Lumbini Technological University

Institute of Engineering and Information Technology

Banke, Lumbini Province, Nepal

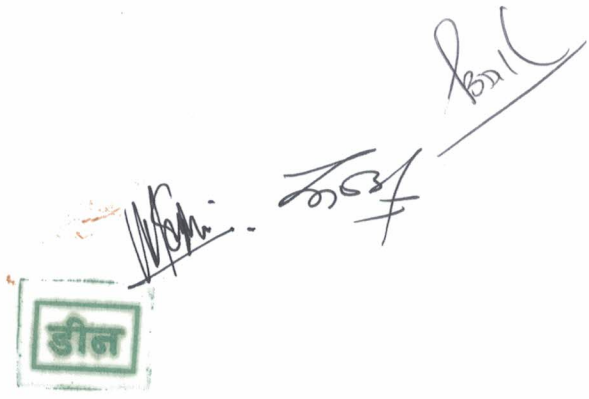
B. Tech. in Information Technology

(B.Tech. in IT)

Program Code: 101

Syllabus

Batch 2023



डीन

B. Tech. in Information Technology

1. Introduction

Lumbini Technological University (LTU) was established on July 2, 2022 (Ashar 18, 2079 BS) by the Act of Province Legislature with the goal of advancing higher education in the field of information technology, engineering, agricultural and forestry, and tourism, among others, with a particular focus on the application and development of technology and innovation.

LTU's mission is to promote and offer comprehensive programs at undergraduate and graduate levels that produce graduates who have both disciplinary expertise and the ability to handle real-world problems by combining theoretical knowledge with practical application along with exposure visits to world-renowned technological schools. LTU recognizes that academic programs are not the only way to foster a rich learning experience. To this end, it also places a strong emphasis on research and continuous education programs by supporting and encouraging students and faculty to engage in rigorous research activities, thus fostering a culture of innovation and creativity that benefits both students and faculty alike.

B. Tech. in Information Technology (B. Tech. in IT) Program

Technology continues to play an increasingly significant role in shaping our world with traditional jobs being replaced by tech-oriented ones, which demands competent professionals. IT education drives technological progress, employment opportunities, economic growth, digital inclusion, innovation, improved education systems, and enhanced government services and governance.

The B. Tech. in Information Technology program at Lumbini Technological University is designed to produce skilled IT professionals who can contribute to the advancement of technology and address the evolving needs of the digital era. Through a blend of theoretical knowledge, practical experience, and industry exposure, students are prepared to embark on successful and fulfilling careers in the dynamic and ever-growing field of Information Technology.

2. Duration and Total Credit

The B. Tech. in IT program spans over four years, divided into eight semesters. The total credit of courses is 121, where 1 credit is equivalent to 16 hours of lecture.

3. Eligibility for Admission

Applicants must have passed Class 12 from National Examinations Board (NEB) or equivalent with minimum D+ Grade / Second Division (Grade D in case of A-level). They must pass Lumbini Technological University Entrance Test for admission.

4. Curriculum

The curriculum of B. Tech. in IT program is thoughtfully structured to cover a wide range of topics, including computer programming, software development, database management, networking, cybersecurity, web development, and IT project management, with the following objectives:

- Provide students with a solid foundation in the fundamental concepts and principles of computer science, computer engineering, and IT.
- Equip students with practical skills, tools, and techniques necessary for solving real-world problems in these fields.
- Foster critical thinking, problem-solving, and analytical skills among students.

- Provide students with opportunities to work in teams and collaborate on projects.
- Encourage innovation, creativity, and entrepreneurship among students.
- Foster a sense of ethics, responsibility, and social awareness among students.
- Prepare students for successful careers or further studies in related fields.

In order to achieve these objectives, the curriculum comprises the following components:

- **Core Courses:** These courses provide a solid foundation in IT. Students are also required to complete courses in mathematics, natural sciences, social sciences, and humanities to develop critical thinking and communication skills.
- **Electives:** These courses provide flexibility to the students to choose elective courses that align with their interests and career goals.
- **Project-based Learning:** Students are required to work on a series of projects throughout the program. These projects provide students with opportunities to apply their knowledge and skills to real-world problems, work in teams, and develop communication and collaboration skills.
- **Industry Engagement:** The program provides opportunities for students to engage with industries (including government agencies) through internships, co-op programs, and guest lectures by industry professionals.
- **Research and Innovation:** Students are encouraged for research activities, innovation challenges, and participation in IT-related competitions to foster creativity and entrepreneurship among students.

The course structure is presented in Table 1.

5. Evaluation

Students are evaluated based on their performance in both internal assessments and semester exams. Final evaluation employs absolute grading according to the grade scale outlined in the Student Registration and Examination Guidelines, BS 2080.

The model question for semester examination is presented in Annex A.

6. Syllabus

The detailed syllabus of courses is presented in Annex B.

Table 1: Course framework of B. Tech. in IT program

First Year			
Semester I	**Credit**	**Semester II**	**Credit**
BIT101: Mathematics I	3	BIT151: Mathematics II	3
BIT102: English Communication	3	BIT152: OOP in Java	3
BIT103: Foundations of Information Technology	3	BIT153: Microprocessor & Computer Architecture	3
BIT104: Digital Logic	3	BIT154: UI/UX Design	3
BIT105: C Programming	3	BIT155: Operating System	3
Total	**15**	**Total**	**15**
Second Year			
Semester III	**Credit**	**Semester IV**	**Credit**
BIT201: Statistics	3	BIT251: Research Methodology	3
BIT202: Database Management System	3	BIT252: Python Programming	3
BIT203: Data Communication and Networks	3	BIT253: Management Information System	3
BIT204: Web Technology I	3	BIT254: Web Technology II	3
BIT205: Data Structure and Algorithm	3	BIT255: System Analysis and Design	3
		BIT290: Project I	2
Total	**15**	**Total**	**17**
Third Year			
Semester V	**Credit**	**Semester VI**	**Credit**
BIT301: Artificial Intelligence	3	BIT351: Digital Governance	3
BIT302: Software Engineering	3	BIT352: Mobile Application Development	3
BIT303: E-Commerce	3	BIT353: Machine Learning	3
BIT304: Data Science	3	BIT354: Cyber Security & Ethical Hacking	3
BIT305: Information Security	3	Elective I	3
Non-credit: DevOps	-		
Total	**15**	**Total**	**15**
Fourth Year			
Semester VII	**Credit**	**Semester VIII**	**Credit**
BIT 401: IT Project Management	3	BIT 451: IT Entrepreneurship	3
BIT 402: Embedded Systems & Internet of Things (IoT)	3	BIT 452: Data Warehouse & Data Mining	3
BIT 403: Blockchain Technology	3	BIT 499: Internship	6
Elective II	3		
BIT 445: Capstone Project	5		
Total	**17**	**Total**	**12**
		Total Credit	**121**

Note: The course code of courses offered in B. Tech. in IT starts with 'BIT'.

Annex A: Model Question Format

	Lumbini Technological University **Institute of Engineering and Information Technology** **Examination Division** Banke, Lumbini Province Nepal **Exam Year/Month**				
Program:				**Full Marks:**	40
Course Title:				**Pass Marks:**	20
Course Code:		**Year/Semester:**		**Time:**	2 hrs

Candidates are required to give their answers in their own words as far as possible. The figures in the margin indicate full marks.

Group - A

Short Answer Questions

Attempt Any FIVE questions [5x3=15]

1. Question 1
2. Question 2
3. Question 3
4. Question 4
5. Question 5
6. Question 6

Group - B

Long Answer Questions

Attempt Any FIVE questions [5x5=25]

7. Question 7
8. Question 8
9. Question 9
10. Question 10
11. Question 11
12. Question 12

Annex B: Syllabus

Year I Semester I

Mathematics I

Course Code	BIT101	Year/Semester	I/I
Credit Weightage	3 (48 hrs)	Lecture	3 hrs/wk
		Tutorial	2 hrs/wk
		Practical	0
		Total	80 hrs

1. Course Description

This course is designed to provide students with a comprehensive understanding basic concept and its uses to solving the problems of limits, continuity, differentiation and integration with their applications, vector algebra, analytic geometry and vector-valued functions, multivariable calculus.

2. Course Objectives

The course aims to achieve the following objectives:

- Understand and formulate real world problems into mathematical statements.
- Develop solutions to mathematical problems at the level appropriate to the course.
- Describe or demonstrate mathematical solutions either numerically or graphically.

3. Course Details

3.1. Lecture (48 hrs)

Unit 1: Logic, Set theorem and Real Number (7 hrs)

- Elementary Logic or Proposition, Logical Connectives, Law of Logic
- Concept, Notation and Specification of Sets, Type of Sets, Operations on Sets (Union, Intersection, Difference, Complement) and their Venn-diagrams, Laws of Algebra of Sets (without proof), Cardinal Number of Set.
- Real Number System, Intervals, Absolute Value of Real Number

Unit 2: Relation, Functions and Graphs (9 hrs)

- Order Pairs, Cartesian Product, Relation, Domain and Range of a Relation, Inverse Relation
- Type of Relations: Reflective, Symmetric, Transitive, and Equivalence Relations
- Function, Domain and Range of a Function, Inverse Function, Special Function (Identity, Constant), Algebraic (Linear, Quadratic, Cubic), Trigonometric and their Graphs, Exponential and Logarithmic Functions, Composite Function

Unit 3: Analytical Geometry (6 hrs)

- Circle: Circle, Equation and Graphs, Condition of Tangency, Tangent and Normal
- Parabola, Ellipse, Hyperbola, Related Terms and Graphs of Conic Section, Classifying the Conic Section by Eccentricity

Unit 4: Vector Algebra (4 hrs)

- Vectors in Space, Algebra of Vectors in Space, Length
- Distance between two Points, Scalar Product and Cross Product of two and three Vectors with their Geometrical Interpretation, Projection of a Vector

Unit 5: Matrix and Linear algebra (9 hrs)

- Matrices and its Types, Equality of Matrices, Algebra of Matrices,
- Determinant, Singular and Non-Singular Matrix, Rank of a Matrix
- Linear System of Equations, Solution of Linear System of Equations by Gauss Elimination, Eigenvalues and Eigenvectors, Eigenspace

Unit 6: Vector Space (6 hrs)

- Vector Space and Subspace, Linear Independence, Basis of a Vector Space

- Linear Transformation
- Orthogonal Matrix and Transformation by Orthogonal Matrix

Unit 7: Basic Graph Theory **(7 hrs)**

- Directed graphs
- Shortest Path Problems
- Shortest Spanning Trees

3.2. List of Tutorials

The following tutorial activities shall be conducted under the supervision of the subject teacher.

- Calculating sets using different operations
- Calculating sets using Venn diagrams
- Problems based on algebra of sets
- Evaluating number system, finding intervals and value based on absolute value
- Problems using logic
- Problems with Cartesian product, relation
- Finding domain and range of a function and relation
- Sketch the graphs of various functions
- Calculating different functions
- Problems based on circle, parabola, ellipse, hyperbola
- Finding the value using Dot and vector product between 2 and 3 vectors
- Finding area by scalar product of 2 vectors and volume by scalar product of 3 vectors
- Finding the projection vector
- Operating different matrices
- Evaluating determinant
- Finding rank of matrix
- Solving system of linear equations by using rank of a matrix
- Finding eigenvalues, eigenvectors and eigenspace of a matrix
- Finding basis, test the orthogonal matrix

4. Evaluation Scheme

Evaluation of students' performance is divided into two main parts: (1) Internal evaluation and (2) Semester examination, with breakdown as follows.

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal evaluation	Class attendance and performance	5	40	20
	Assignment	15		
	Seminar/project/presentation	20		
	Practical/lab examination	-	-	-
	Internal examination	20	20	10
	Total of internal evaluation		**60**	**30**
Semester examination	**Total of semester examination**		**40**	**20**

5. Books

- David C. Lay, Linear Algebra and Its Applications (4th Edition), Addison Wesley, Pearson.
- Thomas, G. & Finney, R. Calculus and Analytical Geometry. New Delhi, Narosa Publishing House.
- Kreyszig, E. Advanced Engineering Mathematics, New Delhi, John Wiley and Sons Inc.

English Communication

Course Code	BIT102	Year/Semester	I/I
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	2 hrs/wk
		Practical	0
		Total	80 hrs

1. Course Description

This course is designed to equip students with a wide range of techniques for effectively utilizing English in both academic and communicative contexts.

2. Course Objectives

The course aims to achieve the following objectives:

- Acquaint the students with necessary grammar and use them to construct sensible sentences.
- Enable the students to answer the questions given after the comprehension exercises.
- Familiarize the students with the various genres of literature and analyze them.
- Enable the students with the process of writing.
- Enable the students to produce different texts and genres of writing.
- Develop students' ability to speak fluently and interactively.
- Make the students capable of producing professional writing such as proposals, and reports.

3. Course Details

3.1. Lecture (48 hrs)

Unit 1: Reading Comprehension (12 hrs)

The Bet by Anton Chekov (Story); The Three Dancing Goats by Anonymous writer (Story); The Gift of the Magi by O'Henry (Story); The Road Not Taken by Robert Frost (Poem); Letter from Foreign Grave by D. B. Gurung (Poem); College Teachers by R.L.S. Malky and Mary K Ruether, Eds. (Essay); Humility by Noah Harari (Essay); I want a Wife by Judy Brady (Essay); 9 Thirty-Eight Who Saw the Murder Didn't Call the Police by Martin Gangsberg (Essay); Connotation and types of Innovation by Jin Chen & Ximing Yin (Essay).

Unit 2: Sentence Construction and Mechanics in Writing (6 hrs)

Use of tenses; active and passive voice; subject-verb agreement; conditional sentences; prepositions; mechanics in writing: punctuations, spelling, capitalization, italics, abbreviations, etc.

Unit 3: Writing Skill (20 hrs)

General writing: job application and cv, greeting card and invitation letter, email and SMS, brochures, posters, blogs, business letter, notice and minute, memoranda, essay; technical writing: proposal writing/project writing; types of formal reports: progress report, feasibility report, empirical report, research report, technical report; report writing: report writing: memo report, letter report, project/field report.

Unit 4: Oral Communication (10 hrs)

greeting and introducing; giving instruction/direction; describing thanking and accepting an apology; expressing individual views; group/round table discussions; conduct meeting; interview (techniques, body language, dos and don'ts); talk delivery; presenting a brief report.

4. Evaluation Scheme

Evaluation of students' performance is divided into two main parts: (1) Internal evaluation and (2) Semester examination, with breakdown as follows.

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal evaluation	Class attendance and performance	5	40	20
	Assignment	15		
	Seminar/project/presentation	20		
	Practical/lab examination	-	-	-
	Internal examination	20	20	10
	Total of internal evaluation		**60**	**30**
Semester examination	**Total of semester examination**		**40**	**20**

5. Books

5.1. Text Book

- Matreyek, W. (1983). Communicating in English. London: Pergamon Press.
- Davis, F. & Rimmer, W. (2013). Active Grammar 1. Cambridge: Cambridge University Press.
- Thompson, A.J. & Martinet, A. V. (1986). A Practical English grammar. London: Oxford University Press.
- Pilai, S. (2016). Spoken English for my World. Oxford: Oxford University Press.

5.2. Reference Book

- Futherford, A. J. (2001). Basic Communication Skills for Technology. Pearson: Pearson Education.
- Mandel, S. R. & Kirszner, L. G. (eds.)(2009). Patterns for College Writing. New York: St. Martin's.

Foundation of Information Technology

Course Code	BIT103	Year/Semester	I/I
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hr/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

This course covers the basic concepts of computers and information technology including introduction, hardware, software, memory, input/output, data representation, database, networks and data communication, Internet, multimedia, and computer security.

