bisection method, Newton-Raphson method • **Error Analysis:** Round-off error, truncation error, stability	1	25%

Textbooks:

1. Linear Algebra and Its Applications by Gilbert Strang (for Module 1)
2. Introduction to Probability by Dimitri P. Bertsekas and John N. Tsitsiklis (for Module 2)
3. Discrete Mathematics and Its Applications by Kenneth H. Rosen (for Module 3)

Reference books:

1. Numerical Recipes by Press et al. (for Module 4)
2. Convex Optimization by Boyd and Vandenberghe (for Module 5)
3. Mathematics for Computer Science (MIT OpenCourseWare)

Digital Resources:

- 3Blue1Brown YouTube series (linear algebra, calculus)
- Khan Academy probability and linear algebra modules
- Jupyter notebooks for computational examples

COURSE CODE: BTIT202

COURSE NAME: DATA STRUCTURES & ALGORITHMS (DSA)

Course Objective

To develop a deep understanding of fundamental and advanced data structures, their implementations, and associated algorithms. The course emphasizes analyzing time/space complexity, selecting appropriate data structures for problem-solving, and applying algorithmic design paradigms to real-world computing challenges.

Course Outcomes (COs)

Upon successful completion, students will be able to:

CO1	**Analyze** the time and space complexity of algorithms using asymptotic notations and recurrence relations.
CO2	**Implement** fundamental and advanced data structures using appropriate programming languages.
CO3	**Select** and justify appropriate data structures for solving specific computational problems.
CO4	**Design** and implement algorithms using standard paradigms (divide & conquer, greedy, dynamic programming).

Unit	Content	Credit	Weightage

I	**Foundations and Analysis** • **Algorithm Analysis:** Asymptotic notations (Big-O, Big-Ω, Big-θ), best/worst/average cases • **Complexity Analysis:** Time-space tradeoffs, amortized analysis, recurrence relations • **Abstract Data Types (ADTs):** Specification vs. implementation • **Review of C/C++/Python essentials:** Pointers, recursion, memory management	1	25%

II	**Fundamental Data Structures**
	• **Arrays:** Operations, dynamic arrays, applications
	• **Linked Lists:**
	  ○ Singly, doubly, circular linked lists
	  ○ Operations, memory representation
	  ○ Applications: polynomial representation, sparse matrices
	• **Stacks:**
	  ○ Array and linked list implementations
	  ○ Applications: Expression evaluation, recursion, backtracking
		1	25%

III	**Trees and Advanced Structures**
	• **Trees:** Terminology, binary trees, properties
	• **Binary Search Trees (BST):** Operations (insert, delete, search), traversal (in/pre/post order, level order)
	• **Balanced Trees:**
	  ○ AVL Trees: Rotations, insertion, deletion
	  ○ Red-Black Trees (concepts)
	• **Heaps:** Min-heap, max-heap, heap operations, heap sort
	• **Tries:** Structure, applications in autocomplete
	• **B-Trees:** Structure, applications in databases
		1	25%

IV	**Hashing and Dictionaries** • **Hash Tables:** Direct addressing, hash functions • **Collision Resolution:** Chaining, open addressing (linear/quadratic probing, double hashing) • **Performance Analysis:** Load factor, rehashing • **Applications:** Dictionaries, symbol tables, caching	1	25%

Textbooks:

1. Introduction to Algorithms by Cormen, Leiserson, Rivest, Stein (CLRS)
2. Data Structures and Algorithms in Python/C++ by Goodrich, Tamassia, Goldwasser
3. Algorithms by Sedgewick and Wayne

Reference Books:

1. The Algorithm Design Manual by Steven Skiena
2. Programming Pearls by Jon Bentley
3. Cracking the Coding Interview by Gayle Laakmann McDowell

Online Platforms:

- Visualization: VisuAlgo, Algorithm Visualizer
- Practice: LeetCode, HackerRank, Codeforces
- MOOCs: MIT OpenCourseWare (6.006), Coursera (Stanford Algorithms)

Tools & Software:

- Python with libraries (timeit, memory_profiler, matplotlib for plots)
- C++ STL/Java Collections Framework
- Jupyter Notebooks for interactive learning

COURSE CODE: BTIT203
COURSE NAME: DIGITAL LOGIC DESIGN (DLD)
Course Objective
To provide a comprehensive understanding of digital systems design fundamentals, from basic logic gates to complex sequential circuits and memory elements. The course aims to develop the ability to analyze, design, and implement digital circuits that form the foundation of computer architecture, embedded systems, and digital hardware.
Course Outcomes (COs)
Upon successful completion, students will be able to:

CO1	**Apply** number systems, codes, and binary arithmetic to solve digital computation problems.
CO2	**Analyze** and **simplify** Boolean functions using algebraic methods and K-maps.
CO3	**Design** and **implement** combinational circuits for arithmetic operations, data routing, and code conversion.
CO4	**Design** sequential circuits including counters, registers, and finite state machines.

Unit	Content	Credit	Weightage
I	**Number Systems and Codes** • **Number Systems:** Binary, Octal, Hexadecimal, conversions • **Complements:** 1's, 2's, 9's, 10's complements • **Binary Arithmetic:** Addition, subtraction, multiplication, division • **Binary Codes:** BCD, Gray code, Excess-3, ASCII, Unicode • **Error Detection & Correction:** Parity bits, Hamming codes	1	25%
II	**Boolean Algebra and Logic Gates** • **Boolean Algebra:** Postulates, theorems, De Morgan's laws • **Logic Gates:** AND, OR, NOT, NAND, NOR, XOR, XNOR - symbols, truth tables • **Gate-level Implementation:** SOP, POS forms • **Universal Gates:** Implementation using NAND/NOR only • **Integrated Circuits:** SSI, MSI, LSI, VLSI concepts, TTL/CMOS families	1	25%

III	**Combinational Logic Design (12 Hours)**
	• **Minimization Techniques:**
	  ○ Karnaugh Maps (2-5 variables)
	  ○ Quine-McCluskey method
	  ○ Don't care conditions
	• **Combinational Circuits:**
	  ○ Adders (Half, Full, Ripple carry, Carry look-ahead)
	  ○ Subtractors
	  ○ Comparators
	  ○ Multiplexers (MUX) and Demultiplexers (DEMUX)
		1	25%

	○ Encoders and Decoders (Binary, BCD to 7-segment) • **Code Converters:** Binary to Gray, BCD to Excess-3, etc. • **Hazards:** Static, dynamic hazards and elimination		
IV	**Sequential Logic Design** • **Latches:** SR latch, D latch, gated latches • **Flip-flops:** SR, JK, D, T, Master-slave, edge-triggered • **Flip-flop Conversions:** One type to another • **Registers:** Shift registers (SISO, SIPO, PISO, PIPO), Universal shift register • **Counters:** ○ Asynchronous (Ripple) counters ○ Synchronous counters ○ Mod-N counters, up/down counters ○ Ring counter, Johnson counter • **Finite State Machines (FSM):** Mealy and Moore models	1	25%

Textbooks:

1. Digital Design by M. Morris Mano and Michael D. Ciletti
2. Fundamentals of Digital Logic with Verilog Design by Stephen Brown and Zvonko Vranesic
3. Digital Logic and Computer Design by M. Morris Mano

Reference Books:

1. Digital Electronics by R.P. Jain
2. Verilog HDL: A Guide to Digital Design and Synthesis by Samir Palnitkar
3. Introduction to Logic Design by Alan B. Marcovitz

Online Platforms:

- Simulation Tools: Logisim, Digital Works, CircuitVerse
- HDL Simulators: EDA Playground, ModelSim (student edition)
- MOOCs: NPTEL (Digital Circuits), Coursera (Digital Systems)

Laboratory Component

Tools: Digital Trainer Kits, ICs, Breadboards, Multimeters, Simulation Software (Logisim/Proteus/Quartus), Verilog/VHDL compiler

List of Experiments:

1. **Lab 1:** Verification of truth tables for logic gates
2. **Lab 2:** Implementation of Boolean functions using logic gates
3. **Lab 3:** Design and implementation of adders/subtractors
4. **Lab 4:** Implementation of code converters
5. **Lab 5:** Design using multiplexers and demultiplexers
6. **Lab 6:** Flip-flop characteristics and conversions
7. **Lab 7:** Design of synchronous/asynchronous counters
8. **Lab 8:** Shift register applications
9. **Lab 9:** Finite State Machine implementation
10. **Lab 10:** HDL programming: Basic gate implementation
11. **Lab 11:** HDL programming: Combinational circuit (adder/MUX)
12. **Lab 12:** Mini-project: Digital clock/calculator/traffic light controller

MK UNIVERSITY

PATAN, GUJARAT

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

COURSE CODE: BTIT204

COURSE NAME: ENGINEERING GRAPHICS & DESIGN

Course Objective

1. To develop the ability to communicate design ideas using standard engineering drawing practices.
2. To introduce computer-aided design (CAD) tools for creating and modifying technical drawings.
3. To foster spatial visualization and reasoning skills.
4. To relate engineering graphics principles to computer science applications such as GUI design, game environments, 3D modeling, and hardware schematics.

Course Outcomes (COs)

Upon successful completion, students will be able to:

CO1	Apply standard engineering drawing conventions to create and interpret technical drawings.
CO2	Generate orthographic, isometric, and sectional views of objects using manual and CAD tools.
CO3	Use CAD software to produce 2D drafts and basic 3D models relevant to CS applications.
CO4	Visualize and represent objects in 3D space, improving spatial reasoning skills.

Unit	Content	Credit	Weightage
I	**Fundamentals of Engineering Drawing** • Introduction to drawing instruments, standards (BIS/ISO) • Lettering, line types, dimensioning • Geometrical constructions	1	25%
II	**Orthographic Projections** • Principles of first-angle and third-angle projection • Projection of points, lines, planes, and solids • Sectional views	1	25%
III	**Isometric and Pictorial Views** • Isometric projection of simple solids • Perspective basics	1	25%
IV	**Computer-Aided Design (CAD)** • Introduction to CAD software (e.g., AutoCAD, Fusion 360, SketchUp) • Basic 2D drafting and 3D modeling • Editing and dimensioning in CAD	1	25%

Textbooks:

- "Engineering Drawing" by N.D. Bhatt (Charotar Publishing)
- "Engineering Drawing" by P.S. Gill (SK Kataria & Sons)
- "Engineering Drawing with CAD" by M.B. Shah & B.C. Rana (Pearson)

Reference Books:

- "Technical Drawing" by Giesecke, Mitchell, Spencer, Hill, Dygdon, Novak, Lockhart (Prentice Hall)
- "Engineering Drawing and Design" by Cecil Jensen, Jay Helsel, Dennis Short (McGraw-Hill)
- "Fundamentals of Engineering Drawing" by Warren J. Luzadder & Jon M. Duff (Prentice Hall)

Digital Libraries

- Engineering Design Graphics Journal (EDGJ)
- CAD Society resources
- IEEE Computer Graphics and Applications

COURSE CODE: BTIT205
COURSE NAME: MANAGEMENT
Course Objective

1. To understand basic management principles in technology organizations
2. To develop skills in software project management and team leadership
3. To analyze business strategies for technology products/services
4. To comprehend financial, legal, and ethical aspects of tech businesses

Course Outcomes (COs)
Upon successful completion, students will be able to:

CO1	Apply management principles to software project planning and execution
CO2	Lead and motivate technical teams effectively using appropriate leadership
CO3	Analyze technology business strategies and make informed decisions
CO4	Develop basic financial plans and budgets for IT projects

Unit	Content	Credit	Weightage
I	**Introduction to Management in Technology** • **1.1** Evolution of management thought • **1.2** Functions of management (Planning, Organizing, Leading, Controlling) • **1.3** Types of technology organizations: product vs service, startup vs enterprise • **1.4** Role of engineers in management • **1.5** Technical vs managerial career paths	1	25%
II	**Project Management for Software** • **2.1** Software Development Life Cycle (SDLC) models • **2.2** Agile methodologies (Scrum, Kanban, XP) • **2.3** Project planning, scheduling, and estimation techniques • **2.4** Risk management in software projects • **2.5** Tools: Jira, Trello, Asana, Microsoft Project	1	25%
III	**Team Management and Leadership** • **3.1** Team formation and dynamics (Tuckman's model) • **3.2** Motivation theories (Maslow, Herzberg, McClelland) • **3.3** Leadership styles for tech teams • **3.4** Conflict resolution and negotiation	1	25%

	• **3.5** Diversity and inclusion in tech workplaces		
IV	**Technology Strategy and Innovation** • **4.1** Technology adoption lifecycle (Crossing the Chasm) • **4.2** Competitive analysis and SWOT in tech • **4.3** Innovation management and R&D • **4.4** Intellectual property in software (patents, copyrights, open source) • **4.5** Digital transformation strategies	1	25%

Textbooks:

1. "The Mythical Man-Month" by Frederick P. Brooks Jr.
2. "Peopleware: Productive Projects and Teams" by Tom DeMarco & Timothy Lister
3. "Software Project Management" by Bob Hughes & Mike Cotterell
4. "The Lean Startup" by Eric Ries
5. "Crossing the Chasm" by Geoffrey A. Moore

Reference Books:

1. "The Phoenix Project" by Gene Kim, Kevin Behr, George Spafford
2. "Project Management for Engineering, Business and Technology" by John M. Nicholas & Herman Steyn
3. "Engineering Management: Challenges in the New Millennium" by William B. Rouse
4. "Managing the Unmanageable: Rules, Tools, and Insights for Managing Software People and Teams" by Mickey W. Mantle & Ron Lichty

Online Resources:

1. Harvard Business Review (Tech Management section)
2. MIT Sloan Management Review
3. Project Management Institute (PMI) resources
4. Agile Alliance materials
5. Case studies from: Stanford, IIMs, ISB

MK UNIVERSITY

PATAN, GUJARAT

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

SEMESTER-III

COURSE CODE: BTIT301
COURSE NAME: DISCRETE MATHAMATICS
Course Objective

1. To develop mathematical reasoning and proof-writing skills
2. To understand and apply discrete structures in CS contexts
3. To master formal logic and its applications in computing
4. To analyze counting problems and probability in algorithms

Course Outcomes (COs)
Upon successful completion, students will be able to:

CO1	Construct formal proofs using various proof techniques
CO2	Apply logic to formulate and solve computational problems
CO3	Solve counting problems using combinatorial methods
CO4	Model relationships and networks using graph theory

Unit	Content	Credit	Weightage
I	**Foundations - Logic and Proofs** • **1.1 Propositional Logic** ○ Propositions, connectives, truth tables ○ Logical equivalences, tautologies, contradictions ○ Normal forms (DNF, CNF) • **1.2 Predicate Logic** ○ Quantifiers (∀, ∃), nested quantifiers ○ Rules of inference **1.3 Proof Techniques** ○ Direct, contrapositive, contradiction ○ Mathematical induction, strong induction ○ Structural induction (on trees, expressions) ○ Proof by cases, counterexamples	1	25%
II	**Sets, Relations, and Functions** • **2.1 Set Theory** ○ Operations, laws, cardinality ○ Power sets, Cartesian products ○ Infinite sets, countable/uncountable sets (Cantor's diagonalization) • **2.2 Relations** ○ Properties (reflexive, symmetric, transitive, equivalence) ○ Equivalence relations and partitions ○ Partial orders, Hasse diagrams, lattices • **2.3 Functions** ○ Injections, surjections, bijections ○ Inverse functions, composition ○ Floor, ceiling, modulo functions ○ Growth of functions (Big-O, Ω, Θ notation)	1	25%

III	**Combinatorics and Discrete Probability** • **3.1 Counting Principles** ○ Sum and product rules ○ Inclusion-exclusion principle ○ Pigeonhole principle and applications • **3.2 Permutations and Combinations** ○ With/without repetition ○ Binomial coefficients, Pascal's triangle ○ Combinatorial identities **3.3 Advanced Counting** ○ Recurrence relations ○ Generating functions ○ Solving recurrences (characteristic equations) ○ **3.4 Discrete Probability** ○ Finite probability spaces ○ Conditional probability, Bayes' theorem ○ Random variables, expectation, variance ○ Probabilistic method applications	1	25%
IV	**Graph Theory** • **4.1 Basic Concepts** ○ Graphs, multigraphs, pseudographs ○ Directed graphs, weighted graphs ○ Paths, cycles, connectivity • **4.2 Special Graphs** ○ Complete, bipartite, regular graphs ○ Trees and their properties ○ Planar graphs, Euler's formula ○ Graph coloring, chromatic number ○ **4.3 Graph Algorithms** ○ Graph representation (adjacency matrix/list) ○ Traversal (DFS, BFS) ○ Shortest path (Dijkstra's algorithm) ○ Minimum spanning trees (Prim's, Kruskal's) ○ Matching problems (maximum bipartite matching)	1	25%

Textbooks:

- "Discrete Mathematics and Its Applications" by Kenneth H. Rosen (7th/8th edition)
- "Concrete Mathematics" by Graham, Knuth, Patashnik
- "Discrete Mathematics for Computer Scientists" by Stein, Drysdale, Bogart
- "Mathematics for Computer Science" by Eric Lehman, F.T. Leighton, A.R. Meyer
- "Discrete Mathematics with Graph Theory" by Goodaire & Parmenter

Reference Books:

- "Introduction to Algorithms" by Cormen et al. (Appendix on mathematics)
- "The Art of Computer Programming, Vol. 1" by Donald Knuth
- "A Textbook of Discrete Mathematics" by Swapan Kumar Sarkar

Online Resources:

- MIT OpenCourseWare: 6.042 Mathematics for Computer Science
- Coursera: Discrete Mathematics Specialization (UC San Diego)
- NPTEL: Discrete Mathematics courses

MK UNIVERSITY

PATAN, GUJARAT

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

COURSE CODE: BTIT302
COURSE NAME: OBJECT ORIENTED PROGRAMMING (OOP)

Course Objective

1. To understand object-oriented programming principles and their advantages
2. To master C++ syntax, semantics, and standard library
3. To develop skills in designing and implementing OOP solutions
4. To understand memory management and performance considerations in C++

Course Outcomes (COs)

Upon successful completion, students will be able to:

CO1	Design and implement classes using OOP principles
CO2	Apply inheritance and polymorphism to create extensible software
CO3	Utilize templates and STL for generic programming
CO4	Implement proper memory management and exception handling

Unit	Content	Credit	Weightage
I	**C++ Fundamentals and Review of C Concepts** • **1.1 Introduction to C++** o History, standards (C++98, C++11, C++14, C++17, C++20) o Compilation process (g++, clang++) o Basic program structure • **1.2 C Review in C++ Context** o Data types, operators, control structures o Functions, pointers, arrays o Structures and unions • **1.3 C++ Enhancements over C** o Input/output streams (cin, cout) o References vs pointers o Function overloading o Default arguments, inline functions o Namespaces	1	25%
II	**Introduction to Object-Oriented Programming** • **2.1 OOP Concepts** o Abstraction, encapsulation, inheritance, polymorphism o Classes vs objects o Benefits of OOP • **2.2 Classes and Objects** o Class definition, access specifiers (public, private, protected) o Member functions, constructors, destructors o Static members, friend functions/classes o this pointer • **2.3 Object Lifecycle** o Constructors (default, parameterized, copy)	1	25%

	○ Destructors ○ Dynamic object creation (new, delete) ○ Object arrays		
III	**Advanced Class Features and Memory Management** • **3.1 Operator Overloading** ○ Unary and binary operators ○ Friend operators ○ Overloading I/O operators (<<, >>) ○ Type conversion operators • **3.2 Memory Management** ○ Stack vs heap allocation ○ Dynamic arrays, 2D arrays ○ Memory leaks and debugging tools (Valgrind) ○ RAII (Resource Acquisition Is Initialization) • **3.3 Advanced Class Design** ○ Constant objects and member functions ○ Mutable members ○ Nested classes, local classes	1	25%
IV	**Inheritance and Polymorphism** • **4.1 Inheritance Basics** ○ Base and derived classes ○ Access control (public, private, protected inheritance) ○ Constructors and destructors in inheritance ○ Function overriding • **4.2 Types of Inheritance** ○ Single, multiple, multilevel, hierarchical, hybrid ○ Virtual base classes ○ Diamond problem and solution • **4.3 Polymorphism** ○ Virtual functions, pure virtual functions ○ Abstract classes and interfaces ○ Virtual destructors ○ Late binding vs early binding ○ vtable and vpointer concepts	1	25%

Textbooks:

- "The C++ Programming Language" by Bjarne Stroustrup (4th edition)
- "Object-Oriented Programming in C++" by Robert Lafore
- "C++ Primer" by Stanley B. Lippman, Josée Lajoie, Barbara E. Moo

Reference Books:

- "Effective C++" and "Effective Modern C++" by Scott Meyers
- "Design Patterns: Elements of Reusable Object-Oriented Software" by GoF
- "C++ Concurrency in Action" by Anthony Williams
- "Programming: Principles and Practice Using C++" by Bjarne Stroustrup
- Data Structures and Algorithms in C++" by Michael T. Goodrich et al.

Online Resources:

- **cppreference.com** (Official C++ reference)
- **learncpp.com** (Free tutorials)
- **C++ Core Guidelines**
- **ISO C++ Standard documents**
- **Stack Overflow C++ tag**
- **YouTube:** CppCon, Meeting C++ conferences

Laboratory Experiments

Lab 1: Basic C++ programs with classes and objects
Lab 2: Operator overloading and friend functions
Lab 3: Inheritance and polymorphism implementations
Lab 4: File handling and exception handling
Lab 5: Template programming
Lab 6: STL containers and algorithms
Lab 7: Smart pointers and modern C++ features
Lab 8: Multithreading basics
Lab 9-10: Mini-project implementing design patterns
Lab 11-12: Comprehensive project (Library Management, Banking System, etc.)

COURSE CODE: BTIT303
COURSE NAME: COMPUTER ORGANIZATION & ARCHITECTURE (COA)
Course Objective

This course introduces the fundamental concepts of computer organization and architecture, covering the design and operation of digital computers from gate-level to system-level. It explores processor design, memory hierarchy, I/O systems, and parallel processing, providing the foundation for understanding modern computer systems and their performance characteristics.

Course Outcomes (COs)

Upon successful completion, students will be able to:

CO1	Analyze computer performance using various metrics
CO2	Design basic arithmetic and logic units
CO3	Explain processor data path and control unit operations
CO4	Analyze memory hierarchy and its impact on performance

Unit	Content	Credit	Weightage

I	**Introduction and Basic Computer Structure**
	• **1.1 Evolution of Computers**
	  ○ Generations of computers (Vacuum tubes to VLSI)
	  ○ Moore's Law and its implications
	  ○ Types of computers: embedded, desktop, servers, supercomputers
	• **1.2 Computer Components and Functions**
	  ○ Von Neumann architecture
	  ○ Harvard architecture
	  ○ CPU, memory, I/O subsystems
	• **1.3 Performance Metrics**
		1	25%

	○ Clock rate, CPI, MIPS, MFLOPS ○ Amdahl's Law ○ Benchmarking and performance evaluation		
II	**Data Representation and Computer Arithmetic** • **2.1 Number Systems** ○ Binary, octal, hexadecimal ○ Integer representation: signed magnitude, 1's complement, 2's complement ○ Floating-point representation: IEEE 754 standard • **2.2 Computer Arithmetic** ○ Addition/subtraction of signed numbers ○ Multiplication algorithms: Booth's algorithm ○ Division algorithms: Restoring and non-restoring ○ Floating-point arithmetic operations	1	25%
III	**Processor Design and Instruction Set Architecture** • **3.1 Instruction Set Architecture (ISA)** ○ CISC vs RISC philosophies ○ Addressing modes (immediate, direct, indirect, indexed, relative) ○ Instruction formats (zero, one, two, three address) • **3.2 MIPS Architecture (Case Study)** ○ Register set and memory organization ○ Instruction types: R-type, I-type, J-type ○ Sample MIPS programs • **3.3 CPU Organization** ○ Register organization ○ ALU design ○ Control unit: Hardwired vs microprogrammed control	1	25%
IV	**Processor Datapath and Control** • **4.1 Single-Cycle Implementation** ○ Datapath components (PC, registers, ALU, memory) ○ Control signals generation ○ Performance limitations • **4.2 Multi-Cycle Implementation** ○ Breaking instructions into multiple cycles ○ Finite state machine control • **4.3 Pipelining** ○ Basic 5-stage pipeline (IF, ID, EX, MEM, WB) ○ Pipeline hazards: structural, data, control ○ Hazard detection and resolution ○ Forwarding (bypassing) and stalling	1	25%

Textbooks:

- "Computer Organization and Design: The Hardware/Software Interface" by David A. Patterson and

John L. Hennessy (MIPS/RISC-V editions)
- "Computer Architecture: A Quantitative Approach" by John L. Hennessy and David A. Patterson
- "Computer Organization and Architecture" by William Stallings

Reference Books:

- "Structured Computer Organization" by Andrew S. Tanenbaum
- "Digital Design and Computer Architecture" by David Harris and Sarah Harris
- "Computer Systems: A Programmer's Perspective" by Bryant and O'Hallaron
- "Introduction to High Performance Computing for Scientists and Engineers" by Georg Hager and Gerhard Wellein

Online Resources:

1. NPTEL Courses: Computer Organization by Prof. S. Raman (IIT Madras)
2. MIT OpenCourseWare: Computation Structures
3. UC Berkeley CS61C: Great Ideas in Computer Architecture
4. Coursera: Computer Architecture by Princeton University
5. Wikipedia: CPU design articles
6. ARM Developer Documentation
7. RISC-V Specifications

COURSE CODE: BTIT304
COURSE NAME: DATA BASE MANAGEMENT SYSTEM (DBMS)
Course Objective
This course introduces fundamental concepts of database systems, covering data modeling, database design, SQL programming, transaction management, and system architecture. Emphasis is placed on relational database theory, normalization, and practical implementation using modern DBMS technologies. The course prepares students to design, implement, and manage database systems for real-world applications.