2. Course Objectives

The course aims to achieve the following objectives:

- Identify the components of a computer system
- Familiarize with hardware/software components
- Knowledge of database management systems
- Internet, networking and mobile computing and internet security.
- Apply different tools and technique to create documents using office package software
- Use of CUI and GUI Interface to operate computer applications
- Explore concept of emerging technologies.

3. Course Details

3.1. Lecture (48 hrs)

Unit 1: Introduction to Computer System (3 hrs)

- Introduction to Computer
- Characteristics of Computer
- History of computer
- Generation of Computer
- Classification of Computer: Size, Work, Brand
- Anatomy of a Digital Computer
- Scope and Applications of Computer

Unit 2: Computer Hardware (6 hrs)

- Introduction to hardware
- Central Processing Unit: Definition and Role of CPU, ALU, CU, Register and their functions
- Memory Unit: Memory Hierarchy, memory representation
- Memory & Its Classification: Primary memory (RAM, ROM), Secondary memory (Magnetic Disk, Optical Disk, Magnetic Tape)
- Instruction Format: Definition, Components
- Instruction Set: Definition, Types of Instruction Set (CISC, RISC)
- Instruction Cycle: Fetch, Decode, Execute, Role of Program Counter and Instruction Register
- Interconnecting the Units of a Computer: Bus Architecture, Address Bus, Data Bus, Control Bus
- Inside the computer cabinet
- Microprocessor: Definition and Function
- Input devices: Definition and Types of Devices
- Output Devices: Definition and Types of Devices

Unit 3 Computer Software **(5 hrs)**

- Introduction to computer software
- Categories of Software
- System software: Introduction, Types (Operating System: Introduction and Objectives of Operating System, Types of Operating System, Functions of Operating System), Diver Driver, Firmware, Utility Software, Translator
- Application Software: Introduction, Types (General Purpose, Customized Software), Applications
- Software Acquisition

Unit 4: Data Representation **(4 hrs)**

- Introduction to Number System and their conversion
- Binary Arithmetic: Addition, Subtraction, Multiplication, Division
- Complement Methods: 1's Complement, 2's Complement
- Binary Data Representation: Signed and Unsigned Numbers
- Binary Coding Schemes: BCD, Gray Code, Excess-3, ASCII
- Logic Gates: OR, AND, NOT, NAND, NOR, XOR, XNOR

Unit 5: Fundamentals of Database Management System **(5 hrs)**

- Introduction to DBMS: Definition and Purpose, Key Components, Advantage of DBMS Over Flat File System, Applications of DBMS
- Database Models: Relational Database Model, Hierarchical Database Model, Network Database Model, Object-oriented Database Model, NoSQL Database Model
- Database Language: DDL, DML, DTL, DCL
- Data Security: Definition and its importance
- Data Warehouse: Definition and Purpose, Key Features
- Data Mining: Definition and Objectives, Steps, Application
- Database Users

Unit 6: Data Communication and Computer Network **(6 hrs)**

- Introduction to communication system
- Mode of Communication: Simplex, Half Duplex, Full Duplex
- Computer Network: Introduction, Importance, Basic Components
- Network Types: PAN, LAN, MAN, WAN
- Network Topologies: Bus, Ring, Star, Tree, Mesh, Hybrid
- Transmission Media: Guided Media (Twisted-Pair, Co-axial, Fiber Optics), Unguided (Microwaves, Radiowaves, Infrared)
- Network Devices: Modem, Router, Repeater, Hub, Switch, Bridge, Gateway, NIC
- TCP/IP & OSI Reference Model
- Communication Protocols: HTTP, TCP/IP, FTP, SMTP, Telnet, Gopher, POP3
- Centralized vs. Distributed System,
- Wireless Networking (Wireless PAN, Wireless LAN, Wireless MAN, Wireless WAN)

Unit 7: The Internet and Internet Services **(6 hrs)**

- Introduction: Overview, History of Internet, Advantage and Disadvantage
- Internetworking Protocol: TCP/IP
- The Internet Architecture: Client, ISP, Regional ISP, and Backbone
- Managing the Internet: Governing bodies of the Internet
- Connecting to Internet: How to connect to the Internet?

- Internet Connections: Dial-up Connection, Broadband Connection, DSL, Cable Modem, ISDN, Cellular, Wireless connection
- Internet Address: IP address and domain names
- WWW and Its Evolution
- Internet Services: Communication Services, Information Retrieval Services, Web Service, WWW
- Uses of Internet: Different uses of Internet
- Architecture of Web
- Uniform Resource Locator (URL), Browsers and It's Feature, Search Engine
- Web Servers: Apache, IIS, Proxy Server

Unit 8: Computer Security **(5 hrs)**

- Introduction: Introduction to computer security
- Security Threat and Security Attack: What is security threat? Security attack and its types.
- Malicious Software: Virus, worm, Trojan horse
- Security Services: Confidentiality, integrity, authentication, nonrepudiation
- Security Mechanism: Cryptography, Digital Signature, Firewall, User Identification and
- Authentication, Intrusion Detection Systems
- Cyber Security: Introduction, Types, Importance
- Security Management: Overview and Importance

Unit 9: Emerging Technologies **(8 hrs)**

- Cloud Computing: Definition and Characteristics, Cloud Service Model (IaaS, PaaS, SaaS), Deployment model (Public, Private, Community, Hybrid)
- Mobile Computing: Introduction and its Concept, Application
- Quantum computing: Introduction, Quantum Bits and Applications
- E-governance: Definition and objectives, Types, Advantage and Disadvantage
- E-commerce: Introduction, Types, Pros and Cons
- E-Learning: Introduction: Introduction, methods
- Ubiquitous Learning: Definition and characteristics, Challenges and Opportunity
- E-Banking: Online Banking, Services and security measures
- Mobile Banking: Services and Application, Security consideration
- Crypto Currency: Introduction, working Mechanism, Risk
- GIS: Basis of GIS, Application
- Artificial Intelligence: overview, Applications ethics and bias
- Machine Learning: Fundamental, Types, Features
- Deep learning: Introduction, application
- Robotics: Introduction, Types, Applications
- Virtual Reality, Augmented Reality, Ambient Reality: Introduction, Future Trends, Applications
- Internet of Things: Introduction, Architecture and Components
- Blockchain Technology: Overview, Applications

3.2. Laboratory Work

After completing this course, students should have practical knowledge of different hardware components of computer, operating systems, Word Processors, Spreadsheets, Presentation Graphics, Database Management Systems, and Internet and its services. The laboratory works include:

(1) Basic of Computer and Network

- Basic building block of PC, hardware and peripheral introduction

- Assemble and disassemble of computer
- Learning about network cable and crimping different cable types

(2) Learn CUI and GUI Based Operating System

- Install different types of operating system
- Learning command line interface such as Command line/terminal
- Learn MS-DOS Internal command and External command

(3) Learn word processing software

- Concept of word processing, Introduction of MS Word, Features, Keyboard Shortcuts, Editing & Formatting documents, Table, Advance features of MS Word, Mail Merge, Macros

(4) Learn spreadsheets software

- Worksheet basic, working with formulas and cell referencing, Auto sum, Formatting to worksheet, Previewing and Printing worksheet, Graphs and Charts, Functions, Macros

(5) Learn presentation software

- Introduction, Use of Standard Formatting toolbar, drawing toolbar, editing slides, changing templates, Slide Layouts, inserting clipart & Pictures into slide, Slide Transitions, Animation, Inserting sound and movies into slides

(6) Email, Internet and online tools for office management

- Learn different email system, configuring email into personal computer, using internet-based tools for office management

4. Evaluation Scheme

Evaluation of students' performance is divided into two main parts: (1) Internal evaluation and (2) Semester examination, with breakdown as follows.

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal evaluation	Class attendance and performance	5	20	10
	Assignment	10		
	Seminar/project/presentation	5		
	Practical/lab examination	20	20	10
	Internal examination	20	20	10
	Total of internal evaluation		**60**	**30**
Semester examination	**Total of semester examination**		**40**	**20**

5. Books

5.1. Text Books

- A. Goel, "Computer Fundamentals." Pearson Education India.
- P. Norton, "Introduction to Computers." McGraw-Hill Education

5.2. Reference Books

- E. Balagurusamy, "Computing Fundamentals." McGraw-Hill Education.
- P. Kumar, A. Tomar, R. Sharmila, "Emerging Technologies in Computing: Theory, Practice, and Advances." CRC Press.

Digital Logic

Course Code	BIT104	Year/Semester	I/I
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hr/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

This course is designed to provide students with a comprehensive understanding of basic principles of digital logic design, its implementation and applications.

2. Course Objectives

The course aims to achieve the following objectives:

- Introduce number system and their conversion
- Familiarize logic gates
- Identify and solve the Boolean algebraic expressions
- Design combinational circuits
- Design sequential circuits
- Design clocked sequential circuits

3. Course Details

3.1 Lecture (48 hrs)

Unit 1: Number Systems and Binary Code (6 hrs)

Positional Number Systems: Binary, Octal, Decimal and Hexadecimal and their conversion; Radix and diminished radix complement method Representation of Signed and Unsigned Number , Floating Point Representation, Binary Codes: Alphanumeric (ASCII, EBCDIC), BCD, Weighted Binary Codes, Non-Weighted Binary Codes (Excess-3 and Gray Codes), Error Detecting Codes: Parity

Unit 2: Boolean Algebra and Logic Gates (8 hrs)

Logic Gates: AND, OR, NOT, NAND, NOR, X-OR, X-NOR, Boolean Algebra: Postulates of Boolean Algebra, Associativity, Inverse, Closure, Commutativity, and Distributives, Basic Theorems and Properties of Boolean Algebra (DeMorgan's Law, Absorption Law, Common Identity Lay), and Principle of Duality, Boolean Variables and Constants, Algebraic Simplification ,Canonical and Standard Forms: Canonical SOP, Canonical POS, Standard SOP, Standard POS, Maxterms, Minterms and Conversion between them, Realization of Boolean Expression using NAND and NOR Gate only ,IC Families

Unit 3: Simplification of Boolean Functions (6 hrs)

K-Map Method: 2, 3, and 4 variable K-Maps, The Tabulation Method – Detection and Selection of Prime Implicants, Product of Sum (POS) and Sum of Product (SOP) Simplification using K-Map, Don't Care Conditions.

Unit 4: Combinational Logic Circuits (12 hrs)

Adders: Half Adder, Full Adder; Subtractors: Half Subtractor, Full Subtractor; Binary Parallel Adders (2-bit, 4-bit, N-bit), BCD Adder; Encoders, Decoders and Priority Encoders; Multiplexers (2:1, 4:1, 8:1, 16:1) and De-Multiplexers (1:2, 1:4, 1:8, 1:16); Magnitude Comparator, Parity Generator and Checker; Combinational Circuit Analysis Procedures, Combination Circuit Realization using Universal Gates; Use of Multiplexers and Decoders for Combinational Logic Design; Introduction to ROM, PLA and PAL and their use in Combination Logic Circuits.

Unit 5: Sequential Logic Circuits (10 hrs)

Flip-flops: Latch, RS Flip-flop, JK Flip-flop, JK Master-Slave Flip-flop, D Flip-flop and T Flip-flop; Triggering of Flip-flop; Flip-flop Excitation Tables; Asynchronous vs Synchronous Sequential Circuits; Analysis of Sequential Circuit using State Diagrams

Unit 6: Registers and Counters **(6 hrs)**

Registers as basic Memory Blocks; Shift Registers; Ripple (Asynchronous) Counters, Synchronous Counters; Introduction to Memory Units, Memory Address Registers and Memory Buffer Registers

3.2 List of Tutorials

- Number system conversions
- Binary code conversions
- Circuit design using basic gates
- Realization of circuit with universal gates
- Simplification of Boolean function using K-map
- Combinational circuit design (Magnitude comparator, Parity generator and checker, Combinational circuit analysis)
- Counter design (Asynchronous Up and Down counters, Synchronous Up and Down counters)
- Analysis of sequential circuit

3.3 Laboratory Work

- Implementation of Logic Gates.
- Realization of Universal Gates.
- Converting Boolean Expression into Logic Circuits.
- Implementation of Adder and Subtractor Circuits.
- Implementation of Binary Parallel Adder.
- Design and Implementation of Code Converter Circuits.
- Design and Implementation of Selector (Mux/DeMux) Circuits.
- Design and Implementation of Magnitude Comparator Circuits.
- Design and Implementation of Parity Generator and Checker Circuits.
- Implement PLA and PAL Circuits.
- Implementation of Flip-flops.
- Design and Implementation of Master-Slave Flip-flop.
- Design and Implementation of Shift Registers.
- Design Synchronous Counters (MOD-4, MOD-8, MOD-10).
- Design Synchronous Counters (Binary Counter, UP/Down Counter, BCD).
- Design a Clocked Sequential Circuit.

4. Evaluation Scheme

Evaluation of students' performance is divided into two main parts: (1) Internal evaluation and (2) Semester examination, with breakdown as follows.

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal evaluation	Class attendance and performance	5	20	10
	Assignment	10		
	Seminar/project/presentation	5		
	Practical/lab examination	20	20	10
	Internal examination	20	20	10
	Total of internal evaluation		**60**	**30**
Semester examination	**Total of semester examination**		**40**	**20**

5. Books

- M. Morris Mano (2016), Digital Logic and Computer Design (1st Edition), Pearson
- Thomas L. Floyd (2015), Digital Fundamentals (11th Edition), Pearson
- A. Saha & N. Manna, Digital Principles and Logic Design, Infinity Science Press
- John Patrick Hays, Introduction to Digital Logic Design

C Programming

Course Code	BIT105	Year/Semester	I/I
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hr/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

This course combines the principles of computational thinking with hands-on C programming to introduce students to the world of computer science and programming. Students will learn howto approach and solve problems using computational thinking concepts and apply them to implement solutions. Through hands-on programming exercises and projects, students will gain a solid foundation in both computational concepts and C programming.

2. Course Objectives

The course aims to achieve the following objectives:

- Introduce principles of computational thinking and their relevance in problem-solving
- Familiarize students with the basic syntax, data types, and control structures
- Develop students' skills in algorithm design, implementation, and analysis
- Foster logical reasoning and systematic problem-solving approaches
- Provide practical experience in developing small-scale C programs.

3. Course Details

3.1 Lecture (48 hrs)

Unit 1: Introduction to Computational Thinking (2 hrs)

Overview of computational thinking and its significance in problem-solving. Problem- solving strategies and abstraction. Algorithm design and analysis.

Unit 2: Introduction to Programming and C Programming (4 hrs)

Types of programming and language translator, algorithm and flowchart. Overview of C and its historical significance, syntax and semantics. Installing and setting up the development environment (IDE or text editor).

Unit 3: Basic Syntax and Data Types (8 hrs)

Structure of a C program: header files, main function, and comments. Data types: int, float, char, and their modifiers. C tokens and character sets. Data types, constants and variables. Expression, statements and comments. Escape sequences and delimiters I/O functions. Formatted I/O operators: arithmetic operators, relational operators; logical operators; assignment operators, increment and decrement operators; ternary operator, bitwise operator; other operators (comma, sizeof). Expression evaluation. Operator precedence and associativity, type conversions, preprocessor directives: macros and their usage.

Unit 4: Control Statements (8 hrs)

Decision-making structures: if, if-else, nested if-else. Switch statement. Looping structures: while, for, do-while. Nesting control structures for complex conditions. Jump statements: break, continue, goto.

Unit 5: Arrays and Strings (6 hrs)

Declaring and initializing arrays. Accessing array elements and multi-dimensional arrays. Strings and string manipulation functions. Pointers and Array, array of pointers. Strings and string

manipulation functions (strlen, strcpy, strcat, strcmp).

Unit 6: Functions **(6 hrs)**

Creating and calling functions. Communication between functions, passing arguments, using library functions. Recursive functions.

Unit 7: Pointers **(6 hrs)**

Introduction to pointers and memory addresses. Pointer arithmetic and dynamic memory allocation (malloc, calloc, realloc, free). Passing arguments to functions: call by value and call by reference. Pointers and arrays.

Unit 8: Structures and Unions **(3 hrs)**

Defining and using structures. Nested structures and arrays of structures. Understanding unions and their uses.

Unit 9: File Handling **(5 hrs)**

File handling concepts: opening, reading, writing, and closing files. File opening modes. Sequential and random access file operations. Handling errors during file operations.

3.2 Practical/ Lab

- Demonstrate concept of syntax, data type, variable, constant and comment
- Demonstrate concept of I/O
- Demonstrate arithmetic operations
- Demonstrate concept of operators
- Demonstrate concept of type conversion
- Use preprocessor directives: include, macros
- Demonstrate concept of decision making structure: if, if-else, if-else ladder, nested if-else
- Demonstrate concept of switch statement
- Demonstrate concept of loop: for, while, do-while and their nested structure
- Demonstrate concept to jump statements: break, continue, goto
- Demonstrate concept of array: single and multi-dimension
- Demonstrate the concept of array of string
- Demonstrate the use of string manipulation functions
- Create and call the function
- Demonstrate concept of return type and non-return type function
- Demonstrate concept of passing argument by function: call by value and call by reference
- Use recursive function
- Create and use pointer
- Demonstrate pointer arithmetic
- Demonstrate dynamic memory allocation: malloc, calloc, realloc, free
- Use of pointer in array
- Define and user of structure
- Demonstrate the concept of nested structure
- Demonstrate the concept of array of structure
- Define and use of union
- Demonstrate file handling operations: opening, reading, writing, closing
- Demonstrate sequential file access operation
- Demonstrate random file access operation
- Handle error in file handling operation.

4. Evaluation Scheme

Evaluation of students' performance is divided into two main parts: (1) internal evaluation and (2) semester examination, with breakdown as follows.

Evaluation	Components	Weightage	Full	Pass

			Marks	Marks
Internal evaluation	Class attendance and performance	5	20	10
	Assignment	10		
	Seminar/project/presentation	5		
	Practical/lab examination	20	20	10
	Internal examination	20	20	10
	Total of internal evaluation		**60**	**30**
Semester examination	**Total of semester examination**		**40**	**20**

5. Books

- Y. P. Kanetkar, "Let Us C". BPB Publications.
- B. Gottfried, "Programming with C". McGraw Hill Education.
- B. Kernighan and D. Ritchie, "The C Programming Language". Pearson.

Year I Semester II

Mathematics II

Course Code	BIT151	Year/Semester	I/II
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hr/wk
		Practical	0
		Total	64 hrs

1. Course Description

This course is designed to provide students with a comprehensive understanding the concept and its uses to solving the problems of limits and continuity, differentiation and its applications, integration and its applications, ordinary differential equations, linear programming problem, application of matrix, computational method.

2. Course Objectives

The course aims to enable students to:

- Understand and formulate real world problems into mathematical statements.
- Develop solutions to mathematical problems at the level appropriate to the course.
- Describe or demonstrate mathematical solutions either numerically or graphically.

3. Course Details

3.1 Lecture **(48 hrs.)**

Unit 1: Limits and Continuity **(6hrs.)**

Limit, One-sided limits, Limits at infinity, Infinite limits and asymptotes, Continuity of a function.

Unit 2: Differentiation and its Applications **(9hrs.)**

Derivative, Geometrical interpretation, Derivative of a function, Rules of derivative, Implicit differentiation, Extreme values of function, Rolle's Theorem, Mean-Value Theorem, Geometrical meaning of Mean-Value Theorem, Monotonic functions and first derivative test.

Unit 3: Integration and its Applications **(9hrs.)**

Integration, Integration by parts, Trigonometric integrals, Trigonometric substitutions, Definite integral, Improper integrals, Area between curves, Length of plane curves, Area of surface of revolution.

Unit 4: Ordinary Differential Equations **(7hrs.)**

First order differential equations, Review of separable and exact differential equation, First-order linear and Bernoulli's differential equations and its solution.

Second order linear differential equations (Homogeneous) with constant and variable coefficient and their solution.

Unit 5: Linear programming problem **(5hrs.)**

Linear Programming Problem (LPP), Graphical Solution of LPP in Two Variables, Solution of LPP by Simplex Method (up to 3 variables).

Unit 6: Application of Matrix **(6hrs.)**

Block matrix, Partitioned Matrices, The Leontief Input Output Model, Applications to Computer Graphics (shear transformation, homogeneous coordinates in 2D & 3D, projection maps).

Unit 7: Computational Method **(6hrs.)**

System of Linear Equations, Solution of System of Linear Equations by Gauss Seidel Method and Matrix Inversion Method, Bisection method, Newton-Raphson Method for Solving Non-linear Equations.

3.2 Tutorials

The following tutorial activities shall be conducted to cover all the required contents of this course. This will enable the students to complete the related mathematical problems under the supervision of the subject teacher.

- Finding limit of a function
- Finding different asymptotes to a curve
- Examine the continuity of a function at a point and finding limiting value
- Find derivative of a function at a point, show that differentiability implies continuity but the converse may not be true, problems on derivative of implicit function
- Finding extreme value
- Problems in Mean value theorems: Rolle's theorem, Lagrange's mean value theorem and related problems
- Evaluate various indefinite integrals
- Finding value of Definite integrals, Improper Integrals
- Evaluation of area arc length and area of revolution
- Solution of separable, homogeneous and exact differential equation Linear, Bernoulli equation
- Solve second order homogeneous ODE with constant and variable coefficients, Euler-Cauchy equation
- Solving linear programming problem by graphical method
- Simplex method for standard problems
- Finding block matrix and partitioned matrix
- Develop and evaluate Leontief Input Output Model
- Problems based on matrix transformation - shear transformation, homogeneous coordinates in 2D & 3D, projection maps
- Solution of system of linear equations by Gauss Seidel method and Matrix Inversion method
- Solution of an equation by Bisection method and Newton-Raphson method.

4. Evaluation Scheme

Evaluation of student's performances is divided into two main parts: (1) Internal Evaluation and (2) Semester examination, with breakdowns as follows:

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal evaluation	Class attendance and performance	5	40	20
	Assignment	15		
	Seminar/project/presentation	20		
	Practical/lab examination	-	-	-
	Internal examination	20	20	10
	Total of internal evaluation		**60**	**30**
Semester examination	**Total of semester examination**		**40**	**20**

5. Books

- Thomas, G. & Finney, R. Calculus and Analytical Geometry. New Delhi: Narosa Publishing House.
- Kreyszig, E. Advanced Engineering Mathematics. New Delhi: John Wiley and Sons Inc.
- David C. Lay, Linear Algebra and Its Applications, 4th Edition, Addison Wesley, Pearson.

OOP in Java

Course Code	BIT152	Year/Semester	I/II
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hrs/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

This course is designed to provide students with a comprehensive understanding of object-oriented programming concepts using Java programming language. The course covers different aspects of programming, including fundamental concepts, different programming structures, classes and objects, inheritance, interface, package and modules, exception handling and some essential Java classes. Through a combination of lectures, practical exercises, and projects, students will gain the necessary skills to write computer programs using Java programming language.

2. Course Objectives

The course aims to achieve the following objectives:

- To introduce different concepts of object-oriented programming.
- To Provide concepts and skills of Java development environment and fundamental programming structures in Java.
- To familiarize students with concepts classes, objects, inheritance, and packages and modules and teach them how to use these concepts to write Java programs.
- To provide knowledge of exception handling and some essential Java classes.
- To teach students how to implement exception handling concepts and essential Java classes.

3. Course Details

3.1 Theory (48 hrs)

Unit 1: Introduction (6 Hrs.)

Java Introduction; Short History of Java; Java Buzzwords; Java in Web Development, Mobile Application Development, and Enterprise Application Development; Java Virtual Machine (JVM); Java Runtime Environment (JRE); Bytecode; Writing Simple Java Programs; Compiling and Running Java Programs using Command Line and IDE; Using Command Line Arguments; Using Scanner for Reading Input and System.out.print() for Writing Output

Unit 2: Fundamental Programming Structures (12 Hrs.)

Writing Comments; Primitive Data Types; Variables and Constants; Type Conversion and Casting; Operators (Arithmetic, Bitwise, Logical, Assignment, and Conditional); Precedence and Associativity of Operators; Control Statements (if, switch, for, while, do-while, for-each, nested statements, break, continue, and return); Working with Big Numbers; Arrays (One Dimensional and Multidimensional)

Unit 3: Classes and Objects (10 Hrs.)

Object Oriented Principles; Defining Classes; Adding Variables and Methods; Creating Objects and Accessing Class Members; Method Parameters and Return Types; Constructors; static Fields and Methods; Method Overloading; this Keyword; Access Control; Nested and Inner Classes; Recursive Methods; Garbage Collection

Unit 4: Inheritance and Interface (6 Hrs.)

Inheritance Basics; Defining Subclasses; Using super and final Keywords; Abstract Class; The Object Class; Dynamic Method Dispatch; Declaring, Extending, and Implementing Interfaces

Unit 5: Exception Handling **(5 Hrs.)**

Exception Handling Fundamentals; Exception Types; Uncaught Exceptions; Using try, catch, throw, throws, and finally; Java's Built-in Exceptions; Generics

Unit 6: Packages and Modules **(4 Hrs.)**

Built-in Packages; Creating and Using Packages; Modules

Unit 7: Essential Java Classes **(5 Hrs.)**

String; Primitive Type Wrappers; Math Class; File Input / Output

3.2 Java Programming Comprehensive Tutorials

- Setting Up Your Java Development Environment: A tutorial on installing and configuring the Java Development Kit (JDK) and Integrated Development Environment (IDE) for Java programming.
- Creating Your First Java Program: A step-by-step guide to writing, compiling, and running a simple Java program to understand the basic structure.
- Access Modifiers: Understanding access modifiers like public, private, protected, and default, and their implications on class members.
- Packages and Data Types: Exploring Java packages and different data types, with hands-on exercises to practice variable declarations and data manipulation.
- Working with Classes and Objects: Tutorials covering the creation of classes, instantiation of objects, and implementing constructors and methods.
- Object-Oriented Concepts: Exploring inheritance, polymorphism, encapsulation, and abstraction with practical examples.
- Advanced Inheritance and Interfaces: Tutorials on creating subclasses, implementing interfaces, and understanding multiple inheritance in Java.
- Conditional Statements: Tutorial sessions on using if-else statements, switchcase constructs, and logical operators for conditional programming.
- Exception Handling in Java: Comprehensive tutorials on try-catch blocks, checked vs. unchecked exceptions, and best practices for handling exceptions.
- Working with Files in Java: Step-by-step guidance on reading from and writing to files using Java I/O classes.

3.3 Laboratory Works

Write Java programs to implement all the concepts mentioned in each unit of the syllabus.

4. Evaluation Scheme

Evaluation of student's performances is divided into two main parts: (1) Internal Evaluation and (2) Semester examination, with breakdowns as follows:

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal evaluation	Class attendance and performance	5	20	10
	Assignment	10		
	Seminar/project/presentation	5		
	Practical/lab examination	20	20	10
	Internal examination	20	20	10
	Total of internal evaluation		**60**	**30**
Semester examination	**Total of semester examination**		**40**	**20**

5. Books

- Cay S. Horstmann (2019). Core Java Volume I – Fundamentals, Eleventh Edition, Pearson Education.
- Herbert Schildt (2022). Java: The Complete Reference, Twelfth Edition, McGraw Hill.
- Joshua Bloch (2018). Effective Java, Third Edition, Pearson Education.
- Marc Loy, Patrick Niemeyer, and Daniel Leuck (2020). Learning Java – An Introduction to Real World Programming with Java, O'Reilly, 2020.

Microprocessor & Computer Architecture

Course Code	BIT153	Year/Semester	I/II
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hr/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

This course is designed to provide students with a comprehensive understanding of microprocessor 8085 architecture, basic computer structure, central processing unit, microprogramed control, computer arithmetic, input/output, memory and pipelining mechanism.

2. Course Objectives

The course aims to achieve the following objectives:

- Introduce Microprocessor and Microprocessor based system.
- Explore features and functioning of intel 8085 microprocessor.
- Write an ALP using 8085 microprocessors.
- Familiarize basic computer.
- Identify the organization and design of basic computer.
- Explain microprogrammed control logic.
- Explore central processing unit.
- Familiarize with IO and memory organization.
- Identify computer arithmetic algorithms.
- Introduce pipelining mechanism.

3. Course Details

3.1 Theory **(48hrs.)**

Unit 1: Introduction **(4 hrs.)**

Microprocessor, Components of Microprocessor: ALU, Register Array, Control Unit, Microprocessor as a CPU, Microprocessor based system, Microcontroller, Microprocessor vs Microcontroller, Von-Neumann and Harvard Architecture

Unit 2: Microprocessor 8085 **(10 hrs.)**

Features of 8085 MPU, Operations: Microprocessor-initiated, Internal and Peripheral operations. Pinout and Signals of 8085, Bus Organization, Flags, Interrupts, Register Organization, Functional block diagram, Instruction Set, Opcode and Operands, Addressing modes, Assembly Language Programming with 8085. Instruction Cycle and Timing diagrams, Machine Cycle & T-State.

Unit 3: Basic Computer Organization and Design **(6 hrs.)**

Basic Computer, Instruction codes, Computer Registers, Computer Instructions, Instruction Cycle, Memory-Reference Instructions, Input-Output and Interrupt, Design of Basic Computer, Design of Accumulator Logic.

Unit 4: Microprogrammed Control **(6 hrs.)**

Control Memory, Address Sequencing, Microprogram, Microinstruction format, Symbolic and Binary microprogram, Design of Control Unit.

Unit 5: Central Processing Unit **(6 hrs.)**

Register Organization, Stack Organization, Instruction Format, Addressing modes, Data Transfer and Manipulation, Program Control, RISC and CISC architecture.

Unit 6: Computer Arithmetic **(6 hrs.)**

Fixed-point vs Floating-point arithmetic, Fixed-point: Addition, Subtraction, Multiplication and Division algorithms.

Unit 7: Input/output and Memory Organization **(6 hrs.)**

Input/output: Peripheral Devices, Input-Output Interfaces, Modes of data transfer, Direct Memory Access (DMA), DMA Controller, Pin Diagram & Internal configuration of 8237 processor.
Memory: Memory Hierarchy, Auxiliary memory, Associative memory, Cache memory, Virtual Memory.

Unit 8: Pipelining **(4 hrs.)**

Pipelining, Arithmetic pipelining, Instruction pipelining, Pipelining Hazards.

3.2 List of Tutorials

The various tutorial activities that outfits this course should cover all the content of this course to give students a space to engage more actively with the course content in the presence of instructor. Students should submit tutorials as assignments or class works to the instructor for evaluation. The following tutorial activities of 15 hrs should be conducted to cover all the content of this course:

A. Discussion based Tutorials

- Comparative analysis of Von-Neumann, Harvard and modified Harvard Architecture.
- Applications of microprocessor and microcontroller based system in real world.
- Hardware and programmable module of 8 bit microprocessor.
- Interpret the concept of memory segmentation and pipelining in 16 bit microprocessor.
- Review each entity memory hierarchy for modern processors.
- Interfacing of DMA, interrupt controller and UART with 8085 microprocessor.
- Comparative analysis of different aspects of computing system as defined in Flynn's Classification.