Course Outcomes (COs)
Upon successful completion, students will be able to:

CO1	Design database schemas using ER modeling and normalization
CO2	Implement and query databases using SQL and PL/SQL
CO3	Apply transaction management and concurrency control
CO4	Design efficient storage structures and indexing schemes

Unit	Content	Credit	Weightage

I	**Introduction to Database Systems**
	• **1.1 Database System Concepts**
	  ○ Data, information, and knowledge
	  ○ File systems vs database systems
	  ○ Three-schema architecture (internal, conceptual, external)
	  ○ Data independence (logical and physical)
	• **1.2 Database System Architecture**
		1	25%

	○ DBMS components: Query processor, storage manager ○ Database users and administrators ○ Database languages: DDL, DML, DCL • **1.3 Database Applications** ○ Traditional applications (banking, airlines, universities) ○ Modern applications (e-commerce, social media, IoT) ○ Emerging trends (big data, cloud databases)		
II	**Data Models and Database Design** • **2.1 Entity-Relationship Model** ○ Entities, attributes, relationships ○ ER diagram notation (Chen notation, Crow's foot) ○ Cardinality constraints (1:1, 1:N, M:N) ○ Weak entities, composite attributes, multivalued attributes • **2.2 Enhanced ER Modelling** ○ Specialization and generalization ○ Aggregation ○ Inheritance in EER • **2.3 Relational Model** ○ Relations, tuples, attributes, domains ○ Keys: Super, candidate, primary, foreign ○ Relational algebra operations ▪ Basic: select, project, union, set difference, Cartesian product ▪ Additional: rename, intersection, natural join, division ○ Relational calculus (tuple and domain)	1	25%
III	**Structured Query Language (SQL)** • **3.1 SQL Fundamentals** ○ Data types, schema definition ○ Basic queries: SELECT, FROM, WHERE ○ Aggregate functions, GROUP BY, HAVING ○ Set operations: UNION, INTERSECT, EXCEPT • **3.2 Advanced SQL** ○ Nested subqueries (correlated and non-correlated) ○ JOIN operations: INNER, LEFT, RIGHT, FULL OUTER ○ Views: creation, updating, materialized views ○ Integrity constraints: NOT NULL, UNIQUE, CHECK, DEFAULT • **3.3 SQL Programming** ○ PL/SQL or T-SQL basics ○ Stored procedures, functions, triggers ○ Cursors, exception handling	1	25%

	○ Embedded SQL, dynamic SQL		
IV	**Database Design Theory** • **4.1 Functional Dependencies** ○ Definition and properties ○ Armstrong's axioms ○ Closure of attribute sets ○ Canonical cover • **4.2 Normalization** ○ First Normal Form (1NF) ○ Second Normal Form (2NF) ○ Third Normal Form (3NF) ○ Boyce-Codd Normal Form (BCNF) ○ Higher normal forms (4NF, 5NF) overview • **4.3 Decomposition Algorithms** ○ Lossless join decomposition ○ Dependency preserving decomposition ○ Synthesis algorithm for 3NF ○ Decomposition algorithm for BCNF	1	25%

Textbooks:

- "Database System Concepts" by Abraham Silberschatz, Henry F. Korth, and S. Sudarshan
- "Fundamentals of Database Systems" by Ramez Elmasri and Shamkant B. Navathe
- "Database Management Systems" by Raghu Ramakrishnan and Johannes Gehrke

Reference Books:

- "SQL and Relational Theory: How to Write Accurate SQL Code" by C.J. Date
- "Transaction Processing: Concepts and Techniques" by Jim Gray and Andreas Reuter
- "NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence" by Pramod J. Sadalage and Martin Fowler
- "Data Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems" by Martin Kleppmann

Online Resources:

- Stanford Online: Databases course by Jennifer Widom
- Coursera: Database series by University of Michigan
- NPTEL: Database Management Systems courses
- PostgreSQL Documentation
- MySQL Reference Manual
- MongoDB University (free courses)
- db-fiddle.com (online SQL playground)

Laboratory Experiments

- **Lab 1:** Installation and configuration of DBMS (MySQL/PostgreSQL)
- **Lab 2:** Basic SQL – DDL, DML commands
- **Lab 3:** SQL queries with aggregate functions and grouping
- **Lab 4:** Advanced SQL – Subqueries, joins, views
- **Lab 5:** PL/SQL programming – Stored procedures and functions

- **Lab 6:** Triggers and cursors implementation
- **Lab 7:** ER modeling and conversion to relational schema
- **Lab 8:** Normalization exercises
- **Lab 9:** Index creation and performance analysis
- **Lab 10:** Transaction management and concurrency control
- **Lab 11:** NoSQL database basics (MongoDB)
- **Lab 12:** Mini-project – Complete database application

COURSE CODE: BTIT305
COURSE NAME: ECONOMICS FOR ENGINEERS
Course Objective
This course introduces fundamental economic principles and decision-making frameworks specifically tailored for engineering students. It covers microeconomics, macroeconomics, engineering economics, and technology economics, with special emphasis on applications in the technology sector. The course equips future engineers with economic thinking skills essential for project evaluation, business decisions, and technology strategy in engineering careers.

Course Outcomes (COs)
Upon successful completion, students will be able to:

CO1	Apply microeconomic principles to analyze technology markets
CO2	Perform engineering economic analysis for project evaluation
CO3	Calculate and interpret key financial metrics for tech projects
CO4	Analyze macroeconomic factors affecting technology industry

Unit	Content	Credit	Weightage
I	**Introduction to Economics for Engineers** • **1.1 Why Economics for Engineers?** ○ Economic decision-making in engineering projects ○ Cost-benefit analysis in technology development ○ Role of engineers in business and economic growth • **1.2 Basic Economic Concepts** ○ Scarcity, choice, and opportunity cost ○ Production possibilities frontier ○ Economic systems: market, command, mixed ○ Circular flow of economic activity • **1.3 Engineering as Economic Activity** ○ Value creation through technology ○ Engineering efficiency vs economic efficiency ○ Time value of money in engineering projects	1	25%

II	**Microeconomics for Technology Markets**
	• **2.1 Demand and Supply Analysis**
	  ○ Market forces in technology products
	  ○ Elasticity: price, income, cross elasticity
	  ○ Technology adoption curves
	• **2.2 Consumer Behaviour**
	  ○ Utility theory and indifference curves
	  ○ Budget constraints
	  ○ Consumer surplus in technology markets
	• **2.3 Production and Costs**
	  ○ Production functions (short run vs long run)
	  ○ Cost concepts: fixed, variable, marginal, average
	  ○ Economies of scale in tech industry
	  ○ Learning curves in software development
	• **2.4 Market Structures**
	  ○ Perfect competition
	  ○ Monopoly and natural monopolies in tech
	  ○ Oligopoly (telecom, operating systems)
	• Monopolistic competition (app stores, SaaS)
	• Network effects and winner-take-all markets
		1	25%

III	**Engineering Economics**
	• **3.1 Time Value of Money**
	  ○ Present value, future value calculations
	  ○ Discounting and compounding
	  ○ Net Present Value (NPV) method
	  ○ Internal Rate of Return (IRR)
	• **3.2 Investment Appraisal Techniques**
	  ○ Payback period
	  ○ Accounting Rate of Return (ARR)
	  ○ Benefit-Cost Ratio (BCR)
	  ○ Break-even analysis for tech products
	• **3.3 Cost Analysis for Engineering Projects**
	  ○ Life Cycle Costing (LCC)
	  ○ Capital vs operational expenditures (CapEx vs OpEx)
	  ○ Depreciation methods (straight-line, declining balance)
	  ○ Total Cost of Ownership (TCO) for IT systems
		1	25%

IV	**Macroeconomics for Engineers**
	• **4.1 National Income Accounting**
	  ○ GDP, GNP, NNP concepts
	  ○ Economic growth and development
	  ○ Technology's contribution to GDP
		1	25%

	• **4.2 Money, Banking, and Inflation** ○ Monetary policy and interest rates ○ Inflation's impact on engineering projects ○ Cryptocurrencies and digital payments • **4.3 Business Cycles** ○ Economic fluctuations and technology investment ○ Recessions and tech industry resilience ○ Leading indicators for tech sector		

Textbooks:

- "Engineering Economy" by William G. Sullivan, Elin M. Wicks, and C. Patrick Koelling
- "Principles of Economics" by N. Gregory Mankiw
- "Managerial Economics and Business Strategy" by Michael Baye and Jeff Prince

Reference Books:

- "Economics for Engineers" by T. T. Sethi and A. K. Nath
- "Financial Management for Engineers" by C. M. Chang
- "The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation" by Eric Ries

Online Resources:

- Khan Academy: Microeconomics and Macroeconomics courses
- Coursera: Economics for Engineers courses
- Harvard Business Review: Case studies on technology economics
- St. Louis Fed FRED: Economic data for analysis
- TechCrunch, Wired: Current technology economics articles
- Bureau of Economic Analysis (BEA) data
- World Bank Open Data

COURSE CODE: BTIT306
COURSE NAME: SOFTWARE GIT
Course Objective

This practical course introduces version control systems with a focus on Git, covering fundamental concepts, workflows, and best practices for collaborative software development. Students will learn to manage code repositories, collaborate effectively, integrate with modern development tools, and understand industry-standard Git workflows used in professional software engineering.

Course Outcomes (COs)

Upon successful completion, students will be able to:

CO1	Set up and configure Git for individual and team development
CO2	Implement branching strategies and resolve merge conflicts
CO3	Collaborate effectively using remote repositories and pull requests

Unit	Content	Credit	Weightage

MK UNIVERSITY

PATAN, GUJARAT

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

I	**Introduction to Version Control** • **1.1 Why Version Control?** ○ The problem of code collaboration ○ History of version control: Local → Centralized → Distributed ○ Benefits: Collaboration, backup, experimentation, accountability • **1.2 Version Control Systems Landscape** ○ CVS, SVN (Centralized) ○ Git, Mercurial (Distributed) ○ Comparison of different VCS • **1.3 Real-world Applications** ○ Open-source projects (Linux kernel, Android) ○ Enterprise development workflows ○ Academic and research code management		
II	**Git Fundamentals** • **2.1 Git Architecture** ○ Three states: Working directory, staging area, repository ○ Three main sections: .git directory, staging area, working tree ○ Snapshots, not differences • **2.2 Getting Started with Git** ○ Installation and configuration (git config) ○ Setting up identity (name, email) ○ Global vs local configuration ○ Initializing repositories (git init, git clone) • **2.3 Basic Git Commands** ○ git status - Checking repository state ○ git add - Staging changes ○ git commit - Committing changes ○ git log - Viewing history ○ git diff - Comparing changes		
III	**Branching and Merging** • **3.1 Understanding Branches** ○ What are branches? (Lightweight movable pointers) ○ HEAD pointer concept ○ Creating and switching branches (git branch, git checkout, git switch) • **3.2 Branch Operations** ○ Listing branches ○ Creating feature branches ○ Deleting branches		

	○ Remote branch tracking • **3.3 Merging Strategies** ○ Fast-forward merges ○ Three-way merges ○ Merge conflicts: identification and resolution ○ git merge vs git rebase • **3.4 Advanced Branching** ○ Rebasing: cleaning up commit history ○ Interactive rebase (git rebase -i) ○ Cherry-picking commits ○ Stashing changes (git stash)		

Textbooks:

- "Pro Git" by Scott Chacon and Ben Straub (Free online at git-scm.com/book)
- "Version Control with Git" by Jon Loeliger and Matthew McCullough

Reference Books:

- "Git Pocket Guide" by Richard E. Silverman
- "Learning Git" by Anna Skoulikari
- "Git for Teams" by Emma Jane Hogbin Westby

Online Resources:

- Official Git Documentation: git-scm.com/doc
- GitHub Guides: guides.github.com
- GitLab Documentation: docs.gitlab.com
- Atlassian Git Tutorials: atlassian.com/git
- Git Immersion Tutorial: gitimmersion.com
- Learn Git Branching: learngitbranching.js.org (Interactive)
- Oh My Git!: ohmygit.org (Game for learning Git)

MK UNIVERSITY

PATAN, GUJARAT

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

SEMESTER -IV

COURSE CODE: BTIT401
COURSE NAME: OPERATING SYSTEM

Course Objective

This course provides a comprehensive understanding of operating system concepts, design principles, and implementation techniques. It covers process management, memory management, storage systems, and system security, focusing on practical applications and modern OS design for computer science engineering.

Course Outcomes (COs)

Upon successful completion, students will be able to:

CO1	Analyze and implement process management and schedulingalgorithms
CO2	Design memory management schemes including paging and virtual memory
CO3	Implement file systems and storage management techniques
CO4	Apply security and protection mechanisms in operating systems

Unit	Content	Credit	Weightage
I	Introduction and Process Management 1.1 Introduction to Operating Systems 1.2 Process Management 1.3 CPU Scheduling 1.4 Process Synchronization	1	25%
II	Memory Management 2.1 Memory Management Basics 2.2 Paging 2.3 Segmentation 2.4 Virtual Memory	1	25%
III	Storage Management and File Systems 3.1 Mass-Storage Structure 3.2 File System Interface 3.3 File System Implementation, 3.4 I/O Systems	1	25%

IV	**System Security**
	
	• **Security Problems**
	  ○ Program threats: Trojan horse, Trap door, Logic bomb, Stack and buffer overflow
	  ○ System and network threats: Worms, Port scanning, DoS
	• **Security Measures**
	  ○ Authentication: Passwords, Physical identification, Biometrics
	  ○ One-time passwords
	  ○ Cryptography: Symmetric and Asymmetric encryption
	  ○ Digital signatures, Certificates
	  ○ Firewalls, Intrusion detection systems
	
	4.3 Virtual Machines
	
	• **Virtual Machine Concepts**
	  ○ Benefits of virtualization
	  ○ Implementation methods
	  ○ Examples: VMware, Java Virtual Machine
	• **Containerization**
	  ○ Docker architecture
	  ○ Containers vs Virtual Machines
	  ○ Container orchestration basics (Kubernetes overview)
	
	4.4 Distributed Systems
	
	• **Distributed System Structure**
	  ○ Types of distributed OS
	  ○ Network structure, Communication protocols
	• **Distributed File Systems**
	  ○ NFS architecture
	  ○ AFS design
	• **Distributed Coordination**
	  ○ Event ordering, Mutual exclusion
	  ○ Atomicity, Concurrency control
	  ○ Deadlock handling in distributed systems
	
	4.5 Case Studies and Modern OS Features
	
	Linux Case Study, Windows Case Study
		1	25%

Textbooks:

- "Operating System Concepts" by Abraham Silberschatz, Peter Baer Galvin, and Greg Gagne (10th edition)
- "Modern Operating Systems" by Andrew S. Tanenbaum (4th edition)

Reference Books:

- "Operating Systems: Three Easy Pieces" by Remzi H. Arpaci-Dusseau and Andrea C. Arpaci-Dusseau (Free online)
- "The Design of the UNIX Operating System" by Maurice J. Bach
- "Windows Internals" by Mark Russinovich, David Solomon, and Alex Ionescu

Online Resources:

- NPTEL: Operating Systems course by Prof. Chester Rebeiro (IIT Madras)
- MIT OpenCourseWare: Operating System Engineering
- Linux Kernel Documentation: kernel.org/doc/html/latest/
- OSDev Wiki: osdev.org
- GeeksforGeeks OS Section

COURSE CODE: BTIT402
COURSE NAME: THEORY OF COMPUTATION
Course Objective
This course introduces the mathematical foundations of computation, covering automata theory, formal languages, computability, and complexity theory. It provides the theoretical basis for understanding what can be computed, how efficiently it can be computed, and the fundamental limits of computation.