B. Assembly level programming tutorials

- Assembly level programming illustrating data transfer instructions, arithmetic instructions, logical instructions and branching instructions in 8085 microprocessor.
- Enlighten PUSH and POP instructions using appropriate assembly level programs.
- Assembly level programming for simple sequencing program, multiplication, division, table processing and sorting.
- Timing diagram for different types of instructions and RTL of each machine cycle involved during the execution of instructions.

C. Design Thinking tutorials

- Design a CPU for any given registers set, instruction set and state diagram. Show the RTL code for each execution cycle.
- Develop a control unit for any given state diagram.
- Design a micro-sequencer control unit for any given specifications following design procedure.

3.3 Laboratory Works

Laboratory works using 8085 Microprocessor Training Kit

- Illustrate Data Transfer, Arithmetic, Logical, Branching and Machine Control Instructions.
- Demonstrate instructions from each addressing mode.
- Perform 8 – bits addition, subtraction, multiplication and division.
- Perform 16 – bits addition and subtraction.
- Transfer block of data from one memory location to another.
- Arrange data in ascending/descending order.
- Find largest/smallest datum from data array.
- Identify even/odd, prime/composite number.
- Convert number from one base (radix) to another.

4. Evaluation Scheme

Evaluation of student's performances is divided into two main parts: (1) Internal Evaluation and (2) Semester examination, with breakdowns as follows:

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal evaluation	Class attendance and performance	5	20	10
	Assignment	10		
	Seminar/project/presentation	5		
	Practical/lab examination	20	20	10
	Internal examination	20	20	10
	Total of internal evaluation		**60**	**30**
Semester examination	**Total of semester examination**		**40**	**20**

5. Books

- Ramesh Gaonkar, Microprocessor Architecture, Programming and Applications with the 8085, Penram International Publishing India
- M. Morris Mano, Computer System Architecture, PHI.
- Ajit Pal, Microprocessor: Principles and Applications, TATA McGraw-Hill
- William Stalling, Computer Organization and Architecture, Pearson

UI/UX Design

Course Code	BIT154	Year/Semester	I/II
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hr/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

This course will teach you the fundamental process of experience design as well as how to effectively evaluate your work with the people you are designing for. You'll learn fundamental design research methods that will help you understand people, the sequences of their actions, and the context in which they work. You will learn practical techniques for making sense of what you see and transforming your observations into meaningful actionable insights and unique opportunity areas for design through the assignments.

2. Course Objectives

The course readings, activities, and assignments were designed to give students opportunities to meet the following objectives by the end of the semester:

- To create and document a portfolio-quality concept design using well-known experience design methods.
- To select appropriate methods and theories based on an understanding of the strengths and limitations of available approaches.
- To articulate your approach to design and user experience design specifically in a clear manner.
- To describe popular research and design processes used in UX as well as alternative approaches to UX design.
- To implement design prototypes in tools such as Photoshop, Illustrator, Figma, Adobe XD, HTML, CSS, Sketch, and Axure.
- To experiment with peer design critique.

3. Course Details

3.1 Theory (48 hrs)

Unit 1: Introduction to User Experience (6 hrs.)

User Interface Design, User Experience Design, UI vs UX, User-centric Design, Key framework of UX design, Universal and inclusive design, UX design tools

Unit 2: User Experience Design (12 hrs.)

Users, empathy map, users story, edge cases, users journey map, storyboarding, psychology principles that influence design, Ideation: exploring ideas, business needs and competitive analysis.

Unit 3: User Experience Research (12 hrs.)

UX research, power of UX research, bias in UX research, choosing the right research method, research plan, goals and questions, UX research process, ethics, privacy and data in real world, usability studies, its types and process.

Unit 4: Prototyping (12 hrs.)

Prototypes: wireframes and low-fidelity prototypes, introduction to Figma, digital wireframes, Lo-Fi prototype, usability testing and iteration on design.

Unit 5: Visual Design/ UI Design **(8 hrs)**

Introduction to Visual Design/ UI Design, Design Principles: Contrast, Visual hierarchy, Alignment, White space, Gestalt principle, Colour theory, Typography, layouts, Iconography and symbols, transition from Lo-Fi to Hi-Fi Design, Design System.

3.2 List of Tutorials

- Navigating the Figma User Interface Designing a Web Application
- Working with UI Elements and Content Making a Design Responsive
- Reusing Design Elements
- Building Case Studies
- Refining User Research and Creating Personas
- Designing Login Pages using Figma
- Developing Professional Website

3.3 Laboratory Works

Mini Research Project: Mini project for this course is an applied piece of user experience research. This may be:

- An investigation of user needs for a proposed software product, information system, web site, or
- An evaluation and assessment for a prototype or an existing system or service.

The more real it is, the better the project. The best projects are for an actual client. But you can also pick a site or product with which you have no direct connection.

Given the constraints of the semester, this is likely to be more of a pilot project than an actual, completed project for a client. It is most important that you get to practice a variety of methods and the entire process from planning to analysis and delivery of a product. It is unlikely that you will have time to go into enough depth with each method to have defendable results. This can be an individual or a group project; group projects are recommended.

4. Evaluation Scheme

Evaluation of student's performances is divided into two main parts: (1) Internal Evaluation and (2) Semester examination, with breakdowns as follows:

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal evaluation	Class attendance and performance	5	20	10
	Assignment	10		
	Seminar/project/presentation	5		
	Practical/lab examination	20	20	10
	Internal examination	20	20	10
	Total of internal evaluation		**60**	**30**
Semester examination	**Total of semester examination**		**40**	**20**

5. Books

- Elizabeth Goodman, Mike Kuniavsky, 2012. Observing the User Experience: A Practitioner's Guide to User Research, 2nd edition.
- Designing the User Interface, Ben Schneiderman, McGraw Hill Edition, 4th Edition.
- I. Scott MacKenzie. 2013. Human-Computer Interaction: An Empirical Research Perspective. Morgan Kaufmann
- Don Norman. 2013. The Design of Everyday Things: Revised and Expanded Edition. Basic Books.

- Tomer Sharon, 2016. Validating Product Ideas.

Operating System

Course Code	BIT155	Year/Semester	I/II
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hr/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

This course is designed to provide students with a comprehensive insight to the operating systems principle, design, implementation and applications.

2. Course Objectives

The course aims to achieve the following objectives:

- Introduce Operating System, its functions, types and structure.
- Identify process management mechanisms.
- Familiarize with memory management techniques.
- Introduce file and directory structure.
- Insight IO management theory and practices.
- Identify Security management activities.

3. Course Details

(48 hrs)

3.1 Theory

Unit 1: Introduction **(8 hrs.)**

Operating System Concepts, Functions of Operating System, Types of Operating System, System Call, Operating System Structure, Virtual Machine.

Unit 2: Process Management **(12 hrs.)**

Processes, Threads, Synchronization and Inter-Process Communication, Process Scheduling. deadlock characterization, handling deadlocks, deadlock prevention, avoidance, detection, recovery.

Unit 3: Memory Management **(10 hrs.)**

Memory Management: logical and physical addresses, swapping, contiguous allocation, paging, segmentations.
Virtual memory: demand paging, the performance of demand paging, page replacement algorithms, allocation of frames, thrashing.

Unit 4: File System **(6 hrs.)**

File operations, Access Methods, Directories and Levels, Directories Operations, file system mounting and sharing, protection, access control, File system layout, File system implementation.

Unit 5: Input/Output Management **(8 hrs.)**

I/O devices, Devices Controller, Memory Mapped I/O, Direct Memory Access (DMA), I/O software Principles: programmed I/O, Interrupt driven I/O, DMA based I/O, I/O Software Layers
Disk management: Disk structure, Disk scheduling algorithms.

Unit 6: Security **(4 hrs.)**

Fundamentals of Operating System Security, Controlling Access to Resources, Formal Models of Secure System, Attacks.

3.2 List of Tutorials

- Operating System Services (Class discussion)
- New Trends in Operating Systems (Oral Presentation).

- Evaluating the performance of different Process Scheduling Algorithm, Disk Scheduling Algorithm and Page Replacement Algorithm through hypothetical scenarios.
- WAP that creates multiple processes and manages their execution using different scheduling algorithms such as Round Robin, Priority Scheduling, and Shortest Job First.
- Implementing page replacement algorithms such as FIFO, LRU, and Clock, and evaluating their performance in a simulated virtual memory environment.
- WAP that simulates FCFS, SCAN and CSCAN disk scheduling algorithms.

3.3 Laboratory Works

- Simulate the process scheduling algorithms.
- Simulate the page replacement algorithms.
- Observe Disk Scheduling algorithms.
- Case Study: Unix, Linux and Android.

4. Evaluation Scheme

Evaluation of student's performances is divided into two main parts: (1) Internal Evaluation and (2) Semester examination, with breakdowns as follows:

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal evaluation	Class attendance and performance	5	20	10
	Assignment	10		
	Seminar/project/presentation	5		
	Practical/lab examination	20	20	10
	Internal examination	20	20	10
	Total of internal evaluation		**60**	**30**
Semester examination	**Total of semester examination**		**40**	**20**

5. Books

- Andrew S. Tanenbaum, Herbert Bos, Modern Operating Systems, Pearson.
- Silberschatz, Galvin and Gagne, Operating System Concepts, Addition Wesley.
- P. Pal Choudhury, Operating System Principle and Design, PHI
- D. M. Dhamdhere, System Programming and Operating System, Tata McGraw-Hill.

Statistics

Course Code	BIT201	Year/Semester	II/I
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hrs/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

This course introduces B. Tech students to the fundamental principles of statistics and its applications in engineering and scientific fields. Students will learn to analyze and interpret data, make informed decisions based on statistical evidence, and apply statistical techniques to solve real-world engineering problems. The course aims to develop students' understanding of statistical concepts and their ability to use statistical tools effectively.

2. Course Objectives

By the end of the course, students should be able to:

- Understand the basic concepts of statistics and its relevance in engineering and scientific research.
- Apply descriptive statistics to summarize and present data effectively.
- Comprehend probability theory and its role in modeling uncertain events.
- Analyze and interpret various types of probability distributions, including discrete and continuous distributions.
- Use sampling distributions to make inferences about population parameters.
- Conduct hypothesis tests to make decisions based on sample data.
- Apply regression analysis to model relationships between variables and make predictions.
- Design and analyze experiments to investigate cause-and-effect relationships.
- Apply statistical software to analyze data, perform calculations, and create visualizations.

3. Course Details

3.1 Theory **(48 hrs)**

Unit 1: Descriptive Statistics **(10 hrs)**

Introduction to Statistics- descriptive statistics and inferential statistics; types of Data-Qualitative vs. Quantitative data, levels of measurements- nominal level data, ordinal level data, interval level data, and ratio level data and variables.

Presentations of data: Frequency Distribution-Simple frequency distribution, relative frequency distribution, and percentage frequency distribution; Rule of constructing the frequency distribution, data array, stem and leaf display; Graphical presentations; bar char and pie chart, histogram, cumulative frequency distribution and Ogive. Descriptive measure of data: Quartiles, Deciles, and Percentiles; five number summary, box-and-Whisker plot, shape of the data-skewness and kurtosis; measure of central tendency-Mean, Median, and Mode; Measures of dispersion: Range, Average Deviation Measures, Standard Deviation, Variance, Relative Dispersion and Coefficient of Variation.

डीन

Students also practice and use the statistical software such as SPSS, STATA, R, or MS Excel etc. to calculate mean, standard deveins, skewness, kurtosis, and to visualize the data in graphical form.

Unit 2: Basic Probability **(5 hrs)**

Introduction to basic probability theory, Approaches to assigning probabilities- classical probability, empirical probability and subjective probability; Terminology in probability, principles of counting- multiplication, permutation and combination; contingency table; Probabilities Rules- Additional and multiplication; Probabilities under conditions of Statistical Independence, Probabilities under conditions of statistical dependence; Bayes' theorem.

Unit 3: Random Variables and mathematical expectation **(3 hrs)**

Random Variables- discrete random variable and continuous random variable, mathematical expectation of random variables, additional and multiplicative rule of mathematical expectations.

Unit 4: Probability Distributions **(9 hrs)**

Probability distribution function, joint probability function of two variables; Discrete probability distribution- Binomial Probability Distribution, Poisson Probability Distribution; Continuous probability distribution- Normal probability distributions, standard normal probability distribution function, normal distribution as an approximation of Binomial and Poisson probability distribution.
Students practice and use the statistical software such as SPSS, STATA, R, or MS Excel etc. to calculate the probabilities

Unit 5: Sampling Distributions and Estimation **(7 hrs)**

Introduction to Sampling, reasons for sampling, defining the populations, statistics vs. parameters, sampling methods- probability sampling and non-probability sampling methods; Sampling Distributions of the sample mean and proportions, concept of the Central Limit Theorem, weak law of large number,
Estimations: Introduction of basic terms of inferential statistics such as populations, sample, parameters, significance level, confidence level, critical value, degree of freedom; standard error; method of estimations- point estimations and interval estimations for mean and proportions, determining the sample size for the mean and proportion. Relations between sample size and margin of error.
Students practice and use the statistical software such as SPSS, STATA, R, or MS Excel etc. to calculate the confidence interval and sample size

Unit 6: Hypothesis Testing **(10 hrs)**

Introduction to hypothesis testing, Null and alternative hypotheses, Type I and Type II errors; t-tests for single mean and difference to two mean, z-tests for single mean and difference to two mean, paired t-test, Chi-Square test of goodness of fit, chi-square test for independence to two attributes. One way analysis of variance for difference of more than two means.
Students practice and use the statistical software such as SPSS, STATA, R, or MS Excel etc. to run hypothesis test the interpret the findings generated from the software to make informed decision.

Unit 7: Simple correlation and regression analysis **(4 hrs)**

Introduction, correlation analysis, regression equation using least squares method, measures of variation - computing the sum of squares; the coefficient of determination, standard error of the estimate; Inference about the population slope, correlation coefficients, and regression equation

डीन

using t-test and F-test; confidence interval for the slope, estimation of mean values; Limitations and errors.

Students also practice and use the statistical software such as SPSS, STATA, R, or MS Excel etc. to run correlation and regression analysis and interpret the findings generated from the software to make informed decision.

3.2 Tutorials

Needs tutorial class for practical session for the use of software to analyze the data in unit 1, 5, 6 and 7.

3.3 Laboratory Work

In the practical work, students should be able to use the statistical software such as MS Excel, SPSS, STATA or R to calculate and run the statistical tools such as mean, standard deviations, correlations, regression, hypotheses etc. of the dataset.

4. Evaluation Scheme

Evaluation of student's performances is divided into two main parts: (1) Internal Evaluation and (2) Semester examination, with breakdowns as follows:

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal Evaluation	Class Attendance and Performance	5	20	10
	Assignment	10		
	Seminar/Project/Presentation	5		
	Practical/Lab Examination	20	20	10
	Internal Examination	**20**	**20**	**10**
	Total Internal Marks		**60**	**30**
Semester End Examination			**40**	**20**

5. Books

- Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers and Keying Ye (2012). Probability and Statistics for Engineers & Scientist. 9th Ed. Printice Hall.
- Michael Baron (2013). Probability and Statistics for Computer Scientists. 2nd Ed., CRC Press, Taylor & Francis Group, A Chapman & Hall Book.
- Douglas C. Montgomery & George C. Runger (2003). Applied Statistics and Probability for Engineers. 3rd Ed. John Willey and Sons, Inc.
- Richard A. Johnson (2001). Probability and Statistics for Engineers. 6th Ed. Pearson Education, India
- Jay L. Devore (2012). Probability and Statistics for Engineering and the Sciences. 8th Ed. Cengage Learning, California Polytechnic State University, San Luis Obispo.

डीन

Database Management System

Course Code	BIT202	Year/Semester	II/I
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hrs/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

This course is designed to provide students with a comprehensive understanding of database management system concepts. The course covers different aspects of database management including, basics of database, data modelling using ER diagram, relational data model and relational algebra, structured query language, normalization, transaction processing, concurrency control and database recovery techniques. Through a combination of lectures, practical exercises, and projects, students will gain the necessary skills to design and develop databases.

2. Course Objectives

The course aims to achieve the following objectives:

- Introduce different concepts of database management systems.
- Provide concepts and skills of designing logical data models
- Familiarize with relational algebra
- Provide concepts of structured query language
- Present concepts of normalization
- Acquaint with database transactions
- Deliver concepts of concurrency control and database recovery techniques

3. Course Details

3.1 Theory **(48 Hrs)**

Unit 1: Foundations of Database **(6 Hrs.)**

Introduction; Characteristics of the Database; DBMS; Actors and Workers in DBMS; Advantages of DBMS; Data Models; Schema and Instance; Three Schema Architecture; Data Abstraction and Independence; Database Language, Database System Environment

Unit 2: Entity-Relationship Model **(8 Hrs.)**

Entity Relationship Model; Entity Types, Entity Sets, Attributes, and Keys; Relationship Types, Relationship Sets, Mapping and Structural Constraints; Weak Entity Types; ER Diagram Notations; Naming Conventions in ER Diagram; ER-to-Relational Mapping; Extended ER Model; Specialization and Generalization; Constraints on Specialization and Generalization

Unit 3: The Relational Data Model and Relational Algebra **(6 Hrs.)**

Relational Model; Domains, Attributes, Tuples and Relations; Relational Model Constraints; Relational Database Schemas; Relational Algebra: Unary Relational Operations: SELECT, PROJECT, RENAME; Relational Algebra Operations from Set Theory; Binary Relational Operations: JOIN and DIVISION; Aggregate Functions; Outer Joins

Unit 4: SQL **(9 Hrs.)**

Data Types; SQL Constraints; DDL Statements; DML Statements: Select, Aliasing, Substring Comparison, Arithmetic Operators, Order By, Null Values, Nested Query, Correlated Nested Query,

डीन

Exists, Set Comparison, Inner Join, Outer Join, Aggregate Functions, Group By Having; INSERT, DELETE, and UPDATE Statements; Views, Stored Procedure, Triggers

Unit 5: Normalization **(6 Hrs.)**

Design Guidelines for Relational Databases; Functional Dependencies; Normalization and its importance; First, Second and Third Normal Forms; Boyce-Codd Normal Form; Multivalued Dependency; Fourth Normal Form; Properties of Relational Decomposition

Unit 6: Concepts of Transaction Processing **(6 Hrs.)**

Introduction to Transaction Processing; Transaction and System Concepts; ACID Properties of Transactions; Characterizing Schedules Based on Recoverability; Characterizing Schedules Based on Serializability

Unit 7: Concurrency Control and Database Recovery **(7 Hrs)**

Concurrency Control and its purpose; Two-Phase Locking Technique for Concurrency Control; Concurrency Control Based on Timestamp Ordering; Concepts of Recovery; NO-UNDO/REDO Recovery Based on Deferred Update; Recovery Techniques Based on Immediate Update; Shadow Paging; The ARIES Algorithm; Database Backup and Recovery from Catastrophic Failures

3.2 List of Tutorials

Unit-specific exercises focused on problem-solving.

3.3 Laboratory Works

The practical work includes modelling ER and EER diagrams using appropriate CASE tools. The practical extends writing SQL query scripts to define and manipulate databases. Real world database projects should be designed and developed as a group project under the practical work.

4. Evaluation Scheme

Evaluation of student's performances is divided into two main parts: (1) Internal Evaluation and (2) Semester examination, with breakdowns as follows:

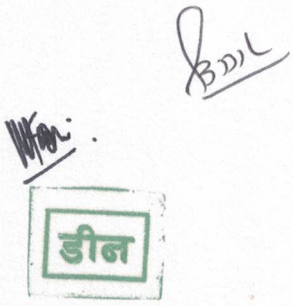
Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal Evaluation	Class Attendance and Performance	5	20	10
	Assignment	10		
	Seminar/Project/Presentation	5		
	Practical/Lab Examination	20	20	10
	Internal Examination	**20**	**20**	**10**
	Total Internal Marks		**60**	**30**
Semester-End Examination			**40**	**20**

5. Books:

- Fundamentals of Database Systems; Seventh Edition; Ramez Elmasri, Shamkant B. Navathe; Pearson
- Database System Concepts; Seventh Edition; Avi Silberschatz, Henry F Korth, S Sudarshan; McGraw-Hill
- An introduction to Database Systems; Eighth Edition; CJ Date, Pearson

डीन

- Database Design and Relational Theory: Normal Forms and All That Jazz; Second Edition; CJ Date; Apress
- Database Management Systems; Third Edition; Raghu Ramakrishnan, Johannes Gehrke; McGraw-Hill
- A First Course in Database Systems; Third Edition; Jaffrey D. Ullman, Jennifer Widom; Pearson
- Learning SQL: Generate, Manipulate, and Retrieve Data; Third Edition; Alan Beaulieu; O'Reilly Media



Data Communication and Networks

Course Code	BIT203	Year/Semester	II/I
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hrs/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

This course is designed to provide students with a comprehensive understanding of data communications and computer network focusing on the protocols, services and devices in each layer of TCP/IP reference model.

2. Course Objectives

The course aims to achieve the following objectives:

- Introduce data communication and computer networks.
- Identify the functions, devices, and protocols used in physical layer.
- Identify the services, devices, and protocols used in datalink layer.
- Identify and configure the services, devices, and protocols used in network layer.
- Explore the services, devices, and protocols used in transport layer.
- Find out the services, devices, protocols and networking tools used in applications layer.

3. Course Details

3.1 Theory **(48 hrs)**

Unit 1: Introduction **(6 hrs)**

Data Communications; Network criteria, Networks and Types; LAN Topologies, Networking devices, Network protocol & standards, OSI Reference model, TCP/IP Reference model, Addressing

Unit 2: Physical Layer **(12 hrs)**

Data and Signals, Periodic and Analog Signals, Digital Signals, Transmission Impairments, Data Rate Limits, Performance, Digital to Digital Conversion, Analog to Digital Conversion, Transmission Modes, Multiplexing, Transmission Media, Switching Techniques

Unit 3: Datalink Layer **(8 hrs)**

Functions, Error detection and Correction: Block Coding, Cyclic codes, Checksum; Data Link Control: Framing, Flow and Error Control, Protocols, Noiseless Channels, Noisy Channels; Multiple Access control: Random Access, Controlled Access and Channelization; Cellular telephony and Satellite network.

Unit 4: Network Layer **(8 hrs)**

Functions; IPv4 Address classes, header format, computing network and host IP addresses, subnetting; IPv6 header format, IPv4 vs IPv6, Routing: Static vs. Dynamic Routing; Routing Algorithms: Shortest-path, Flooding, Distance-vector, Link-state; Congestion control and prevention; Network layer protocols: IP, ICMP, IGMP, RIP, ARP, RARP, OSPF, IGRP, EIGRP, BGP.

डीन

पढ़ि

Unit 5: Transport Layer **(7 hrs)**

Functions; Reliable vs Unreliable Services, Connection-oriented vs connection-less services, Overview of TCP and UDP protocols and their applications, principle of reliable data transfer, congestion control.

Unit 6: Application Layer **(7 hrs)**

Functions; Application layer protocols: DNS, DHCP, SMTP, POP, IMAP, FTP, Telnet, HTTP, WWW, SNMP; Network Simulator and analyser tools: Packet Tracer, Wireshark.

3.2 Tutorial

Unit specific exercises focused on discussion, case studies and solutions of problem.

3.3 Laboratory Works:

- Prepare a specification document for networking devices: Router, Switch etc.
- Prepare Straight and cross-over cables.
- Implement LAN topology using packet tracer.
- Configure DNS and DHCP Server.
- Configure file server.
- Implement concept of subnetting using packet tracer.
- Prepare a VLAN.
- Implement static and dynamic routing using packet tracer.
- Analyse the data packets using Wireshark.
- Connect dissimilar networks.
- Configure active directory service in windows system.

4. Evaluation Scheme

Evaluation of students' performance is divided into two main parts: (1) internal evaluation and (2) semester examination, with breakdown as follows.

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal evaluation	Class attendance and performance	5	40	20
	Assignment	10		
	Seminar/project/presentation	05		
	Practical/lab examination	20	20	10
	Internal examination	20	20	10
	Total of internal evaluation		**60**	**30**
Semester examination	**Total of semester examination**		**40**	**20**

5. Books:

- Behrouz A. Forouzan; Data Communications and Networking, McGraw Hill.
- William Stallings, Data and Computer Communications, Prentice Hall of India, New Delhi.
- A.S. Tanenbaum, Computer Network, Pearson Education International.
- Kurose, Ross, Computer networking: A top-down approach, Pearson.

डीन

Web Technology-I

Course Code	BIT204	Year/Semester	II/I
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hrs/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

This course is designed to provide students with a comprehensive understanding of web development and the underlying technologies that power the World Wide Web. The course covers various aspects of web technology, including HTML, CSS, and JavaScript and website design. Through a combination of lectures, practical exercises, and projects, students will gain the necessary skills to create dynamic and interactive websites.

2. Course Objectives

The course aims to achieve the following objectives:

- Introduce students to the fundamental concepts and technologies that underpin the World Wide Web.
- Provide a solid foundation in HTML and CSS for web page design.
- Familiarize students with JavaScript and teach them how to use it to create dynamic and interactive elements on web pages.
- Teach students the process of design web applications using client side frameworks and libraries.
- Foster the ability to work collaboratively in a team environment, enabling students to effectively contribute to web development projects, communicate ideas, and coordinate tasks.
- Develop students' skills in identifying and resolving common issues and bugs in web applications through systematic troubleshooting and problem-solving techniques.
- Students must be able to register domain and host their project into suitable platform.

3. Course Details

3.1 Theory **(48 Hrs)**

Unit 1: Web Essentials **(4 hrs)**

The Internet, Basic Internet Protocols, the World Wide Web, HTTP (HTTP Request and Response Messages),FTP (Request, Response and Modes), Web Clients and Web Servers, Web Page and Web Site (Static and Dynamic),Web Application architecture(Tier architecture), Client Side and Server Side Scripting, Web 2.0, Web 3.0,Domain Registration, Web Hosting, Web application deployment, Using version control tools

Unit 2: HTML- Hypertext Markup Language **(12 hrs)**

HTML Introduction, HTML Documents, HTML Elements and Attributes, Text Formatting Elements, Layout elements, Image elements, Table elements, Form elements, HTML 5 and its features, HTML 5 Elements and Attributes, XHTML and its feature, Roles of html element for Search engine

डीन

optimization, Meta tags, Sitemap, HTML tags for forms element, HTML tags for lists, HTML audio and video tag, HTML Canvas, W3 standards for HTML , Semantic Markup with HTML5, Using Icon (Adding SVG Icon into web page, Resizing SVG icon,using font awesome for icon)

Unit 3: CSS- Cascading Style Sheet **(14 hrs)**

Introduction to CSS, CSS Syntax, Three Ways to apply CSS(Inline, Internal, External), CSS Selectors (Universal, ID, Class, Element, CSS combinatory, Pseudo-classes, and Pseudo-elements, CSS Attributes selectors), CSS box Model (Margin, Padding, Border), Using CSS unit(Introduction, Absolute unit, relative unit), CSS background properties, Using CSS Units(Absolute and Relative),Layout related properties (Float, Display: inline, block, inline-block, flex, grid),Position properties (static, relative, fixed, absolute, sticky, z-index), Embedding SVG(Embedding SVG (instead of linking) ,Styling SVG using CSS, Using current color for SVG icon),Defining and using CSS variable, Using inheritance for CSS properties, Designing html form element, Design html table, Responsive Design (Introduction, Media queries for responsive webpage, different responsive break point), CSS transitions (Transition property and transition duration, transition shorthand, adding easing with transition-timing function),CSS animation, Writing, Introduction to Frontend Framework Library (Bootstrap, Tailwind, Foundation)

Unit 4: Client Side Scripting **(18 hrs)**

JavaScript: Introduction, Basic Syntax, Variables(let, const and var) and Data Types, Statements, Operators, Literals, Control Statements, Functions, Objects, Arrays, Built-in Objects, Strings, Forms and Regular Expressions handling, Cookies, JavaScript Event Handling, JavaScript for Slider design, JavaScript for Tab Design, JavaScript for Animation, Introduction HTML DOM, JavaScript Browser BOM, XML introduction, AJAX, JSON, ES6 and its Features (call back ,arrow function, spread operator, map, set, promises, prototype based inheritance),Validating form using JavaScript, Making dynamic UI using DOM properties and methods, Introduction to Client Side Libraries and Frameworks.(Angular, React, Next.JS,Vue.js), Using jQuery for web application (jQuery Introduction, Install and Use jQuery Library, jQuery Syntax, jQuery Selector, jQuery Traversal, jQuery DOM manipulation, jQuery effects, jQuery Event, jQuery UI), JSON(Introduction, JSON Syntax, JSON Data Types, JSON and Security concerns)

3.2 List of Tutorial

Unit-specific exercises focused on problem-solving.

3.3 Laboratory Work

The practical work includes creating web pages and applications with using HTML, CSS, JavaScript and other technologies mentioned in the syllabus. Students also learn to use client libraries and frameworks for designing webpage and User interface.

- Learn to use version control tools
- Design website for different application domain like: (Company/Organization, Portfolio, E-commerce)
- Making responsive design using media query
- Making interactive webpage using JS library and framework
- Setting up application using Client side framework
- Use proper element for Search engine implementation

डीन

कठि

4. Evaluation Scheme

Evaluation of students' performance is divided into two main parts: (1) Internal evaluation and (2) Semester examination, with breakdown as follows.

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal Evaluation	Class Attendance and Performance	5	20	10
	Assignment	10		
	Seminar/Project/Presentation	5		
	Practical/Lab Examination	20	20	10
	Internal Examination	**20**	**20**	**10**
	Total Internal Marks		**60**	**30**
Semester-End Examination			**40**	**20**

5. Books:

- Learning PHP, MySQL & JavaScript: a step-by-step guide to creating dynamic websites, Nixon - O'Reilly – 2021
- Jackson, J. C. (2007). Web Technologies: A Computer Science Perspective.
- Responsive Web design with HTML5 and CSS3 learn responsive design using HTML5 and CSS3 to adapt websites to any browser or screen size, Frain - Packt - 2012
- Web application security: exploitation and countermeasures for modern web applications Hoffman - O'Reilly - 2020

डीन

Data Structure and Algorithm

Course Code	BIT205	Year/Semester	II/I
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hrs/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

The Data Structures and Algorithms course is designed to provide students with a comprehensive understanding of fundamental data structures and algorithms that are essential for efficient and effective problem-solving in computer science and programming. This course forms the backbone of computer programming and software development, as it equips students with essential problem-solving skills and strategies to tackle real-world computational challenges.

2. Course Objectives

- Understand the fundamental concepts of data structures and algorithms, their significance, and their role in computer programming.
- Develop a strong foundation in problem-solving techniques and algorithmic analysis to efficiently solve complex computational problems.
- The course will cover the concept of recursion and how it can be employed to design elegant and efficient algorithms. Students will learn how to convert iterative solutions into recursive ones and vice versa.
- Learn different data structures and their applications, such as arrays, linked lists, stacks, queues, trees, graphs, and hash tables.
- Explore fundamental algorithms for sorting, searching, and graph traversal, and understand their time and space complexities.
- Study advanced algorithmic paradigms, such as divide-and-conquer, greedy algorithms, dynamic programming, and backtracking.