Course Outcomes (COs)
Upon successful completion, students will be able to:

CO1	Design and analyze finite automata and regular expressions
CO2	Construct context-free grammars and pushdown automata for formal languages
CO3	Prove language properties using pumping lemmas and closure properties
CO4	Design Turing machines and understand computability limits

Unit	Content	Credit	Weightage
I	Automata Theory and Regular Languages 1.1 Introduction to Theory of Computation 1.2 Finite Automata 1.3 Regular Expressions and Languages 1.4 Regular Language Properties	1	25%
II	Context-Free Languages and Pushdown Automata 2.1 Context-Free Grammars 2.2 Pushdown Automata 2.3 Properties of Context-Free Languages 2.4 Applications of CFGs and PDAs	1	25%

III	Turing Machines and Computability Theory 3.1 Turing Machines 3.2 Decidability and Recursive Languages 3.3 Reducibility 3.4 Advanced Computability Topics	1	25%
IV	Complexity Theory 4.1 Time and Space Complexity 4.2 NP-Completeness 4.3 Advanced Complexity Classes 4.4 Current Frontiers and Applications	1	25%

Textbooks:

- "Introduction to the Theory of Computation" by Michael Sipser (3rd edition)
- "Automata Theory, Languages, and Computation" by John E. Hopcroft, Rajeev Motwani, and Jeffrey D. Ullman (3rd edition)

Reference Books:

- "Elements of the Theory of Computation" by Harry R. Lewis and Christos H. Papadimitriou
- "Computational Complexity: A Modern Approach" by Sanjeev Arora and Boaz Barak
- "Computability, Complexity, and Languages" by Davis, Sigal, and Weyuker

Online Resources:

- NPTEL: Theory of Computation by Prof. Somenath Biswas (IIT Kanpur)
- MIT OpenCourseWare: Automata, Computability, and Complexity
- Stanford Online: Automata and Complexity Theory
- Pumping Lemma Visualizations: Multiple online tools
- Turing Machine Simulators: Online and downloadable

COURSE CODE: BTIT403
COURSE NAME: MICROPROCESSORS AND MICROCONTROLLERS
Course Objective
This course introduces the architecture, programming, and interfacing of microprocessors and microcontrollers with emphasis on embedded systems development. It covers 8086 microprocessor and 8051 microcontroller architectures, assembly programming, interfacing techniques, and embedded system design principles for computer science applications.

Course Outcomes (COs)
Upon successful completion, students will be able to:

CO1	Analyze and program 8086 microprocessor using assembly language
CO2	Design interfacing circuits for 8086 with memory and peripherals
CO3	Program 8051 microcontroller using both assembly and embedded
CO4	Interface sensors, actuators, and displays with 8051 microcontroller

Unit	Content	Credit	Weightage
I	8086 Microprocessor Architecture and Programming 1.1 Introduction to Microprocessors 1.2 8086 Microprocessor Architecture 1.3 8086 Assembly Language Programming 1.4 8086 Interrupts and DMA	1	25%
II	8086 Interfacing and System Design 2.1 Memory Interfacing 2.2 I/O Interfacing 2.3 Interfacing with Peripherals 2.4 System Design Examples	1	25%
III	8051 Microcontroller Architecture and Programming 3.1 Introduction to Microcontrollers 3.2 8051 Architecture 3.3 8051 Instruction Set and Programming 3.4 C Programming for 8051	1	25%
IV	8051 Interfacing and Embedded Systems 4.1 8051 Interfacing Techniques 4.2 Interrupt System 4.3 Advanced Interfacing 4.4 Embedded Systems Design	1	25%

Textbooks:

- "The 8086 Microprocessor: Programming and Interfacing the PC" by Kenneth J. Ayala
- "The 8051 Microcontroller and Embedded Systems" by Muhammad Ali Mazidi, Janice Gillispie Mazidi, and Rolin D. McKinlay

Reference Books:

- "Advanced Microprocessors and Peripherals" by A.K. Ray and K.M. Bhurchandi
- "Microprocessor Architecture, Programming, and Applications with the 8085" by Ramesh S. Gaonkar
- "Embedded C Programming and the Microchip PIC" by Richard H. Barnett
- "Making Embedded Systems: Design Patterns for Great Software" by Elecia White

Online Resources:

- NPTEL: Microprocessors and Microcontrollers courses
- 8051.com - Tutorials and resources
- Keil Development Tools: keil.com
- Proteus Simulation Software: labcenter.com
- Arduino Platform: arduino.cc (for modern microcontroller concepts)

Laboratory Experiments

8086 Experiments:

1. **Lab 1:** Assembly language programs for arithmetic operations
2. **Lab 2:** String manipulation and array processing
3. **Lab 3:** Interfacing with 8255 PPI
4. **Lab 4:** ADC and DAC interfacing
5. **Lab 5:** Stepper motor control

8051 Experiments:

6. **Lab 6:** 8051 programming in assembly and C
7. **Lab 7:** LED patterns and seven-segment displays
8. **Lab 8:** LCD interfacing and programming
9. **Lab 9:** Keyboard interfacing
10. **Lab 10:** Timer programming and PWM generation
11. **Lab 11:** Serial communication with PC
12. **Lab 12:** Temperature monitoring system
13. **Lab 13:** DC motor speed control

COURSE CODE: BTIT404
COURSE NAME: ADVANCED DATA STRUCTURE

Course Objective

This course explores advanced data structures beyond basic lists, stacks, and queues. It covers efficient data organization techniques, probabilistic structures, geometric structures, and external memory structures. Emphasis is on theoretical analysis, practical implementation, and applications in modern computing domains like databases, networks, and computational geometry.

CO1	Design and implement balanced tree structures for efficient search operations
CO2	Apply probabilistic structures for approximate query processing in large datasets
CO3	Implement geometric structures for spatial data processing and queries
CO4	Design concurrent and persistent data structures for multi-threaded applications

Unit	Content	Credit	Weightage
I	Advanced Tree Structures 1.1 Balanced Search Trees 1.2 Advanced Tree Variants 1.3 Trie Structures 1.4 Spatial Trees	1	25%
II	Hashing and Probabilistic Data Structures 2.1 Advanced Hashing Techniques 2.2 Bloom Filters and Variants 2.3 Sketching Data Structures 2.4 Streaming Algorithms Structures	1	25%
III	Geometric and Graph-Based Structures 3.1 Geometric Data Structures 3.2 Priority Queue Variants 3.3 Graph-Based Structures	1	25%

	3.4 Succinct Data Structures		
IV	External Memory and Concurrent Structures 4.1 External Memory Structures 4.2 Concurrent Data Structures 4.3 Persistent Data Structures 4.4 Specialized Structures and Applications	1	25%

Textbooks:

- "Introduction to Algorithms" by Cormen, Leiserson, Rivest, and Stein (Chapters on advanced structures)
- "The Art of Computer Programming, Volume 3: Sorting and Searching" by Donald E. Knuth
- "Data Structures and Algorithms in Python" by Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser

Reference Books:

- "Handbook of Data Structures and Applications" edited by Dinesh P. Mehta and Sartaj Sahni
- "Algorithms on Strings, Trees, and Sequences" by Dan Gusfield
- "Computational Geometry: Algorithms and Applications" by Mark de Berg et al.
- The Design and Analysis of Concurrent Data Structures" by Mark Moir and Nir Shavit

Online Resources:

- MIT Open Course Ware: Advanced Data Structures
- Stanford CS166: Data Structures (online materials)
- Visual Algo: Advanced data structure visualizations
- CP-Algorithms: Advanced topics with implementations
- Research Papers: Key papers on modern data structures

Laboratory Experiments

1. **Lab 1:** AVL and Red-Black tree implementation with visualization
2. **Lab 2:** B-Tree implementation for disk-based storage simulation
3. **Lab 3:** Bloom filter variants and performance comparison
4. **Lab 4:** Count-Min Sketch for stream frequency estimation
5. **Lab 5:** kd-Tree for nearest neighbor search in 2D
6. **Lab 6:** Fibonacci heap and Dijkstra's algorithm optimization
7. **Lab 7:** Concurrent linked list and queue implementation
8. **Lab 8:** Persistent data structure implementation
9. **Lab 9:** Suffix array construction and pattern matching
10. **Lab 10:** Final project - Real-world application using advanced structures

MK UNIVERSITY

PATAN, GUJARAT

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

COURSE CODE: BTIT405

COURSE NAME: INTRODUCTION TO IOT

Course Objective

This course introduces the fundamentals of Internet of Things (IoT) systems, covering architecture, protocols, sensing/actuation, data processing, and security. Emphasis is placed on hands-on development of IoT applications using popular platforms, understanding end-to-end IoT systems, and exploring real-world applications in smart cities, healthcare, and industry.

CO1	Design IoT system architectures for different application domains
CO2	Select appropriate communication protocols based on application requirements
CO3	Develop IoT applications using popular hardware platforms and cloud services
CO4	Implement security measures for IoT devices and communications

Unit	Content	Credit	Weightage
I	IoT Fundamentals and Architecture 1.1 Introduction to IoT 1.2 IoT Architecture and Layers 1.3 IoT Hardware Components 1.4 IoT Software Stack	1	25%
II	IoT Connectivity and Communication Protocols 2.1 Short-Range Wireless Protocols 2.2 LPWAN (Low Power Wide Area Networks) 2.3 IoT Network Architecture 2.4 IoT Application Layer Protocols	1	25%
III	IoT Data and Platform Services 3.1 IoT Data Processing and Analytics 3.2 IoT Cloud Platforms 3.3 IoT Application Development 3.4 IoT and Emerging Technologies	1	25%
IV	IoT Security, Privacy, and Applications 4.1 IoT Security Fundamentals 4.2 IoT Attacks and Countermeasures 4.3 IoT Privacy and Ethics 4.4 IoT Applications and Case Studies	1	25%

Textbooks:

- "Internet of Things: Principles and Paradigms" by Rajkumar Buyya and Amir Vahid Dastjerdi
- "Getting Started with the Internet of Things" by Cuno Pfister
- "Building the Internet of Things with ESP32 and NodeJS" by Cătălin Mariș

Reference Books:

- "IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things" by David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Robert Barton, Jerome Henry
- "The Internet of Things: Enabling Technologies, Platforms, and Use Cases" by Pethuru Raj and Anupama C. Raman
- "Practical IoT Hacking" by Fotios Chantzis, Ioannis Stais, Paulino Calderon, and Beau Woods

Online Resources:

- IoT Alliance Guidelines and Whitepapers
- AWS IoT Tutorials and Documentation
- Azure IoT Learning Paths
- Arduino Project Hub (IoT projects)
- Hackster.io (IoT project community)
- IoT For All (Online publication)
- IEEE IoT Journal

Laboratory Experiments

- Lab 1: Setting up IoT development environment
- Lab 2: Sensor interfacing and data acquisition (DHT22, MPU6050)
- Lab 3: Wi-Fi connectivity and basic HTTP communication
- Lab 4: MQTT implementation with local broker (Mosquitto)
- Lab 5: Cloud integration (AWS IoT/Azure IoT)
- Lab 6: BLE communication and beacon implementation
- Lab 7: Edge computing with Raspberry Pi (data preprocessing)
- Lab 8: Building IoT dashboard with Node-RED
- Lab 9: Implementing security features (TLS, authentication)
- Lab 10: Complete IoT project - Smart home/Environmental monitoring

MK UNIVERSITY

PATAN, GUJARAT

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

SEMESTER-V

COURSE CODE: BTIT501
COURSE NAME: COMPUTER NETWORK

Course Objective

- Understand fundamental networking concepts, models, and layered architectures.
- Analyze data link layer protocols, error control, and media access mechanisms.
- Design IP addressing schemes and explain routing protocols.
- Evaluate transport layer protocols (TCP/UDP) and congestion control mechanisms.
- Implement basic network applications and analyze network performance.

Course Outcomes:

CO1	Explain network architectures, physical layer technologies, and transmission fundamentals
CO2	Apply error detection/correction techniques and data link layer protocols
CO3	Design subnetting strategies and compare routing algorithms.
CO4	Analyze TCP/UDP operations and implement client-server applications.