3. Course Details

3.1 Theory **(48 hrs.)**

Unit 1: Introduction to Data Structures & Algorithms **(5 Hrs.)**

Overview of algorithms and their significance in computer science. Characteristics of good algorithms: correctness, efficiency, and simplicity. Algorithm Design Techniques: Greedy algorithm, Divide and conquer algorithm, Dynamic programming, Backtracking and randomized algorithm. Algorithm analysis: time complexity, space complexity, and Big Oh notation.

Introduction to data structures, Abstract data type (ADT) and their role in organizing and storing data.

Unit 2: Arrays, Pointers and Linked Lists **(5 Hrs.)**

Introduction to arrays and their properties. Multi-dimensional arrays. Pointers and Dynamic Memory Allocation: malloc, calloc, realloc, free. Singly linked lists: structure, operations, and applications. Doubly linked lists: structure, operations, and applications. Circular linked lists and their advantages.

Unit 3: Stack **(5 Hrs.)**

Basic Concept of Stack, Stack as an ADT, Stack Operations, Stack Applications, Stack implementation as Array and Linked List. Conversion from infix to postfix/prefix expression, Evaluation of postfix/ prefix expressions.

डीन

Unit 4: Queue **(5 Hrs.)**

Basic Concept of Queue, Queue as an ADT, Primitive Operations in Queue, Queue implementation as Array and Linked List. Linear Queue, Circular Queue, Priority Queue, Queue Applications. Double-ended queues (Deque) and their uses.

Unit 5: Recursion **(2 Hrs.)**

Principle of Recursion, Comparison between Recursion and Iteration. Example of recursive algorithms (factorial, Fibonacci Sequence, GCD, and Tower of Hanoi algorithms). Applications and Efficiency of Recursion.

Unit 6: Sorting **(8 Hrs.)**

Introduction; Internal and External Sorting; bubble sort, insertion sort, selection sort, merge sort, quick sort, shell sort, Heap Sort. Complexity of Sorting Algorithms.

Unit 7: Searching and Hashing **(5 Hrs.)**

Introduction to Searching, Search Algorithms: Sequential Search, Binary Search Efficiency of Search Algorithms.

Hashing: Hash Function and Hash Tables, Collision Resolution Techniques

Unit 8: Trees **(6 Hrs.)**

Concept and Definitions, Basic Operations in Binary Tree, Tree Height, Level and Depth
Binary Search Tree, Insertion, Deletion, Traversals, Search in BST, Balancing Tree: AVL tree, DSW Algorithm

Unit 9: Graphs **(7 Hrs.)**

Definition and Representation of Graphs, Graph Traversal, Minimum Spanning Trees: Kruskal and Prims Algorithm. Shortest Path Algorithms: Dijkstra's Algorithm

3.2 Tutorials

Unit-specific exercises focused on problem-solving.

3.3 Laboratory Works

The laboratory work consists of implementing the algorithms and data structures studied in the course. Since all the units consists of their specified practical except first unit, so students are expected to perform all the practical according to its unit lab work.

4. Evaluation Scheme

Evaluation of students' performance is divided into two main parts: (1) Internal evaluation and (2) Semester examination, with breakdown as follows.

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal Evaluation	Class Attendance and Performance	5	20	10
	Assignment	10		
	Seminar/Project/Presentation	5		
	Practical/Lab Examination	20	20	10
	Internal Examination	**20**	**20**	**10**
	Total Internal Marks		**60**	**30**
Semester-End Examination			**40**	**20**

डीन

5. Books:

- Y Langsam, MJ Augenstein and A.M , Tanenbaum Data Structures using C and C++, Prentice Hall India, Second Edition 2015
- Leen Ammeral, Programmes and Data Structures in C, Wiley Professional Computing
- G.W Rowe, Introduction to Data Structure and Algorithms with C and C++ , Prentice Hall India
- Srivastava, S.K. & Srivastava, D. (2011). Data structure Through C in Depth, (2nd Ed.), BPB Publication

Year II Semester IV

Research Methodology

Course Code	BIT251	Year/Semester	II/IV
Credit Weightage	2	Lecture	3 hrs/wk
		Tutorial	1 hr/wk
		Practical	2 hr/wk
		Total	96 hrs

1. Course Objective

This course aims to provide students with essential research skills, covering research design, data collection, and analysis. It integrates both qualitative and quantitative methods, emphasizing practical application through case studies and real-world examples. Students will learn to craft research proposals, analyze data, and present findings, preparing them for academic and professional research in civil engineering.

2. Course Details

1.1 Theory (32 hrs)

Unit 1: Basics of research (2 hrs)

Introduction to research: Definition and importance. Types of research: Qualitative, quantitative, mixed methods. Research process: Steps in conducting research. Ethical considerations in research: Informed consent and avoiding plagiarism. Intellectual property rights: Patents, copyrights, trademarks, and protecting research work.

Unit 2: Research problem and literature review (5 hrs)

Identifying research problems: Characteristics of a good research problem, formulating research questions. Literature review: Purpose and importance, conducting a literature search, writing a literature review.

Unit 3: Research design and method (9 hrs)

Research design basics: Types of research designs (experimental, descriptive, correlational, exploratory), choosing an appropriate research design. Case study and survey research: Designing case studies, designing surveys and questionnaires. Qualitative methods: Interviews and focus groups, observation techniques. Qualitative data analysis: Coding and thematic analysis, software tools for qualitative analysis. Quantitative methods: Surveys and questionnaires, experimental data collection. Quantitative data analysis: Descriptive and inferential statistics, using software tools (e.g., SPSS, Excel). Mixed methods research: Introduction to mixed methods, integration of qualitative and quantitative data, designing mixed methods research, data collection and analysis in mixed methods (strategies for data integration).

Unit 4: Advanced topics in research methodology (6 hrs)

Introduction to meta-analysis: Purpose and importance, conducting a meta-analysis. Systematic reviews: Methodology and best practices, examples and applications. Advanced sampling methods: Probability and non-probability sampling, sample size determination. Advanced sampling designs: Stratified, cluster, and systematic sampling. Introduction to big data: Characteristics and challenges. Data mining techniques: Association rules, classification, and clustering.

Unit 5: Research proposal and report writing **(6 hrs)**

Components of a research proposal: Title, abstract, and introduction; literature review and methodology. Writing a research proposal: Tips for effective proposal writing, examples and templates. Structure of a research report: Introduction, methodology, results, discussion, conclusions, and recommendations. Writing and presenting research findings: Effective writing strategies, preparing and delivering presentations.

Unit 6: Writing and publishing research papers **(4 hrs)**

Structure of a research paper: Abstract, introduction, methodology, results, discussion, and conclusion. Writing tips and strategies: Clarity, coherence, conciseness, and common pitfalls and how to avoid them. Understanding the peer review process: Types of peer review, responding to reviewers' comments. Selecting a journal for publication: Factors to consider, submission guidelines and process.

2.2 Tutorials

Unit-specific exercises focused on case studies and applications in Information Technology, leading to a research paper draft.

2.3. Practical

- Qualitative and Quantitative data analysis using appropriate tools.
- Write a publishable research paper

2. Evaluation Scheme

Evaluation of students' performance is divided into two main parts: (1) internal evaluation and (2) semester examination, with breakdown as follows.

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal evaluation	Class attendance and performance	5	40	20
	Assignment	15		
	Seminar/project/presentation	20		
	Practical/lab examination	-	-	-
	Internal examination	20	20	10
	Total of internal evaluation		**60**	**30**
Semester examination	**Total of semester examination**		**40**	**20**

3. Books

- C.R. Kothari. "Research Methodology - Methods and Techniques." New Age International.
- D. V. Thiel, "Research Methods for Engineers." Cambridge University Press.
- J. W. Creswell and J. D. Creswell, "Research Design: Qualitative, Quantitative, and Mixed Methods Approaches." SAGE Publications.
- R. Kumar, "Research Methodology: A Step-by-Step Guide for Beginners." SAGE Publications.
- U. Flick, "Introducing Research Methodology: Thinking Your Way Through Your Research Project." SAGE Publications.
- G. R. Marczyk, D. DeMatteo, and D. Festinger, "Essentials of Research Design and Methodology." Wiley.

Python Programming

Course Code	BIT252	Year/Semester	II/IV
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hrs/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

This Programming with Python course is designed to provide students with a comprehensive understanding of Python programming and web development using the framework. The course starts with an introduction to Python, covering essential topics such as variables, data types, control flow, and functions. Students will learn how to handle files and gain a solid understanding of object-oriented programming (OOP) in Python.The course then delves into a popular web development framework in Python. Students will learn how to set up a python project, create models, perform database operations, and handle user authentication and authorization. Additionally, students will explore templating engine, forms, and the admin interface. Optional topics include Rest Framework for building RESTful APIs and deploying the application on web servers like Heroku or AWS.Throughout the course, students will work on a mini-project using Framework, applying the concepts they have learned to develop a small-scale web application with dynamic content rendering, user authentication, and database interactions.

2. Course Objectives

The primary objective of this course is to equip students with the knowledge and skills to become proficient Python programmers and web developers using the framework. By the end of the course, students should be able to:

- Understand the fundamentals of Python programming, including variables, data types, control flow, and functions.
- Grasp the concepts of object-oriented programming in Python and apply them to real-world scenarios.
- Develop web applications including setting up a project, creating models, and handling user authentication and authorization.
- Utilize Framework templating engine and forms to render dynamic content and handle user input.

3. Course Details

(48 hrs.)

3.1 Theory

(10hrs)

Unit 1: Python Basics:

Introduction to Python: Overview of Python and its applications, Installing Python and setting up the environment Running, Python scripts and interactive mode. Variables and Data Types: Numeric data types (int, float, complex), String data type and manipulation, tuples, and dictionaries, Boolean values and operators. Control Flow: If-else statements Loops (for and while), Break and continue statements, Exception handling with try-except blocks. Functions: Defining functions and function arguments, Return values and function scopes, Lambda functions and higher-order functions, Recursion. File Handling: Reading and writing text files, Working with CSV and JSON files

Unit 2: Python OOP: **(6 hrs)**

Introduction to Object-Oriented Programming (OOP): Understanding classes and objects, Encapsulation, inheritance, and polymorphism, Constructors and destructors. Working with Classes and Objects: Defining classes and creating objects Instance and class variables, Methods and properties Inheritance and Composition Extending classes through inheritance
Composition and using multiple classes together. Advanced OOP Concepts: Method overloading and overriding, Abstract classes and interfaces, Class decorators

Unit 3: Python Scripting: **(2 hrs)**

Command-Line Arguments, Parsing command-line arguments, Arg parse module for more complex argument parsing. Working with Modules: Creating and importing modules Packages and the module search path. Regular Expressions: Pattern matching using regular expressions, Replacing and searching for patterns

Unit 4: Python Database and Database Connectivity **(4 hrs)**

Introduction to Databases: Understanding relational databases and SQL Overview of popular database systems. SQL Basics: CRUD operations (Create, Read, Update, Delete) Joins, subqueries, and aggregate functions Database Connectivity in Python: Connecting to databases from Python, ExecutingSQL queries with Python

Unit 5: Introduction to Web Development **(10 hrs)**

Understanding web development concepts, Introduction to Framework and its features, setting up any Framework project and running the development server. Framework Models: Defining models and database tables, Querying data with Framework ORM (Object-Relational Mapping). Views and Templates: Creating views and handling URL patterns Using any Framework templating engine Rendering dynamic data in templates Forms and User Input: Handling forms and user input in Framework Form validation and processing Authentication and Authorization: Managing user accounts and authentication Controlling access to views and resources Static Files and Media Handling: Managing static files (CSS, JavaScript, etc.). Uploading and serving media files (images, videos, etc.). Admin Interface: Using any Framework built-in admin interface Customizing the admin interface for models. Rest Framework (optional): Introduction to RESTful APIs, Creating APIs with Rest Framework

3.4 Tutorials

Unit-specific exercises focused on problem-solving.

3.5 Laboratory Works:
Project: Personal Blogging Platform

Description: In this project, you will build a personal blogging platform using Framework. The platform will allow users to register, log in, create blog posts, and view posts from other users. Users can like and comment on blog posts, and admins can manage the posts and user accounts through the admin interface.

4. Evaluation Scheme

Evaluation of students' performance is divided into two main parts: (1) Internal evaluation and (2) Semester examination, with breakdown as follows.

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal Evaluation	Class Attendance and Performance	5	20	10
	Assignment	10		
	Seminar/Project/Presentation	5		
	Practical/Lab Examination	20	20	10
	Internal Examination	**20**	**20**	**10**
	Total Internal Marks		**60**	**30**
Semester-End Examination			**40**	**20**

5. Books:

- "Python Crash Course" by Eric Matthes (2nd Edition): ISBN-13: 978-1593279288
- "Django for Beginners" by William S. Vincent (3rd Edition): ISBN-13: 978-1099701185
- "Python Cookbook" by David Beazley and Brian K. Jones
- "Django 3 By Example" by Antonio Mele

Management Information System

Course Code	BIT253	Year/Semester	II/IV
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hrs/wk
		Practical	2 hrs/wk
		Total	96 rs

1. Course Description

This course is designed to provide students with a comprehensive understanding of management information system. The course covers different aspects of management information system including information system in the global business, collaboration and communication tools, different types of information systems, ethical and social issues, concept of business intelligence, enterprise applications, and decision making.

2. Course Objectives

The course aims to achieve the following objectives:

- To introduce different concepts of management information system.
- To familiarize students with the concepts of information systems used in different level of management in organizations.
- To provide knowledge on impacts of information systems in organizations.
- To provide knowledge of different communication and collaboration tools using to communicate and collaborate in organizations.
- To introduce concepts and impacts of strategic information systems used in organizations.
- To provide knowledge of ethical and social issues related to the use of information systems in business and organizations.
- To teach importance of business intelligence.
- To familiarize different enterprise applications used to support different business activities.
- To provide knowledge of decision-making and decision-making process.

3. Course Details

3.1 Theory (48 hrs.)

Unit 1: Information Systems in Global Business Today **(4 Hrs.)**

Management Information System and its Importance; Globalization Challenges and Opportunities; Digital Firm; Strategic Business Objectives of Information System; Information System and its Functions; Business Perspective on Information Systems; Complementary Assets; Academic Disciplines; Sociotechnical Systems

Unit 2: Global Business and Collaboration **(6 Hrs.)**

Business Process; Relationship of Business Process to Information System; Systems for Different Management Groups; Systems for Linking the Enterprise; E-business, E-commerce, and E-government; Collaboration, Social Business, and Benefits; Building Collaborative Culture and Business Processes; Tools and Technologies for Collaboration and Social Business; Information Systems Function in a Business – Information Systems Department, Organizing Information Systems Function

Unit 3: Information Systems, Organization, and Strategy **(8 Hrs.)**

Features of Organization; Impact of Information System on Organizations; Internet and Organizations; Implications for Design and Understanding of Information Systems; Porter's Competitive Forces Model, Dealing with Competitive Forces using Information Systems, Internet's Impact on Competitive Advantage and Business Value Chain Model; Challenges Posed by Strategic Information System

Unit 4: Ethical and Social Issues in Information Systems **(8 Hrs.)**

Ethical, Social, and Political Issues raised by Information Systems; Key Technology Trends; Responsibility, Accountability, and Liability; Ethical Analysis; Candidate Ethical Principles; Professional Codes of Conduct; Real World Ethical Dilemmas; Information Right; Property Right; Accountability; Liability; System Quality; Quality of Life; Health Risks

Unit 5: Foundations of Business Intelligence **(8 Hrs.)**

File Organization; Problems with Traditional File Environment; DBMS and its Capabilities; Designing Databases; Nonrelational Databases, Cloud Databases, and Blockchain; Tools and Technologies to improve Business Performance and Decision Making; Big Data Challenges; Business Intelligence Infrastructure; Analytical Tools; Assuring Data Quality

Unit 6: Key System Applications for Digital Age **(8 Hrs.)**

Enterprise Systems, Enterprise Software and Business Value of Enterprise Systems; Supply Chain Management System and Software, Business Value of Supply Chain Management System; Customer Relationship Management System and Software, Business Value of CRM; Challenges of Enterprise Applications; Role of Knowledge Management System in Business; Artificial Intelligence and Machine Learning; Uses of AI in Business; Knowledge Work Systems

Unit 7: Enhancing Decision Making **(6 Hrs.)**

Types of Decisions, Decision Making Process, Business Value of Improved Decision Making; Information System, Management Activities, and Management Decision Making; Business Intelligence and Business Analytics Support for Decision Making; Decision Support for Operational, Middle and Senior Management with Business Intelligence

3.2 Tutorials

Unit-specific exercises focused on problem-solving.

3.3 Laboratory Works:

The laboratory work includes developing decision-making applications using spreadsheet and suitable programming language, learning to used different communication and collaboration tools, and different tools and technologies for business intelligence and big data.

4. Evaluation Scheme

Evaluation of students' performance is divided into two main parts: (1) Internal evaluation and (2) Semester examination, with breakdown as follows.

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal Evaluation	Class Attendance and Performance	5	20	10
	Assignment	10		
	Seminar/Project/Presentation	5		
	Practical/Lab Examination	20	20	10
	Internal Examination	**20**	**20**	**10**
	Total Internal Marks		**60**	**30**
Semester-End Examination			**40**	**20**

5. Books:

- Kenneth C. Laudon and Jane P. Laudon (2022). Management Information Systems: Managing the Digital Firm, Pearson, 17th Edition.
- Paul Beynon-Davies (2019). Business Information Systems, 3rd Edition, Red Globe Press.
- Eric Frick (2017). Information Technology Essentials: Introduction to Information Systems Volume 1.
- James O'Brien and George Marakas (2010). Management Information Systems, 10th Edition, McGraw Hill.

Web Technology II

Course Code	BIT254	Year/Semester	II/IV
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hrs/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

This course is designed to provide students with a comprehensive understanding of the underlying technologies that power the Web Applications. The course covers various aspects of web development, including advanced server-side scripting technologies, web servers, databases, and web application development, API design and creation and implementation of MVC Framework. Through a combination of lectures, practical exercises, and projects, students will gain the necessary skills to create real time dynamic and interactive web application along with CMS.

2. Course Objectives

- Understand the fundamental concepts of data structures and algorithms, their significance, and their role in computer programming.
- Develop a strong foundation in problem-solving techniques and algorithmic analysis to efficiently solve complex computational problems.
- The course will cover the concept of recursion and how it can be employed to design elegant and efficient algorithms. Students will learn how to convert iterative solutions into recursive ones and vice versa.
- Learn different data structures and their applications, such as arrays, linked lists, stacks, queues, trees, graphs, and hash tables.
- Explore fundamental algorithms for sorting, searching, and graph traversal, and understand their time and space complexities.
- Study advanced algorithmic paradigms, such as divide-and-conquer, greedy algorithms, dynamic programming, and backtracking.

3. Course Details

3.1 Theory (48 hrs.)

Unit 1: Server Side Scripting: PHP (12 hrs)

Introduction, Syntax, Variables, Data Types, Operators, Control Statements, Functions, Arrays, Strings, Super global, Forms, File Upload, Form validations, Regular Expression for Server side validation, File Handling, Cookies, Session, Error and Exception, Database Handling(Create database, table, add new record, retrieve records, update existing record and delete record), Configuring and Sending Email using localhost and live server, File handling and it's importance for web application log, Introduction to Server Side Frameworks.

Unit 2: Web application security (8 hrs)

Credential stuffing, SQL Injection, Session Hijacking, Cross Site Scripting (XSS), Cross-site request forgery (CSRF), DDoS Attack, Using Client Side and Server Side technique to handle application attack, Using PDO for secure database transaction, implement .htaccess for URL rewrite for security

Unit 3: Implement OOP using PHP **(8 hrs)**

Introduction, Objects, Declaring a class, The new keyword and constructor, Destructor, Access method and properties using $this variable, public ,private, protected properties and methods, Static properties and method ,Class constant, Inheritance & code re-usability, Polymorphous, Parent:: & self:: keyword, Instance of operator, Abstract method and class, Interface ,Traits, Magic Methods, Auto loading class, Namespaces ,Class / Object Functions, Exception Handling ,Structuring Project with OOPs

Unit 4: Designing Web Services and API **(8 hrs)**

Data and Web 2.0, Web Services Architecture, API types and protocols: SOAP, RPC Style SOAP, Document style SOAP, UDDI, WSDL, REST services, Data formats: XML, JSON, Design Rest API: Evolution of REST/JSON API, and Introduction to RESTful API, REST API Architecture, REST API Resources, REST API CRUD Implementation, API Error Handling, REST API verbs and methods, REST API HTTP Status Codes, REST API Change Management and Versioning, REST API Partial Responses, Authenticating API Resource: JSON Web Tokens, API Keys, Prepare API document using Swagger, Testing API using Postman, REST API Real World Examples, API implementation for Major Social Media and Tools (Google, PayPal, Facebook)

Unit 5: Implement Popular Web Framework **(12 hrs)**

MVC Framework - Introduction of web framework, Installation, Directory Structure, URL Routing with different HTTP methods, Implement migration for database, Create Controller, Create View, Passing Data in View, Accessing Variable, Implement Control Statement, Include a File from view, using template engine for dynamic control flow, Form Handling, Connecting Database, Creating and implementation of model, jQuery and AJAX implementation, Error Handling(Manage HTTP error using custom error page), Validation Preserving the Data, File Uploading, Session, Crud Operations (Create, Read, Update, Delete), Implementation of Design pattern (Singleton, Repository ,Factory)

3.2 Tutorials

Unit-specific exercises focused on problem-solving.

3.3 Laboratory Works:

The practical work includes creating web pages and applications with using Object oriented programming using PHP, JavaScript Technology, Designing Rest API, and implement popular web framework mentioned in the syllabus.

- Create dynamic web application with backend CMS
- Create web application to implement Object Oriented PHP structure
- Create and implement REST API for data sharing
- Implement web framework for web application development
- Design and develop compete web-based application like, E-commerce website, Portfolio, different organization website using above technologies.

4. Evaluation Scheme

Evaluation of students' performance is divided into two main parts: (1) Internal evaluation and (2) Semester examination, with breakdown as follows.

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal Evaluation	Class Attendance and Performance	5	20	10
	Assignment	10		
	Seminar/Project/Presentation	5		
	Practical/Lab Examination	20	20	10
	Internal Examination	**20**	**20**	**10**
	Total Internal Marks		**60**	**30**
Semester-End Examination			**40**	**20**

5. Books:

- Learning PHP, MySQL & JavaScript: a step-by-step guide to creating dynamic websites, Nixon - O'Reilly – 2021
- Jackson, J. C. (2007). Web Technologies: A Computer Science Perspective.
- PHP and MySQL Web development, Welling & Thomson - Sams - 2002
- Benharosh, J. (2016). *The essentials of Object Oriented PHP Learn, practice, and apply*.
- Vora, P. (2009). *Web application design patterns*. Morgan Kaufmann Publishers/Elsevier.
- PITT, C. (2021). *Pro php 8 MVC: Model View Controller Architecture -driven application development*. APRESS.
- STAUFFER, M. (n.d.). *Laravel: Up & running* (3rd ed.). O'REILLY MEDIA.

System Analysis and Design

Course Code	BIT255	Year/Semester	II/IV
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hrs/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

This course is designed to provide students with a comprehensive understanding of System and System Development Ecosystem. The course covers different approaches of system analysis and design and various success and challenging real cases of System Development

2. Course Objectives

The course aims to achieve the following objectives:

- Introduce students to the fundamental concepts of System and System Design
- Equip the students with different quality issues of system.
- Familiarize students' system development based upon the valid customer's requirements.

3. Course Details

3.1 Theory

(48 hrs)

Unit 1: Introduction to System Development **(10 hrs)**

Introduction of System Analysis and Design, Understanding the system, System Diagram, Development Process, Management Process, Supporting Processes, System Structure, ,Traditional waterfall SDLC , Other Approaches: Prototyping , Spiral, RAD, Introduction to Agile Development, CASE tools.

Unit 2: System Concept Formation **(6 hrs)**

Introduction to System Concept, Identifying and Selecting the System Development Projects, Corporate and Information Systems Planning, The Problem Identification, justifying a Solution, Generating broad alternative solution, Evaluating the proposal, Feasibility and its types, Selecting an alternative, Preparing statement of a User Requirement

Unit 3: Requirement Analysis **(6 hrs)**

Introduction to Requirement Analysis, Functional and non-functional requirement, User requirements, System requirements, Interface requirements, Information Collection, Questionnaires, Electronic Data Collection, Information by Observation, Using Ethnography, From observation to design, Information by Prototyping, Interface Prototyping. Prototyping Processes, Interviewing, Software requirement document .

Unit 4: Data Flow Diagram (DFD) **(6 hrs)**

Introduction to Data Flow Diagram, DFD symbols, Describing systems by DFD, Logical and Physical DFDS, Convention for good DFDS, DFDS Levelling.

Unit 5: Process Descriptions **(6 hrs)**

Introduction to Process Description, Process descriptions method, Structured English, Decision tables, Extended Decision tables, establishing logical correctness of Decision tables, Using Karnaugh maps to detect logical errors in decision tables, Eliminating redundant specifications.

Unit 6 : Implementation and Maintenance **(8 Hrs.)**

System Implementation: Introduction, System Implementation, Software Application Testing, testing types Installation and its types Documenting the System, Training and Supporting Users, Organizational Issues in Systems Implementation; Maintaining Information Systems: Introduction, Maintaining Information Systems, Conducting Systems Maintenance

Unit 7: Object Modelling **(6 hrs)**

Introduction to Object Modelling, Object environment, Object structures, Modelling behaviour in object modelling, Unified Modelling Language (UML), Class and object diagram, Use case diagram, State diagram, Sequence diagram.

a. Tutorial

Unit-specific exercises focused on problem-solving.

b. Laboratory Work

In the practical session, students will learn to use project management, CASE, and modelling tools. They also prepare a project report that includes at least analysis, design, and implementation phases of system analysis and design.

4. Evaluation Scheme

Evaluation of student's performances is divided into two main parts: (1) Internal Evaluation and (2) Semester examination, with breakdowns as follows:

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal evaluation	Class attendance and performance	5	40	20
	Assignment	10		
	Seminar/project/presentation	05		
	Practical/lab examination	20	20	10
	Internal examination	20	20	10
	Total of internal evaluation		**60**	**30**
Semester examination	**Total of semester examination**		**40**	**20**

5. Books :

- Igor Hawryszkiewycz (IH), "System Analysis and Design"
- Jeffrey A. Hoffer, Joey F. George, Joseph S.Valacich (JH) "Modern System Analysis and Design"
- Ian Somerville (IS) "Software Engineering"

Project I

Course Code	BIT290	Year/Semester	II/IV
Credit Weightage	2	Class Load	4 hrs/wk
		Practical	4 hrs/wk
		Total	64 hrs

1. Course Description

This is a fully practical course that expects students to implement concepts learned during the first two years of their study. Students are encouraged to go beyond the syllabus and develop realistic, technically sophisticated projects.

2. Course Objectives

The objective of this project work is to make the student able to:

- Lead a software project development
- Work in a team
- Use CASE tools
- Write programs and improve programming skills
- Write test cases for software testing and improve QA skills
- Enhance problem-solving, report writing, and presentation skills

3. Nature of Project

- It is an academic project focused on development of a desktop based, web based, or mobile based application using any programming language and technology of their expertise and comfort
- Students can develop the application containing CRUD operations or any other algorithms, if applicable. (Examples: Information Systems, E-Commerce Portals, Game Applications etc.)
- Student should use appropriate CASE tools during SDLC
- While implementing the project, students should be encouraged to write their own modules rather than relying on APIs or Plugins (except in some unavoidable circumstances)
- Emphasis on problem-solving, system design, and implementation
- Project group size: Maximum 3 students in a group

4. Focus of the Study

Each student in a group should have equal participation in every phase of the project. The students should focus on the following different software development phases during the development of their project work;

4.1. Problem Identification

4.2. System Analysis

- Feasibility Study
- System Requirement Specification (SRS)

4.3. System Design

- Architecture Design
- Interface Design
- Database/Procedure/Algorithm Design

4.4. Implementation and Testing

5. Phases of the Project Work

5.1. Proposal Submission and defense
- Timeline: 3rd - 4th week of the 4th Semester

5.2. Mid-Term Progress Report Evaluation and Defense
- Timeline: 11th –12th week of the 4th Semester

5.3. Final Submission and Defense
- Final report submission: At least 10 days before the final defense date
- Defense includes viva, presentation, project demonstration, and report evaluation

6. Supervision

- Supervised by a regular faculty member.
- One supervisor may guide up to four groups.

7. Evaluation Criteria

Total Marks: 100

Evaluation Phase	Evaluators	Marks
Proposal Defense	HoD/Coordinator (4), Supervisor (12), Internal Examiner (4)	20
Mid-Term	HoD/Coordinator (8), Supervisor (24), Internal Examiner (8)	40
Final Defense	HoD/Coordinator (5), Supervisor (15), Internal Examiner (5), External Examiner (15)	40

8. Evaluation Focus on

- Presentation Skills
- Technical Work Quality
- Documentation Quality
- Viva Performance
- Project Demo
- Individual Contributions

9. Roles & Responsibilities

Role	Responsibilities
HoD/Coordinator	Oversee project schedule, arrange defenses, evaluate at all stages
Supervisor	Guide, monitor, and evaluate the project work throughout
Internal Examiner	Evaluate at proposal, midterm, and final defense
External Examiner	Evaluate final presentation, viva, and demo of project
Student	Active in project development, documentation, and defense. Keep supervision logs

10. Report Contents

10.1. Prescribed Content Flow for the Project Proposal

1. Introduction
2. Problem Statement
3. Objectives
4. Methodology
    a. Requirement Identification
        i. Study of Existing System
        ii. Requirement Collection
    b. Feasibility Study
        i. Technical
        ii. Operational
        iii. Economic
    c. High Level System Design
        i. System Flow Chart
        ii. Methodology of Proposed System
        iii. Working Mechanism of Proposed System
5. Project Timeline (Gantt Chart)
6. Expected Outcome
7. References (IEEE Format)

10.2. Prescribed Content Flow for the Project Report

1. Cover & Title Page
2. Certificate Page
    a. Supervisor Recommendation
    b. Internal and External Examiners' Approval Letter
3. Abstract
4. Acknowledgement
5. Table of Contents
6. List of Abbreviations, List of Figures, List of Tables
7. Main Report (Chapters 1–5)
8. Appendices (Screenshots, Source Code, Supervisors Log Sheets)
9. References (IEEE Format)
10. Bibliography (if any)

10.3. Prescribed Chapters in Main Report

Chapter 1: Introduction

1.1. Introduction
1.2. Problem Statement
1.3. Objectives
1.4. Scope and Limitation
1.5. Report Organization

Chapter 2: Background Study and Literature Review

2.1. Background Study (Description of Fundamental theories and concepts of related project)
2.2. Literature Review (Review of the similar project done by other)