Unit	Content	Credit	Weightage
I	**Fundamentals & Physical Layer** • **Introduction to Networks:** Uses, types (LAN, MAN, WAN), network topologies, network models (OSI, TCP/IP). • **Physical Layer:** Transmission media (guided/unguided), digital/analog transmission, multiplexing (FDM, TDM), switching (circuit, packet, message), bandwidth, throughput, latency. • **Data Encoding & Modulation:** NRZ, Manchester, PCM, ASK, FSK, PSK. • **Introduction to Network Hardware:** Hubs, repeaters, cables, connectors.	1	25%
II	**Data Link Layer & MAC Sublayer** • **Data Link Layer Design Issues:** Framing, error control, flow control. • **Error Detection & Correction:** Parity, checksum, CRC, Hamming codes. • **Data Link Protocols:** Stop-and-Wait, Sliding Window, Go-Back-N, Selective Repeat. • **MAC Sublayer:** Channel allocation, multiple access protocols – ALOHA, CSMA, CSMA/CD, CSMA/CA. • **LAN Technologies:** Ethernet (802.3), Wireless LANs (802.11), VLANs, PPP.	1	25%

	• **Switching:** Bridges vs. switches, MAC learning, spanning tree protocol.		
III	**Network Layer & Routing** • **Network Layer Functions:** Logical addressing, routing, fragmentation. • **IP Addressing:** IPv4, subnetting, CIDR, IPv6 basics. • **Routing Algorithms:** Static vs. dynamic routing, distance vector (RIP), link state (OSPF), path vector (BGP). • **IP Protocol:** IP datagram format, ARP, ICMP, DHCP. • **Network Layer Devices:** Routers, routing tables. • **Introduction to QoS and Multicast.**	1	25%
IV	**Transport Layer & Application Layer** • **Transport Layer:** Role and services, connectionless vs. connection-oriented. • **TCP & UDP:** Segment format, TCP connection management (3-way handshake), flow control, congestion control (slow start, AIMD), UDP features. • **Application Layer Protocols:** DNS, HTTP/HTTPS, FTP, SMTP, POP/IMAP. • **Introduction to Network Security:** Basic concepts of firewalls, TLS/SSL, VPNs. • **Socket Programming Basics.**	1	25%

Textbooks:

- "Computer Networking: A Top-Down Approach"
  Kurose & Ross (7th Edition) – Excellent for application-first learning.
- "Computer Networks"
  Andrew S. Tanenbaum & David J. Wetherall (5th Edition) – Strong on theory and layers.
- "Data Communications and Networking"
  Behrouz A. Forouzan (5th Edition) – Clear explanations with good visuals.

Reference Books:

- "TCP/IP Illustrated, Vol. 1" *W. Richard Stevens* – In-depth protocol details.
- "Internetworking with TCP/IP, Vol. 1" *Douglas E. Comer* – Classic TCP/IP reference.
- "Network Programmability and Automation" *Jason Edelman et al.* – For modern network automation topics.

Online Resources:

- Cisco Networking Academy courses
- Geeks for Geeks Computer Networks section
- Wireshark documentation and sample captures

Laboratory Experiments
Simulation & Programming (Using Cisco Packet Tracer / NS3)

- **Experiment 1:** Study of network devices and cables.
- **Experiment 2:** Basic network configuration and connectivity testing.
- **Experiment 3:** Implementation of framing methods (Bit/Byte stuffing).

- **Experiment 4:** Error detection using CRC and Hamming codes.
- **Experiment 5:** Stop-and-Wait and Sliding Window protocol simulation.
- **Experiment 6:** LAN setup and VLAN configuration.
- **Experiment 7:** Subnetting practice and IP address allocation.
- **Experiment 8:** Static and dynamic routing (RIP/OSPF) configuration.
- **Experiment 9:** TCP and UDP client-server programs (Socket programming).
- **Experiment 10:** DNS and HTTP packet analysis.

COURSE CODE: BTIT502
COURSE NAME: WEB TECHNOLOGIES
Course Objective

- Design and develop responsive web applications using HTML5, CSS3, and JavaScript.
- Implement server-side programming using Node.js and PHP with database integration.
- Develop dynamic web applications using React.js and Angular frameworks.
- Apply web security principles, RESTful APIs, and web services.
- Deploy web applications using cloud platforms and DevOps practices.
- Evaluate web application performance, accessibility, and SEO optimization.

Course Outcomes:

CO1	Design responsive web interfaces using HTML5, CSS3, and Bootstrap.
CO2	Develop interactive client-side applications using JavaScript and jQuery
CO3	Implement server-side applications using Node.js and PHP with MySQL/MongoDB
CO4	Build single-page applications using React.js and Angular frameworks

Unit	Content	Credit	Weightage
I	Frontend Technologies & Responsive Design • Web Fundamentals, HTML5, CSS3, Responsive Web Design, Bootstrap Framework, Version Control,	1	25%
II	Client-Side Programming & JavaScript • JavaScript Fundamentals, DOM Manipulation, Advanced JavaScript, jQuery Library, Frontend Build Tools, Browser Storage	1	25%
III	Server-Side Development & Databases • Server-Side Programming Concepts, Node.js and Express.js, PHP Programming, Database Integration Authentication & Security, Real-time Communication	1	25%
IV	Modern Frameworks & Deployment • React.js Framework, Angular Framework, Web Services & APIs, Testing Web Applications, Performance	1	25%

	Optimization, Deployment & DevOps, Progressive Web Apps (PWAs), SEO Fundamentals, Emerging Trends		

Textbooks:

- "Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics" *Jennifer Niederst Robbins* (5th Edition) *Excellent foundation for web design basics*
- "Eloquent JavaScript: A Modern Introduction to Programming" *Marijn Haverbeke* (3rd Edition) *Free online, comprehensive JavaScript guide*

Reference Books:

- "HTML and CSS: Design and Build Websites" *Jon Duckett Visually appealing, beginner-friendly HTML/CSS book*
- "JavaScript: The Definitive Guide" *David Flanagan* (7th Edition) *Comprehensive JavaScript reference*
- "Node.js Design Patterns" *Mario Casciaro & Luciano Mammino* (3rd Edition) *Advanced Node.js patterns and best practices*

Online Resources:

- MDN Web Docs (Mozilla Developer Network)
- W3Schools (Tutorials and references)
- Free Code Camp (Full web development curriculum)
- Frontend Masters (Video courses)
- CSS-Tricks (CSS tutorials and articles)
- React.js Official Documentation
- Angular Official Documentation
- Node.js Official Documentation
- NPTEL: "Web Technologies" by IIT Kharagpur

COURSE CODE: BTIT503
COURSE NAME: INFORMATION SECURITY
Course Objective

- Understand fundamental security concepts, threats, vulnerabilities, and risk management.
- Implement cryptographic algorithms and protocols for secure communication.
- Design network security architectures with firewalls, IDS/IPS, and VPNs.
- Apply security controls for systems, applications, and databases.
- Analyze security incidents and implement forensic investigation techniques.
- Evaluate compliance requirements and develop security policies.

Course Outcomes:

CO1	Analyze security threats, vulnerabilities, and implement risk managementframeworks.
CO2	Design and implement cryptographic solutions for data protection
CO3	Configure and evaluate network security devices and protocols
CO4	Implement security controls for operating systems and

	applications

Unit	Content	Credit	Weightage
I	Security Foundations & Cryptography • Introduction to Information Security, Cryptography Fundamentals, Network Security Protocols	1	25%
II	Network & System Security • Network Security Architecture, Firewalls and Access Control, Intrusion Detection/Prevention Systems, Virtual Private Networks (VPNs), Wireless Security, Operating System Security	1	25%
III	Application & Database Security • Web Application Security, Database Security, Malware Analysis & Protection, Mobile Security, Cloud Security	1	25%
IV	Security Operations & Management • Security Operations Center (SOC), Vulnerability Management, Penetration Testing, Incident Response, Legal and Compliance, Security Governance, Emerging Trends	1	25%

Textbooks:

- "Network Security Essentials: Applications and Standards" *William Stallings* (6th Edition) *Comprehensive coverage of network security protocols*
- "Principles of Information Security" *Michael E. Whitman & Herbert J. Mattord* (6th Edition) *Excellent for security management and governance*

Reference Books:

- "The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws" *Dafydd Stuttard & Marcus Pinto* (2nd Edition) *Essential for web application security*
- "Hacking: The Art of Exploitation" *Jon Erickson* (2nd Edition) *Practical exploitation techniques*
- "Cryptography and Network Security: Principles and Practice" *William Stallings* (8th Edition) *Detailed cryptographic algorithms coverage*

Online Resources:

- OWASP (Open Web Application Security Project)
- SANS Institute Reading Room
- NIST Cybersecurity Framework
- MITRE ATT&CK Framework
- CISA (Cybersecurity & Infrastructure Security Agency) Resources
- Kali Linux Documentation
- NPTEL: "Information Security" by Prof. V. Kamakoti

Practical List:

- Session 1: Cryptographic implementation (AES, RSA, hashing)
- Session 2: Wireshark packet analysis and protocol investigation
- Session 3: SSL/TLS analysis and certificate management

- Session 4: Firewall configuration (iptables/Windows Firewall)
- Session 5: Network scanning with Nmap
  6. Session 6: Vulnerability scanning with OpenVAS
- Session 7: Web application security testing with Burp Suite
  8. Session 8: Password cracking techniques and Défense
- Session 9: Snort IDS rule creation and monitoring
- Session 10: SIEM implementation with Splunk/ELK
- Session 11: Digital forensics with Autopsy
- Session 12: Memory forensics with Volatility

COURSE CODE: BTIT504
COURSE NAME: ADVANCED DATABASE MANAGEMENT SYSTEM
Course Objective

1. Analyze advanced database architectures, indexing techniques, and query optimization.
2. Design distributed and parallel database systems with consistency protocols.
3. Implement NoSQL databases for big data applications.
4. Develop data warehousing solutions and OLAP operations.
5. Design secure and reliable database systems with transaction management.
6. Evaluate emerging database technologies and trends.

Course Outcomes:

CO1	Design and optimize relational databases using advanced indexing and query processing techniques.
CO2	Implement distributed database systems with consistency and replication protocols
CO3	Design and develop NoSQL databases for specific application requirements
CO4	Implement data warehousing and business intelligence solutions

Unit	Content	Credit	Weightage
I	Advanced Relational Databases & Optimization • Database System Architecture, Advanced Indexing Techniques, Query Processing & Optimization, Transaction Processing, Database Recovery	1	25%
II	Distributed & Parallel Databases • Distributed Database Concepts, Distributed Query Processing, Distributed Transaction Management, Parallel Database Systems, Database Clustering	1	25%
III	NoSQL & NewSQL Databases • NoSQL Database Fundamentals, MongoDB (Document Database), Cassandra (Column-family Store), Redis (Key-Value Store), NewSQL Databases, Polyglot Persistence	1	25%

IV	Data Warehousing & Emerging Trends • Data Warehousing Fundamentals, ETL Processes, OLAP Operations, Data Cube and Materialized Views, Business Intelligence, Big Data Integration, Database Security, Emerging Database Technologies, Database Performance Tuning	1	25%

Textbooks:

- "Database System Concepts" *Abraham Silberschatz, Henry F. Korth, S. Sudarshan* (7th Edition)
  Comprehensive coverage of database concepts
- "Readings in Database Systems" *Joseph M. Hellerstein & Michael Stonebraker* (5th Edition)
  Collection of influential database research papers

Reference Books:

- "Database Management Systems" *Raghu Ramakrishnan & Johannes Gehrke* (3rd Edition)
  Excellent for query processing and optimization
- "Transactional Information Systems: Theory, Algorithms, and the Practice of Concurrency Control and Recovery" *Gerhard Weikum & Gottfried Vossen*
- "Data Management in the Cloud: Challenges and Opportunities"
  Divyakant Agrawal, Sudipto Das, Amr El Abbadi

Online Resources:

- PostgreSQL Documentation
- MongoDB University (Free courses)
- Cassandra Documentation
- NPTEL: "Advanced Database Management Systems" by IIT Madras
- Stanford Database Group (Research papers and courses)
- Google Research Papers (Spanner, Bigtable)
- AWS/Azure/GCP Database Services Documentation

COURSE CODE: BTIT505
COURSE NAME: CLOUD COMPUTING
Course Objective

- Understand cloud computing concepts, models, and service architectures.
- Analyze virtualization technologies and cloud deployment models.
- Design and deploy applications using cloud platforms (AWS/Azure/GCP).
- Implement storage, compute, and networking solutions in cloud environments.
- Evaluate cloud security, compliance, and cost management strategies.
- Compare emerging cloud technologies and migration approaches.

Course Outcomes:

CO1	Explain cloud computing characteristics, service models, and deployment models.
CO2	Analyze virtualization techniques and containerization technologies
CO3	Deploy and manage applications using major cloud platforms
CO4	Implement cloud storage, networking, and compute services

Unit	Content	Credit	Weightage
I	Cloud Foundations & Virtualization, Introduction to Cloud Computing, Cloud Service Models, Cloud Deployment Models, Virtualization Fundamentals	1	25%
II	Cloud Architecture & Core Services, Cloud Reference Architecture, Compute Services, Storage Services, Networking in Cloud, Database Services	1	25%
III	Cloud Management & Security • Cloud Management & Monitoring, Cloud Security, Compliance & Governance, Cost Management	1	25%
IV	Advanced Cloud Services & Trends • Cloud-Native Development, Big Data & Analytics in Cloud, Multi-cloud & Hybrid Cloud, Emerging Trends	1	25%

Textbooks:

- "Cloud Computing: Concepts, Technology & Architecture" *Thomas Erl, Ricardo Puttini, Zaigham Mahmood Comprehensive coverage of cloud patterns and best practices*
- "Cloud Computing: A Practical Approach" *Anthony T. Velte, Toby J. Velte, Robert Elsenpeter*
  Excellent for hands-on learning and implementation
- "Architecting the Cloud: Design Decisions for Cloud Computing Service Models" *Michael J. Kavis Great for architectural decision-making*

Reference Books:

- "The Cloud Adoption Playbook" *Moe Abdula, Ingo Averdunk, et al. Practical guide for enterprise cloud adoption*
- "Site Reliability Engineering: How Google Runs Production Systems" *Betsy Beyer, et al. SRE principles in cloud context*
- "Cloud Native Patterns: Designing Change-tolerant Software" *Cornelia Davis Modern cloud-native application design*

Online Resources:

- AWS/Azure/GCP Documentation & Free Tier
- Cloud Academy, A Cloud Guru (Video courses)
- NPTEL: "Cloud Computing" by Prof. Soumya Kanti Ghosh
- Cloud Computing Specialization (Coursera - UIUC)

MK UNIVERSITY

PATAN, GUJARAT

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

SEMESTER-VI

COURSE CODE: BTIT601
COURSE NAME: SYSTEM ADMINISTRATION AND MAINTENANCE
Course Objective

- Understand system administration principles, roles, and responsibilities.
- Configure and manage Windows and Linux server environments.
- Implement network services, directory services, and virtualization.
- Design and maintain storage, backup, and disaster recovery solutions.
- Monitor system performance, security, and implement automation.
- Troubleshoot system issues and maintain IT infrastructure.

Course Outcomes:

CO1	Analyze system administration roles and implement IT service management processes.
CO2	Configure and manage Windows Server and Linux server environments
CO3	Implement network services, directory services, and virtualization technologies
CO4	Design and maintain storage, backup, and disaster recovery solutions

Unit	Content	Credit	Weightage
I	Foundations of System Administration • Introduction to System Administration, IT Service Management, Server Hardware Fundamentals, Operating System Installation & Configuration, User and Group Management, File System Management	1	25%
II	Network Services & Directory Services • Network Configuration & Management, Directory Services, Network Services Implementation, Remote Access Services, Network Security Services	1	25%
III	Storage, Virtualization & Cloud Management • Storage Management, Backup & Disaster Recovery, Virtualization Technologies, Containerization, Cloud Infrastructure Management	1	25%
IV	Monitoring, Security & Automation • System Monitoring & Performance Tuning, System Security & Hardening, Automation & Scripting, High Availability & Clustering, Documentation & Change Management, Troubleshooting Methodology	1	25%

Textbooks:

- "The Practice of System and Network Administration"
  Thomas A. Limoncelli, Christina J. Hogan, Strata R. Chalup (3rd Edition)
  Comprehensive guide to system administration practices
- "Windows Server 2022 & PowerShell All-in-One For Dummies"
  Sara Perrott
  Practical Windows server administration guide

Reference Books:

- "UNIX and Linux System Administration Handbook" *Evi Nemeth, Garth Snyder, Trent R. Hein, Ben Whaley* (5th Edition) *Definitive Linux/Unix administration reference*
- "Mastering VMware vSphere 8" *Nick Marshall, Mike Brown, et al. Comprehensive virtualization administration guide*
- "Automate the Boring Stuff with Python" *Al Sweigart* (2nd Edition) *Practical Python for automation*

Online Resources:

- Microsoft Learn (Windows Server courses)
- Red Hat Learning Subscription
- VMware Education
- AWS Training and Certification
- Google Cloud Training
- NPTEL: "System Administration" courses
- Spiceworks Community
- Server Fault (Q&A forum)

COURSE CODE: BTIT602
COURSE NAME: CRYPTOGRAPHY AND NETWORK SECURITY
Course Objective

- Understand fundamental cryptographic principles, algorithms, and protocols.
- Analyze symmetric and asymmetric encryption techniques and their applications.
- Design secure communication protocols for various network environments.
- Implement cryptographic algorithms and security mechanisms.
- Evaluate network security threats, vulnerabilities, and countermeasures.
- Apply authentication, integrity, and confidentiality mechanisms in real-world scenarios.

Course Outcomes:

CO1	Explain basic cryptographic concepts, classical ciphers, and modern encryption principles.
CO2	Implement and analyze symmetric encryption algorithms (DES, AES, etc.).
CO3	Apply asymmetric cryptography (RSA, ECC) and hash

	functions in security applications
CO4	Design and analyze authentication protocols and digital signature schemes

Unit	Content	Credit	Weightage
I	Foundations of Cryptography • Introduction to Security, Mathematics for Cryptography, Classical Encryption Techniques, Modern Cryptography Principles	1	25%
II	Symmetric Cryptography & Hash Functions • Symmetric Key Algorithms, Cryptographic Hash Functions	1	25%
III	Asymmetric Cryptography & Key Management • Public Key Cryptography, Digital Signatures, Key Management	1	25%
IV	Network Security Applications Authentication Protocols, Email Security, Web Security, Network Security Mechanisms, System Security	1	25%

Textbooks:

- "Cryptography and Network Security: Principles and Practice" *William Stallings* (8th Edition) *Comprehensive coverage with excellent examples and problems*
- "Network Security: Private Communication in a Public World" *Charlie Kaufman, Radia Perlman, Mike Speciner* (2nd Edition) *Great for protocol-level understanding*
- "Applied Cryptography: Protocols, Algorithms, and Source Code in C" *Bruce Schneier* (2nd Edition) *Classic reference with practical implementations*

Reference Books:

- "Introduction to Modern Cryptography" *Jonathan Katz & Yehuda Lindell* (3rd Edition) *Rigorous mathematical treatment*
- "Handbook of Applied Cryptography" *Alfred J. Menezes, Paul C. van Oorschot, Scott A. Vanstone Comprehensive reference with algorithms*
- "Computer Security: Principles and Practice" *William Stallings & Lawrie Brown Broad coverage including system security*

Online Resources:

- Cryptopals Crypto Challenges (Matasano challenges)
- NIST Cryptographic Standards and Guidelines
- NPTEL: "Cryptography and Network Security" by Prof. Debdeep Mukhopadhyay
- Stanford Cryptography I & II (Coursera)
- OWASP Top 10 Security Risks

Practical List:

- **Session 1:** Implementation of classical ciphers (Caesar, Vigenère)

- **Session 2:** Cryptanalysis of substitution ciphers using frequency analysis
- **Session 3:** DES implementation (using libraries) and understanding rounds
- **Session 4:** AES implementation and comparison with DES
- **Session 5:** Block cipher modes of operation implementation
- **Session 6:** Cryptographic hash functions implementation (SHA-256)
- **Session 7:** HMAC implementation and verification
- **Session 8:** RSA algorithm implementation (key generation, encryption, decryption)
- **Session 9:** Diffie-Hellman key exchange simulation
- **Session 10:** Digital signature implementation using RSA

COURSE CODE: BTIT603
COURSE NAME: ADVANCE COMPUTER ARCHITECTURE
Course Objective

- Analyze performance metrics and quantitative principles of computer architecture.
- Design pipelined processors with hazard detection and resolution mechanisms.
- Evaluate memory hierarchy design, cache organizations, and virtual memory systems.
- Implement instruction-level parallelism techniques including superscalar and VLIW architectures.
- Compare different parallel architectures including multicore, multiprocessor, and GPU architectures.
- Design power-efficient and reliable computing systems.

Course Outcomes:

CO1	Calculate and analyze performance metrics using CPU performance equations.
CO2	Design and analyze pipelined processors with hazard handling mechanisms
CO3	Evaluate memory hierarchy designs including cache organizations and virtual memory
CO4	Implement instruction-level parallelism techniques and analyze their impact

Unit	Content	Credit	Weightage
I	Performance Evaluation & Instruction Set Principles • Introduction to Computer Architecture, Quantitative Principles, Instruction Set Architecture (ISA), Case Studies	1	25%
II	Pipeline & Instruction-Level Parallelism • Pipelining Fundamentals, Hazard Resolution Techniques, Advanced Pipelining	1	25%

III	Memory Hierarchy & I/O Systems • Memory Hierarchy Design, Advanced Cache Optimizations, Virtual Memory, Storage Systems	1	25%
IV	Parallel Architectures & Emerging Trends • Parallel Processing Architectures, Multicore Architectures, Specialized Architectures, Emerging Trends	1	25%

Textbooks:

- "Computer Architecture: A Quantitative Approach" John *L. Hennessy & David A. Patterson* (6th Edition) *The "Bible" of computer architecture – Comprehensive and authoritative*
- "Computer Organization and Design: The Hardware/Software Interface" *David A. Patterson & John L. Hennessy* (ARM or RISC-V Edition) *Excellent for fundamental concepts with practical examples*

Reference Books:

- "Advanced Computer Architecture: Parallelism, Scalability, Programmability" *Kai Hwang Great coverage of parallel architectures*
- "Modern Processor Design: Fundamentals of Superscalar Processors" *John Paul Shen & Mikko H. Lipasti In-depth coverage of modern processor design*
- "Parallel Computer Architecture: A Hardware/Software Approach" *David E. Culler, Jaswinder Pal Singh, Anoop Gupta Comprehensive parallel architecture coverage*

Online Resources:

- MIT Open Course Ware: 6.823 Computer Systems Architecture
- NPTEL: "Computer Architecture" by Prof. Smruti Ranjan Sarangi
- Computer Architecture - Coursera (Princeton University)
- IEEE/ACM Transactions on Computer Architecture
- SPEC Benchmark Suite documentation

COURSE CODE: BTIT604
COURSE NAME: ADVANCE WEB TECHNOLOGIES
Course Objective

- Design and develop modern web applications using advanced frameworks and architectures.
- Implement microservices architecture and containerization for scalable web applications.
- Develop Progressive Web Applications (PWAs) and mobile-responsive web solutions.
- Apply advanced security measures, performance optimization, and testing strategies.
- Integrate emerging technologies like WebSockets, WebRTC, and WebAssembly.
- Deploy and monitor web applications using DevOps practices and cloud platforms.

Course Outcomes:

CO1	Design and develop Single Page Applications (SPAs) using modern JavaScript frameworks.
CO2	Implement microservices architecture and containerization for web applications

CO3	Develop Progressive Web Applications (PWAs) with offline capabilities
CO4	Apply advanced security measures and performance optimization techniques

Unit	Content	Credit	Weightage
I	Modern JavaScript Frameworks & SPAs • Advanced JavaScript Concepts, TypeScript Fundamentals, React.js Advanced Concepts, Vue.js Framework, State Management Solutions	1	25%
II	Microservices & Containerization • Microservices Architecture, Containerization with Docker, Container Orchestration, Service Communication, API Design & Documentation, Service Mesh Implementation	1	25%
III	Advanced Web Features & Real-time Applications • Progressive Web Applications (PWAs), Real-time Web Applications, Web Performance Optimization, Web Accessibility (A11y), Internationalization (i18n) & Localization, Web Assembly (Wasm)	1	25%
IV	DevOps, Security & Deployment • Modern Development Workflow, Cloud Deployment Platforms, Web Application Security, Monitoring & Observability, Search Engine Optimization (SEO), Emerging Technologies, Performance Budgeting & Optimization	1	25%

Textbooks:

- "Learning React: Modern Patterns for Developing React Apps" *Alex Banks & Eve Porcello* (2nd Edition) *Modern React patterns and best practices*
- "Microservices Patterns: With examples in Java" *Chris Richardson Comprehensive microservices architecture guide*

Reference Books:

- "Designing Data-Intensive Applications" *Martin Kleppmann Modern perspective on scalable web applications*
- "Progressive Web Apps: Building Faster, Engaging, and More Resilient Web Applications" *Dean Alan Hume Practical PWA implementation guide*
- "The Docker Book: Containerization is the New Virtualization" *James Turnbull Comprehensive Docker reference*

Online Resources:

- React Official Documentation

- Vue.js Official Documentation
- Next.js Documentation
- Docker Documentation
- Kubernetes Documentation
- MDN Web Docs (Advanced web APIs)
- Web.dev (Google's web fundamentals)
- NPTEL: "Advanced Web Technologies" courses

Practical List:

- **Session 1:** React Hooks implementation with TypeScript
- **Session 2:** Next.js application with SSR and SSG
- **Session 3:** Vue 3 Composition API application
- **Session 4:** State management with Redux Toolkit
- **Session 5:** Graph QL server with Apollo
- **Session 6:** Docker containerization of web application
- **Session 7:** Microservices architecture implementation
- **Session 8:** Kubernetes deployment with Helm
- **Session 9:** Service mesh implementation with Istio
- **Session 10:** PWA implementation with Service Workers
- **Session 11:** Real-time chat application with Web Sockets
- **Session 12:** Web Assembly module integration

COURSE CODE: BTIT605
COURSE NAME: ENTERPRISE COMPUTING
Course Objective

1. Understand enterprise computing architecture, infrastructure, and business processes.
2. Design and implement Enterprise Resource Planning (ERP) systems.
3. Develop enterprise applications using middleware and integration technologies.
4. Analyze business processes and implement workflow automation solutions.
5. Design and manage enterprise data management and business intelligence systems.
6. Evaluate enterprise security, governance, and compliance requirements.

Course Outcomes:

CO1	Analyze enterprise computing architecture and business processrequirements.
CO2	Design and implement ERP systems for organizational needs
CO3	Develop enterprise applications using middleware and integration platforms
CO4	Implement business process automation and workflow management systems

Unit	Content	Credit	Weightage
I	Enterprise Architecture & Infrastructure • Introduction to Enterprise Computing, Enterprise Architecture Frameworks, Enterprise Infrastructure, Middleware Technologies, Service-Oriented	1	25%

	Architecture (SOA), Enterprise Application Integration (EAI)		
II	Enterprise Resource Planning (ERP) Systems • ERP Fundamentals, Major ERP Systems, ERP Implementation Lifecycle, ERP Customization and Extension, ERP Data Management, ERP Analytics and Reporting	1	25%
III	Business Process Management & Workflow Systems • Business Process Management (BPM), Workflow Management Systems, Enterprise Content Management (ECM), Customer Relationship Management (CRM), Supply Chain Management (SCM), Enterprise Mobility	1	25%
IV	Enterprise Security, Governance & Emerging Trends • Enterprise Security Architecture, IT Governance and Compliance, Enterprise Data Management, Business Intelligence and Analytics, Emerging Technologies in Enterprise Computing, Enterprise Application Development, Disaster Recovery and Business Continuity, Enterprise Performance Management	1	25%

Textbooks:

- "Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions" *Gregor Hohpe & Bobby Woolf Definitive guide to enterprise integration patterns*
- "ERP: Making It Happen: The Implementers' Guide to Success with Enterprise Resource Planning" *Thomas F. Wallace & Michael H. Kremzar Practical ERP implementation guide*

Reference Books:

- "Enterprise Architecture as Strategy: Creating a Foundation for Business Execution" *Jeanne W. Ross, Peter Weill, David Robertson Strategic perspective on enterprise architecture*
- "Business Process Management: Concepts, Languages, Architectures" *Mathias Weske* (3rd Edition) *Comprehensive BPM reference*
- "The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modelling" *Ralph Kimball & Margy Ross* (3rd Edition) *Essential for enterprise data management*

Online Resources:

- SAP Community Network
- Oracle Enterprise Architecture Framework
- The Open Group (TOGAF resources)
- Gartner Research (Enterprise technology trends)
- Forrester Research
- NPTEL: "Enterprise Systems" courses
- Coursera: "Enterprise Architecture" specializations
- edX: "Digital Transformation" courses

MK UNIVERSITY

PATAN, GUJARAT

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

SEMESTER-VII

COURSE CODE: BTIT701
COURSE NAME: IT PROJECT MANAGEMENT
Course Objective

- Understand project management principles, methodologies, and frameworks in IT context.
- Apply project management processes for initiating, planning, executing, monitoring, and closing IT projects.
- Develop project plans, schedules, budgets, and risk management strategies.
- Analyze stakeholder requirements and manage project communications effectively.
- Implement agile methodologies and hybrid approaches for IT projects.
- Evaluate project performance using metrics and manage project quality and resources.

Course Outcomes:

CO1	Analyze IT project requirements and develop project charters and scope statements.
CO2	Create comprehensive project plans including schedule, budget, and resource allocation
CO3	Apply risk management techniques to identify, analyze, and mitigate project risks
CO4	Implement agile methodologies and manage IT project iterations

Unit	Content	Credit	Weightage
I	Foundations of IT Project Management • Introduction to Project Management, Project Management Frameworks, Project Life Cycle, Project Selection and Prioritization, Project Initiation, Organizational Structures and Culture, Professional Ethics in IT Project Management	1	25%
II	Project Planning & Scope Management • Project Planning Processes, Scope Management, Time Management, Cost Management, Resource Management	1	25%
III	Agile Project Management & Execution • Agile Methodologies, Hybrid Methodologies, Project Quality Management, Risk Management, Procurement Management	1	25%
IV	Monitoring, Control & Project Closure • Project Monitoring and Control, Communication Management, Stakeholder Management, Project Closure, IT Project Governance, Tools and Software	1	25%

	for IT Project Management, Emerging Trends in IT Project Management		

Textbooks:

- "Information Technology Project Management" *Kathy Schwalbe* (10th Edition) *Comprehensive IT-focused project management textbook*
- "A Guide to the Project Management Body of Knowledge (PMBOK Guide)" *Project Management Institute* (7th Edition) *PMI's standard for project management*

Reference Books:

- "Agile Estimating and Planning" *Mike Cohn Excellent for agile project planning techniques*
- "Scrum: The Art of Doing Twice the Work in Half the Time" *Jeff Sutherland Practical guide to Scrum methodology*
- "The Mythical Man-Month: Essays on Software Engineering" *Frederick P. Brooks Jr. Classic software project management insights*

Online Resources:

- Project Management Institute (PMI) resources
- Scrum Alliance learning resources
- Agile Alliance resources
- NPTEL: "Software Project Management" by IIT Kharagpur
- Coursera: "IT Project Management" specialization
- edX: "Project Management" courses
- MIT Open Course Ware: Project management materials

COURSE CODE: BTIT702
COURSE NAME: CYBER FORENSICS
Course Objective

- Understand cyber forensics principles, legal frameworks, and investigation methodologies.
- Apply digital evidence collection, preservation, and chain of custody procedures.
- Analyze file systems, operating systems, and network forensic evidence.
- Implement forensic tools and techniques for data recovery and investigation.
- Conduct malware analysis and mobile device forensics investigations.
- Prepare forensic reports and testify as expert witnesses in legal proceedings.

Course Outcomes:

CO1	Analyze cyber forensics principles, legal requirements, and investigationmethodologies.
CO2	Implement digital evidence collection, preservation, and chain of custody procedures
CO3	Conduct file system and operating system forensic analysis
CO4	Perform network forensics and malware analysis investigations

Unit	Content	Credit	Weightage

I	Foundations of Cyber Forensics • Introduction to Cyber Forensics, Legal and Ethical Framework, Forensic Investigation Methodology, Digital Evidence Fundamentals, Forensic Laboratory Standards, Ethical Considerations	1	25%
II	Computer & Operating System Forensics • Computer Hardware Forensics, File System Forensics, Windows Forensics, Linux/Unix Forensics, Mac OS Forensics, Data Recovery Techniques, Steganography Detection	1	25%
III	Network & Malware Forensics • Network Forensics Fundamentals, Network Traffic Analysis, Network Device Forensics, Internet and Cloud Forensics, Malware Forensics, Incident Response	1	25%
IV	Mobile & Specialized Forensics • Mobile Device Forensics, Android Forensics, iOS Forensics, Cloud Forensics, IoT Forensics, Database Forensics, Forensic Report Writing, Anti-Forensics Techniques, Emerging Trends	1	25%

Textbooks:

- "Guide to Computer Forensics and Investigations" *Bill Nelson, Amelia Phillips, Christopher Steuart* (6th Edition)
  Comprehensive computer forensics textbook with practical approach "Digital Forensics with Open Source Tools" *Cory Altheide & Harlan Carvey Excellent for practical open-source forensics tools*

Reference Books:

- "The Art of Memory Forensics: Detecting Malware and Threats in Windows, Linux, and Mac Memory" *Michael Hale Ligh, Andrew Case, Jamie Levy, Aaron Walters Definitive guide to memory forensics*
- "File System Forensic Analysis" *Brian Carrier In-depth file system forensics reference*
- "Network Forensics: Tracking Hackers through Cyberspace" *Sherri Davidoff & Jonathan Ham Comprehensive network forensics guide*

Online Resources:

- NIST Digital Forensics Guidelines
- SANS Digital Forensics and Incident Response (DFIR) Resources
- National Cyber Crime Reporting Portal (India)
- Open-Source Digital Forensics Tools Documentation
- NPTEL: "Cyber Forensics" courses
- Coursera: "Digital Forensics" specializations

MK UNIVERSITY

PATAN, GUJARAT

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

COURSE CODE: BTIT703
COURSE NAME: WIRELESS NETWORKS

Course Objective

- Understand wireless communication fundamentals, propagation models, and modulation techniques.
- Analyze wireless network architectures, protocols, and standards.
- Design and implement Wi-Fi networks with security configurations.
- Evaluate cellular network technologies from 2G to 5G and beyond.
- Implement wireless sensor networks and IoT communication protocols.
- Design secure wireless networks and troubleshoot performance issues.

Course Outcomes:

CO1	Analyze wireless propagation characteristics and modulatio techniques.
CO2	Design and implement IEEE 802.11 (Wi-Fi) networks with various configurations
CO3	Evaluate cellular network architectures and mobile communication technologies
CO4	Implement wireless sensor networks and IoT communication protocols

Unit	Content	Credit	Weightage
I	Wireless Communication Fundamentals • Introduction to Wireless Networks, Wireless Transmission Fundamentals, Antenna Fundamentals, Modulation Techniques, Multiple Access Techniques, Wireless Network Architecture	1	25%
II	Wi-Fi Networks & IEEE 802.11 Standards • IEEE 802.11 Standard Family, Wi-Fi Physical Layer, Wi-Fi MAC Layer, Wi-Fi Network Operation, Wi-Fi Configuration and Management, Enterprise Wi-Fi	1	25%
III	Cellular Networks & Mobile Communication • Cellular Network Fundamentals, Generations of Cellular Networks, Cellular Network Components, Mobile IP and Mobility Management, Satellite Communication	1	25%
IV	Advanced Wireless Technologies & Security • Wireless Sensor Networks (WSN), Internet of Things (IoT) Wireless Technologies, Wireless Network Security, Wireless Network Planning and Design, Wireless Network Troubleshooting, Emerging	1	25%

	Wireless Technologies, Wireless Network Management		

Textbooks:

- "Wireless Communications and Networks" *William Stallings* (2nd Edition) *Comprehensive coverage of wireless technologies*
- "802.11 Wireless Networks: The Definitive Guide" *Matthew S. Gast* (2nd Edition) *Excellent Wi-Fi reference book*

Reference Books:

- "Wireless Communications: Principles and Practice" *Theodore S. Rappaport* (2nd Edition) *Classic wireless communication textbook*
- "Fundamentals of WiMAX: Understanding Broadband Wireless Networking" *Jeffrey G. Andrews, Arunabha Ghosh, Rias Muhamed WiMAX and broadband wireless reference*
- "5G Mobile and Wireless Communications Technology" *Afif Osseiran, Jose F. Monserrat, Patrick Marsch Comprehensive 5G technology guide*

Online Resources:

- IEEE 802.11 Standards Documentation
- 3GPP Specifications (Cellular standards)
- Wi-Fi Alliance documentation
- Cisco Wireless Networking Guides
- NPTEL: "Wireless Communication" courses
- Coursera: "Wireless Communication" specializations
- Aruba/HPE Wireless Networking Resources

Practical List:

- **Session 1: Wi-Fi network setup and basic configuration**
- **Session 2: Wireless site survey and channel analysis**
- **Session 3: Wi-Fi security configuration (WPA2, WPA3)**
- **Session 4: Enterprise Wi-Fi with RADIUS authentication**
- **Session 5: Wireless packet analysis with Wireshark**
- **Session 6: GSM network architecture simulation**
- **Session 7: LTE network configuration basics**
- **Session 8: Mobile IP configuration and testing**
- **Session 9: 5G network slicing concepts**
- **Session 10: Zigbee network setup and configuration**
- **Session 11: Bluetooth Low Energy (BLE) communication**
- **Session 12: LoRaWAN network implementation**

COURSE CODE: BTIT704
COURSE NAME: IT SERVICE MANAGEMENT
Course Objective

- Understand IT service management principles, frameworks, and business alignment.
- Apply ITIL framework processes for service design, transition, and operation.
- Implement service desk operations, incident, problem, and change management.
- Design service level agreements and manage service catalog and portfolio.
- Evaluate service performance using metrics and continuous improvement methods.
- Analyze emerging trends and integrate ITSM with DevOps and cloud services.

Course Outcomes:

CO1	Analyze IT service management concepts and business valuealignment.
CO2	Apply ITIL framework processes for service lifecycle management
CO3	Implement incident, problem, change, and service request management
CO4	Design service level agreements and manage service catalog

Unit	Content	Credit	Weightage
I	ITSM Foundations & ITIL Framework • Introduction to IT Service Management, Key ITSM Concepts, ITIL 4 Framework, Service Strategy Fundamentals, ITSM Governance and Compliance	1	25%
II	Service Design & Transition • Service Design Principles, Service Catalog Management, Service Level Management, Capacity Management, Availability Management, IT Service Continuity Management, Service Transition Processes	1	25%
III	Service Operation & Continual Improvement • Service Operation Principles, Service Desk Function, Incident Management, Problem Management, Request Fulfillment, Access Management, Event Management, Continual Service Improvement (CSI)	1	25%
IV	Advanced ITSM & Integration • ITSM Tools and Technologies, DevOps and ITSM Integration, Cloud Service Management, ITSM for Digital Transformation, ITSM Metrics and Analytics, ITSM Implementation Strategy, Emerging Trends in ITSM, Career Paths in ITSM	1	25%

Textbooks:

- "ITIL 4 Foundation: ITIL 4 Edition" *Axelos Official ITIL 4 Foundation guide*
- "Foundations of IT Service Management with ITIL 4" *Jan van Bon, Mike Pieper, Ruby Tjassing Comprehensive ITIL 4 references*

Reference Books:

- "The ITIL 4 Practitioner: Guiding Principles for ITSM Professionals" *Claire Agutter Practical implementation guide*
- "Implementing Service Quality based on ISO/IEC 20000" *Nicolas M. Dhuin Standards-based approach to ITSM*

- "The Phoenix Project: A Novel About IT, DevOps, and Helping Your Business Win"
  Gene Kim, Kevin Behr, George Spafford Business novel about IT operations and DevOps

Online Resources:

- Axelos Official ITIL Resources
- ITSM.tools (Online community and resources)
- ServiceNow Community
- NPTEL: "IT Service Management" courses
- Coursera: "IT Service Management" specializations
- LinkedIn Learning: ITIL and ITSM courses
- YouTube: ITSM Academy and Pink Elephant channels

COURSE CODE: BTIT705
COURSE NAME: BLOCK CHAIN TECHNOLOGY
Course Objective

- Understand blockchain fundamentals, architecture, and cryptographic foundations.
- Analyze consensus mechanisms, smart contracts, and distributed ledger technologies.
- Develop blockchain applications using Ethereum and Hyperledger platforms.
- Design decentralized applications (DApps) with Web3 integration.
- Evaluate blockchain use cases across industries and regulatory frameworks.
- Implement blockchain security measures and analyze emerging trends.

Course Outcomes:

CO1	Analyze blockchain architecture, cryptographic foundations, and consensus mechanisms.
CO2	Design and implement smart contracts using Solidity on Ethereum platform
CO3	Develop decentralized applications (DApps) with Web3.js integration
CO4	Implement permissioned blockchain solutions using Hyperledger frameworks

Unit	Content	Credit	Weightage
I	Blockchain Foundations & Cryptography • Introduction to Blockchain Technology, Cryptographic Foundations, Distributed Systems Concepts, Blockchain Architecture, Bitcoin Protocol Deep Dive, Alternative Consensus Approaches	1	25%

II	Smart Contracts & Ethereum Platform • Ethereum Fundamentals, Smart Contract Development, Development Tools & Frameworks, Token Standards, Decentralized Applications (DApps), Ethereum Improvement Proposals (EIPs), Layer 2 Scaling Solutions	1	25%
III	Enterprise Blockchain & Hyperledger • Enterprise Blockchain Requirements, Hyperledger Ecosystem, Hyperledger Fabric, Hyperledger Sawtooth, Hyperledger Besu, Other Enterprise Platforms, Blockchain Interoperability, Blockchain as a Service (BaaS)	1	25%
IV	Applications, Security & Emerging Trends • Blockchain Use Cases, Blockchain Security, Regulatory & Legal Aspects, Blockchain Analytics, Scalability Solutions, Decentralized Finance (DeFi), Emerging Trends, Blockchain Project Lifecycle, Career Paths in Blockchain	1	25%

Textbooks:

- "Mastering Blockchain: A deep dive into distributed ledgers, consensus protocols, smart contracts, DApps, cryptocurrencies, Ethereum, and more" *Imran Bashir* (3rd Edition) *Comprehensive blockchain technology reference*
- "Blockchain Basics: A Non-Technical Introduction in 25 Steps" *Daniel Drescher Excellent foundational book for beginners*

Reference Books:

- "Mastering Ethereum: Building Smart Contracts and DApps" *Andreas M. Antonopoulos & Gavin Wood Definitive Ethereum development guide*
- "Blockchain Revolution: How the Technology Behind Bitcoin and Other Cryptocurrencies is Changing the World" *Don Tapscott & Alex Tapscott*
  Business perspective on blockchain
- "The Bitcoin Standard: The Decentralized Alternative to Central Banking"
  Saifedean Ammous Economic perspective on cryptocurrency

Online Resources:

- Ethereum Official Documentation
- Hyperledger Documentation
- Bitcoin Whitepaper (Satoshi Nakamoto)
- NPTEL: "Blockchain and its Applications" courses
- Coursera: "Blockchain Specialization" (University at Buffalo)
- edX: "Blockchain Fundamentals" (Berkeley)
- YouTube: Ethereum Foundation, ConsenSys Academy
- GitHub: Open-source blockchain projects

Practical List:

- Session 1: Cryptography implementation (SHA-256, ECDSA)
- Session 2: Merkle tree implementation and verification

- Session 3: Basic blockchain implementation in Python
- Session 4: Simple consensus mechanism implementation
- Session 5: Ethereum development environment setup
- Session 6: Basic Solidity smart contract development
- Session 7: ERC-20 token implementation
- Session 8: ERC-721 NFT implementation
- Session 9: DApp development with Web3.js
- Session 10: Hyperledger Fabric network setup
- Session 11: Chaincode development in Go
- Session 12: Private data implementation in Fabric

MK UNIVERSITY

PATAN, GUJARAT

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

SEMESTER-VIII

COURSE CODE: BTIT801
COURSE NAME: ENTERPRISE SOLUTIONS

Course Objective

- Understand enterprise solution architecture, integration patterns, and business process modelling.
- Design and implement Enterprise Resource Planning (ERP), Customer Relationship Management (CRM), and Supply Chain Management (SCM) solutions.
- Analyze enterprise data management, business intelligence, and analytics solutions.
- Develop enterprise application integration using middleware and API management.
- Evaluate cloud-based enterprise solutions and digital transformation strategies.
- Design enterprise security, governance, and compliance frameworks.

Course Outcomes:

CO1	Analyze enterprise solution requirements and design integratedarchitectures.
CO2	Implement ERP, CRM, and SCM solutions for business processes
CO3	Design and develop enterprise data management and business intelligence solutions
CO4	Implement enterprise application integration using middleware platforms

Unit	Content	Credit	Weightage
I	Enterprise Solution Architecture & Integration • Introduction to Enterprise Solutions, Enterprise Architecture Frameworks, Business Process Management (BPM), Enterprise Integration Patterns, Service-Oriented Architecture (SOA), Microservices Architecture, Middleware Technologies	1	25%
II	Core Enterprise Solutions (ERP, CRM, SCM) • Enterprise Resource Planning (ERP), Customer Relationship Management (CRM), Supply Chain Management (SCM), Human Capital Management (HCM), Enterprise Content Management (ECM)	1	25%
III	Enterprise Data & Analytics Solutions • Enterprise Data Management, Business Intelligence & Analytics, Big Data Solutions for Enterprises, Artificial Intelligence in Enterprises, Enterprise Search Solutions, Digital Experience Platforms (DXP)	1	25%

IV	Enterprise Solution Implementation & Management • Enterprise Solution Implementation, Cloud Enterprise Solutions, Enterprise Security & Governance, Enterprise Solution Operations, Integration Platforms as a Service (iPaaS), Emerging Trends, Career Paths in Enterprise Solutions	1	25%

Textbooks:

- "Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions" *Gregor Hohpe & Bobby Woolf Definitive guide to enterprise integration patterns*
- "The Architecture of Enterprise: Creating Business Value through Enterprise Architecture" *John Gøtze & Anders Jensen-Waud Modern enterprise architecture perspective*

Reference Books:

- "ERP: Making It Happen: The Implementers' Guide to Success with Enterprise Resource Planning" *Thomas F. Wallace & Michael H. Kremzar Practical ERP implementation guide*
- "Business Process Management: Concepts, Languages, Architectures" *Mathias Weske* (3rd Edition) *Comprehensive BPM reference*
- "The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modelling" *Ralph Kimball & Margy Ross* (3rd Edition) *Essential data warehousing guide*

Online Resources:

- SAP Community Network
- Salesforce Trailhead (Free learning platform)
- Microsoft Learn (Dynamics 365, Azure)
- Oracle University
- Gartner Research (Enterprise technology trends)
- Forrester Wave Reports
- NPTEL: "Enterprise Systems" courses
- Coursera: "Enterprise Architecture" specializations

Practical List:

- Session 1: Enterprise architecture modelling with ArchiMate
- Session 2: Business process modelling with BPMN 2.0
- Session 3: Process automation with Camunda
- Session 4: Integration pattern implementation
- Session 5: ERP configuration and customization (Odoo/SAP)
- Session 6: CRM implementation (Salesforce/HubSpot)
- Session 7: SCM process implementation
- Session 8: Enterprise data warehouse design
- Session 9: Business Intelligence dashboard development
- Session 10: Enterprise application integration with MuleSoft
- Session 11: API management implementation
- Session 12: Cloud migration strategy development

MK UNIVERSITY

PATAN, GUJARAT

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

COURSE CODE: BTIT802
COURSE NAME: NETWORK IMPLEMENTATION

Course Objective

- Design and implement enterprise network architectures with various topologies.
- Configure and troubleshoot routing protocols, switching technologies, and network services.
- Implement network security measures including firewalls, VPNs, and access control.
- Deploy wireless networks with enterprise-grade security and management.
- Manage network infrastructure using monitoring, automation, and troubleshooting tools.
- Plan and execute network migrations, upgrades, and disaster recovery strategies.

Course Outcomes:

CO1	Design enterprise network architectures following best practices and standards.
CO2	Implement and configure routing protocols and switching technologies
CO3	Deploy network security solutions including firewalls, VPNs, and access control
CO4	Implement and manage enterprise wireless networks

Unit	Content	Credit	Weightage
I	Network Design & Architecture • Network Design Fundamentals, Network Topologies, IP Addressing and Subnetting, Network Documentation, Cabling Infrastructure, Network Devices Selection	1	25%
II	Routing & Switching Implementation • Switch Configuration and Management, Router Configuration, Inter-VLAN Routing, Network Address Translation (NAT), First Hop Redundancy Protocols (FHRP), Quality of Service (QoS) Implementation	1	25%
III	Network Services & Security Implementation • Network Services Implementation, Network Security Implementation, Intrusion Prevention Systems (IPS), Web Security Appliances, Email Security Implementation	1	25%
IV	Wireless, Management & Advanced Implementation • Enterprise Wireless Network Implementation, Network Management Implementation, Network Automation, High Availability Implementation, Data	1	25%

	Center Network Implementation, Cloud Network Implementation. Network Migration and Upgrade Planning, Disaster Recovery Implementation		

Textbooks:

- "Network Implementation: A Practical Guide" *James Boney Practical network implementation guide*
- "CCNA 200-301 Official Cert Guide" *Wendell Odom Comprehensive networking implementation reference*

Reference Books:

- "Network Warrior: Everything You Need to Know That Wasn't on the CCNA Exam" *Gary A. Donahue Real-world network implementation scenarios*
- "The Practice of System and Network Administration" *Thomas A. Limoncelli, Christina J. Hogan, Strata R. Chalup Excellent for network operations and management*
- "IPv6 Essentials" *Silvia Hagen Comprehensive IPv6 implementation guide*

Online Resources:

- Cisco Networking Academy courses
- Juniper Learning Portal
- RFC Documents (IETF standards)
- NPTEL: "Computer Networks" and "Network Security" courses
- Coursera: "Networking" specializations
- YouTube: NetworkChuck, David Bombal, Keith Barker
- Packet Pushers podcast and blog

Practical List:

- Session 1: Network topology design and documentation
- Session 2: VLAN implementation and inter-VLAN routing
- Session 3: Spanning Tree Protocol configuration and optimization
- Session 4: EtherChannel/Link Aggregation configuration
- Session 5: Switch security features implementation
- Session 6: OSPF routing protocol implementation
- Session 7: DHCP and DNS server configuration
- Session 8: NAT and PAT configuration
- Session 9: QoS implementation for voice and video
- Session 10: Firewall and VPN configuration
- Session 11: Wireless network implementation with security
- Session 12: Network monitoring and management setup

COURSE CODE: BTIT803
COURSE NAME: IT INFRASTRUCTURE
Course Objective

- Understand IT infrastructure components, architecture, and design principles.
- Design and implement server infrastructure with virtualization and cloud integration.
- Manage storage systems, backup solutions, and disaster recovery strategies.
- Configure network infrastructure with security and performance optimization.
- Implement monitoring, automation, and infrastructure as code practices.
- Evaluate emerging infrastructure technologies and plan infrastructure lifecycle management.

Course Outcomes:

CO1	Analyze IT infrastructure requirements and design enterprise infrastructure solutions.
CO2	Implement and manage server infrastructure with virtualization technologies
CO3	Design and manage storage systems, backup, and disaster recovery solutions
CO4	Configure network infrastructure with security and performance considerations

Unit	Content	Credit	Weightage
I	IT Infrastructure Foundations & Architecture • Introduction to IT Infrastructure, IT Infrastructure Architecture, Data Center Design, Server Hardware, Power and Cooling Management, Infrastructure Documentation	1	25%
II	Server & Virtualization Infrastructure • Server Operating Systems, Virtualization Technologies, Containerization, Server Management, High Availability and Clustering, Infrastructure as Code (IaC)	1	25%
III	Storage, Backup & Network Infrastructure • Storage Systems, Backup and Recovery, Network Infrastructure, Unified Communications Infrastructure	1	25%
IV	Cloud, Management & Emerging Infrastructure • Cloud Infrastructure, Infrastructure Monitoring and Management, Infrastructure Security, Infrastructure Automation, Emerging Infrastructure Technologies, Infrastructure Lifecycle Management, Infrastructure Operations	1	25%

Textbooks:

- "The Practice of Cloud System Administration: DevOps and SRE Practices for Web Services" *Thomas A. Limoncelli, Strata R. Chalup, Christina J. Hogan Modern infrastructure management practices*
- "Data Center Handbook: Plan, Design, Build, and Operations of a Smart Data Center" *Hwaiyu Geng Comprehensive data center infrastructure guide*

Reference Books:

- "The Practice of System and Network Administration" *Thomas A. Limoncelli, Christina J. Hogan, Strata R. Chalup* (3rd Edition) *Classic infrastructure operations reference*

- "Cloud Native Infrastructure: Patterns for Scalable Infrastructure and Applications in a Dynamic Environment" *Justin Garrison & Kris Nova Modern cloud infrastructure patterns*
- "Virtualization Essentials" *Matthew Portnoy* (2nd Edition) *Excellent virtualization fundamentals*

Online Resources:

- AWS/Azure/GCP Documentation
- VMware Learning Zone
- Red Hat Learning Subscription
- NPTEL: "Cloud Computing" and "Virtualization" courses
- Coursera: "IT Infrastructure" specializations
- YouTube: Tech Channel, CBT Nuggets
- GitHub: Infrastructure as Code repositories