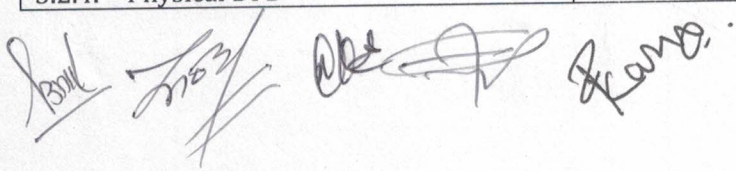
Chapter 3: System Analysis and Design

3.1. System Analysis
3.1.1. Requirement Analysis
  i. Functional Requirement (Use-case Diagram, Use-case Description)
  ii. Non Functional Requirement
3.1.2. Feasibility Analysis
  i. Technical
  ii. Operational
  iii. Economic
  iv. Schedule
3.1.3. System Modeling

Structural Approach	Object Oriented Approach
3.1.3.1. Data Modeling: ER Diagram 3.1.3.2. Process Modeling: DFD	3.1.3.1. Object Modeling: Class and Object Diagram 3.1.3.2. Dynamic Modeling: State and Sequence Diagram 3.1.3.3. Process Modeling: Activity Diagram

3.2. System Design

Structural Approach	Object Oriented Approach
3.2.1. Architectural Design 3.2.2. Database Schema Design 3.2.3. Interface Design (UI/UX) 3.2.4. Physical DFD	3.2.1. Refinement of Class and Object Diagram 3.2.2. Component Diagram 3.2.3. Deployment Diagram



Chapter 4: Implementation and Testing

4.1. Implementation
4.1.1. Tools Used (CASE tools, Programming Language, Database Platforms)
4.1.2. Implementation Details of Modules (Description of Procedures/Functions)
4.2. Testing
4.2.1. Test Case for Unit Testing
4.2.2. Test Case for System Testing

Chapter 5: Conclusion and Future Recommendations

5.1. Lesson Learnt and Outcome
5.2. Conclusion
5.3. Future Recommendations

Note: *While writing above chapters students should avoid basic definitions. They should relate and contextualize the above mentioned concepts with their project work.*

11. Citation and Referencing

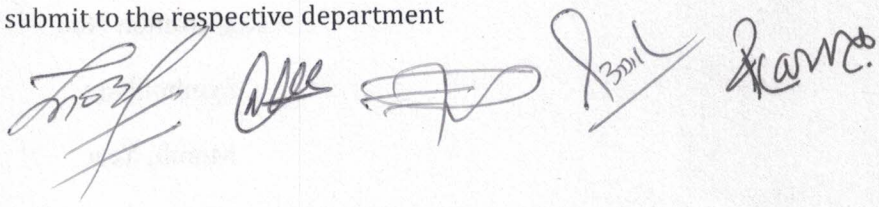
The listing of references should be listed in the references section. The references contain the list of articles, books, url of primary source that are cited in the document. The books, articles, and others that are studied during the literature review and project work but are not cited in the document can be listed in the bibliography section.
The citation and referencing standard should be **IEEE** referencing standard.

12. Report Format Standards

- **Page size:** A4
- **Font:** Times New Roman, 12pt (body), headings bold (16pt/14pt/12pt).
- **Spacing:** 1.5 line, justified.
- **Margins:** Top/Bottom/Right = 1", Left = 1.25".
- **Page Numbers:** Roman (i, ii...) for preliminaries, numeric (1, 2...) from Chapter 1.
- **Headings:** Chapter – 16pt (Bold and center align), Section – 14pt (Bold and left align), Sub-section – 12pt (Bold and left align), Paragraph content – 12pt (Regular)
- **Figures/Tables:** Center-align, Captions – 12pt (Bold)

13. Binding & Submission

- Copies: 2 (College Library, Self)
- Binding: Plain book binding
- Final Copy: Signed final copy submit to the respective department



(Sample of Cover and Title Page)

Lumbini Technological University

Institute of Engineering and Information Technology

TITLE OF PROJECT REPORT

A PRPJECT REPORT

Submitted to

Department of B.Tech in Information Technology

Name of the Collage

In partial fulfillment of the requirement for the B.Tech in Information Technology

Submitted by

Name:

Registration No.:

Symbol No.:

Month, Year

Under the Supervision of

Supervisor Name

Designation

(Sample of Certificate)

Lumbini Technological University
Institute of Engineering and Information Technology
College Name

Supervisor's Recommendation

I, hereby recommend that this project is completed under my supervision by NAME OF THE STUDENT (Year II Semester IV) entitled **"TITLE OF THE PROJECT"** in partial fulfillment of the requirements for the degree of B.Tech in Information Technology is recommended for the final evaluation.

.....
Signature
Name of Supervisor
Designation of Supervisor
Department Name
Address of the college

(Sample of Approval)

Lumbini Technological University

Institute of Engineering and Information Technology

College Name

LETTER OF APPROVAL

This is to certify that this project is completed by NAME OF THE STUDENT (Year II Semester IV) entitled "**TITLE OF THE PROJECT**" in partial fulfillment of the requirements for the degree of B.Tech in Information Technology has been evaluated. In our opinion it is satisfactory in the scope and quality as a project for the required degree.

.....
Signature
Name of Supervisor
Academic Designation
Department Name
Address of the college

.....
Signature
Name of HOD/Program Coordinator
Academic Designation
Department Name
Address of the college

.....
Signature
Name of Internal Examiner
Academic Designation

.....
Signature
Name of External Examiner
Academic Designation

Year III Semester V

Artificial Intelligence

Course Code	BIT301	Year/Semester	III/V
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hrs/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

This course provides an introduction to the principles, techniques, and applications of Artificial Intelligence (AI). It covers fundamental AI concepts, including intelligent agents, problem-solving, knowledge representation, machine learning, natural language processing, and robotics. The course also explores the ethical and societal implications of AI.

2. Course Objectives

The course aims to achieve the following objectives: The main objective of the course is to introduce students to the key concepts and methodologies of Artificial Intelligence. By the end of the course, students should be able to design basic AI systems and understand the potential impact of AI on society.

3. Course Details

3.1 Theory (48 hrs)

Unit 1: Introduction to Artificial Intelligence (4 hrs)

What is AI? Definitions and perspectives; History and evolution of AI; AI in the modern world; Applications and real-world examples; Ethical considerations in AI development and deployment.

Unit 2: Intelligent Agents (4 hrs)

Introduction to intelligent agents; Components of intelligent agents: Perception, reasoning, and action, environment types, agent types; multi-agent systems: Coordination and cooperation.

Unit 3: Problem Solving and Searching (10 hrs)

Problem-solving as search: State space representation; Uninformed search algorithms: Breadth-First Search, Depth-First Search, Depth Limited Search, Iterative Deepening and Uniform-Cost Search; Informed search algorithms: Greedy Best First Search, A* Search, Hill Climbing; Constraint Satisfaction Problems and its examples; Adversarial Search: Minimax, Alpha-Beta Pruning.

Unit 4: Knowledge Representation and Reasoning (10 hrs)

Knowledge representation; Rule based, Semantic Networks, Frames; Logic-based; Propositional Logic: Syntax, Semantics, and Resolution; First-Order Logic: Syntax, Semantics, and Resolution; Uncertainty and Probabilistic Reasoning: Probability Distribution, Baye's Theorem Bayesian Networks, Markov Models; Fuzzy Logic and Fuzzy Systems: Handling imprecisions.

Unit 5: Machine Learning (8 hrs)

Introduction to Machine Learning, Supervised, Unsupervised and Reinforcement Learning; Naïve Baye's Classification, K-means Clustering; Artificial Neural Network, Activation Functions, Perceptron Learning, Back Propagation, Hopfield Network; Genetic Algorithm.

Unit 6: Application of AI (NLP) **(8 hrs)**

Expert System; Introduction to NLP and its challenges; Text preprocessing: Tokenization, POS tagging, and Named Entity Recognition; Language models and NLP tasks: Sentiment Analysis, Text Classification, and Machine Translation; Introduction to Robotics and AI applications in Robotics; Robot Perception, Robot Control; Computer Vision.

Unit 8. Future Trends and Ethical Implications **(4 hrs)**

Emerging trends in AI: Explainable AI, and AI for Good, Generative AI; Ethical considerations in AI development: Bias, Privacy, and Accountability; Societal impact and responsible AI.

3.2 List of Tutorial **(16 hrs)**

Unit-specific exercises focused on problem-solving

3.4 Laboratory Works **(32 hrs)**

The practical works must cover the implementation of following concepts;

- Intelligent agents (simple reflex, model based and utility based)
- State space graphs
- Informed and uninformed search techniques
- Game search implementation using minimax
- Rule based systems
- Frames using the concepts of classes
- Predicate logic
- Naïve Bayes Algorithm
- K-means Algorithm
- ANN for AND and OR gate simulation
- Perceptron and Backpropagation Learning
- Implementing NLP concepts using NLTK.

In addition to above mentioned concepts, the laboratory work should include project work including expert systems, question answering systems etc. The choice of programming language can be as per the skills however the concepts of predicate logic should be implemented using logic based programming.

4. Evaluation Scheme

Evaluation of student's performances is divided into two main parts: (1) Internal Evaluation and (2) Semester examination, with breakdowns as follows:

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal Evaluation	Class Attendance and Performance	5	20	10
	Assignment	10		
	Seminar/Project/Presentation	5		
	Practical/Lab Examination	20	20	10
	Internal Examination	**20**	**20**	**10**
	Total Internal Marks		**60**	**30**
Semester-End Examination			**40**	**20**

5. Recommended Books

- Artificial Intelligence: A Modern Approach; Stuart Russell and Peter Norvig; Pearson
- Artificial Intelligence: Kevin Knight, Elaine Rich, Shivshankar B Nair, Mc Graw Hill
- Introduction to Artificial Intelligence: Intelligence and Expert Systems; Dan W. Patterson, Pearson
- Artificial Intelligence with Python: Your complete guide to building intelligent apps using Python 3.x; Alberto Artasanchez, Prateek Joshi; Packt Publishing

Software Engineering

Course Code	BIT302	Year/Semester	III/V
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hrs/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

This course is designed to provide students with a fundamental understanding of state of arts of Software Engineering. The course covers different aspects of software development, software maintenance and software trends in software industries.

2. Course Objectives

The course aims to achieve the following objectives:

- Introduce students to the fundamental concepts and technologies that underpin the Software Engineering
- Provide a solid basic foundation of Software Development approaches.
- Familiarize students with different software development approaches.

3. Course Detail

3.1 Theory

(48 Hrs)

Unit 1. Introduction **(3 Hrs.)**

Professional Software Development; Software Engineering Ethics

Unit 2: Software Processes **(5 Hrs.)**

Software Process Models; Process Activities; Coping with Change; Process Improvement

Unit 3: Agile Software Development **(4 Hrs.)**

Agile Methods; Agile Development Techniques

Unit 4: Requirements Engineering **(4 Hrs.)**

Functional and Non-functional Requirements; Requirements Engineering Processes; Requirements Elicitation; Requirements Specification; Requirements Validation; Requirements Change

Unit 5: System Modeling **(6 Hrs.)**

Context Models; Interaction Models; Structural Models; Behavioral Models; Model-Driven Architecture

Unit 6: Architectural Design **(5 hrs.)**

Architectural Design Decisions; Architectural Views; Architectural Patterns; Application Architectures

Unit 7: Design and Implementation **(6 Hrs.)**

Object-oriented Design using the UML; Design Patterns; Implementation Issues; Open-source development

Unit 8: Software Testing **(4 Hrs.)**

Development Testing; Test-driven Development; Release Testing; User Testing

Unit 9: Software Evolution **(4 Hrs.)**

Evolution Processes; Legacy Systems; Software Maintenance

Unit 10: Software Management **(7 Hrs.)**

Project management (Risk Management, Managing People, Teamwork); Project Planning (Software Pricing, Plan-driven Development, Project Scheduling, Estimation Techniques, COCOMO); Quality Management (Software Quality, Software Standards, Reviews and Inspections, Quality Management and Agile Development, Software Measurement); Configuration management (Version Management, System Building, Change Management, Release Management).

3.2 List of Tutorial

Unit-specific exercises focused on problem-solving

3.3 Laboratory Work

The practical work includes following activities:

1. Writing the problem statement for any system.
2. Preparing software requirement specification (SRS) for any system.
3. Drawing all UML diagrams using appropriate tools.
4. Learning to use CASE and project management tools.
5. Learning to design and run the test cases using appropriate tools.

In addition to above mentioned activities, students should prepare project report for some system including different activities of software process model.

4. Evaluation Scheme

Evaluation of student's performances is divided into two main parts: (1) Internal Evaluation and (2) Semester examination, with breakdowns as follows:

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal Evaluation	Class Attendance and Performance	5	20	10
	Assignment	10		
	Seminar/Project/Presentation	5		
	Practical/Lab Examination	20	20	10
	Internal Examination	**20**	**20**	**10**
	Total Internal Marks		**60**	**30**
Semester-End Examination			**40**	**20**

5. Recommended Books:

- Ian Sommerville (2016). Software Engineering, 10th Edition, Pearson.
- Roger S. Pressman and Bruce R. Maxim (2019). Software Engineering - a practitioner's approach, 9th Edition, McGraw-Hill Education.

E-Commerce

Course Code	BIT303	Year/Semester	III/V
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hrs/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

This course is designed to introduce students to the world of e-commerce, providing them with the knowledge and skills needed to understand, develop, and manage online businesses. It covers various aspects of e-commerce, including business models, technology, marketing, and ethical considerations. This course is outlined to provide students with a comprehensive understanding of e-commerce principles, strategies, and best practices. In addition to that this course provides an in-depth understanding of digital marketing techniques and strategies. It covers a wide range of topics, including search engine optimization (SEO), social media marketing, email marketing, content marketing, pay-per-click (PPC) advertising, and analytics. Students will gain practical knowledge and skills to create and manage effective digital marketing campaigns.

2. Course Objectives

By the end of this course, students should be able to:

- Understand the fundamental concepts and types of e-commerce.
- Evaluate e-commerce business models and strategies.
- Create and manage an online store using e-commerce platforms.
- Develop effective digital marketing and promotion strategies for e-commerce.
- Explore the legal and ethical aspects of e-commerce.
- Analyse the impact of emerging technologies on e-commerce.
- Understand the fundamentals of digital marketing and its importance in the modern business landscape.
- Plan, create, and execute digital marketing strategies across various channels.
- Use tools and analytics to measure and optimize digital marketing campaigns.
- Develop creative content and advertising that resonates with target audiences.
- Stay updated with industry trends and emerging technologies in digital marketing.

3. Course Details

3.1 Theory **(48 Hrs)**

Unit 1: Introduction of e-Commerce **(5 Hrs)**

Overview of e-commerce and its evolution; e-Business Vs e-Commerce; Service vs Product of e-Commerce; Framework for digital business- technology framework; Types of e-Commerce; Features of e-Commerce; E-Commerce enablers – Gold Rush model.

Unit 2: e-Commerce Infrastructure **(7 hrs)**

The internet; infrastructure; Internet today and future; Web and its function, the mobile platform; Web 2.0 its features and benefit for E-commerce; Web 3.0 and future internet; Cloud computing and

its relation to E-Commerce; Mobile apps and its importance for E-commerce, Chatbot and Software agents and its relevance to E-commerce.

Unit 3: e-Commerce Business Models **(8 hrs)**

B2B Business Model: E-distributor, E-procurement, Exchanges, Industry Consortia, Private Industrial Networks, Electronic Data Interchange and its relation to B2B, Supply chain and collaborative commerce and its relationship to B2B.

B2C Business Model: E-tailor Model, Community Provider, Content Provider, Portal, Transaction Broker, Market Creator, Service Provider.

Unit 4: E-commerce Platform **(7 Hrs)**

Options to enter to the E-Commerce market; Building E-commerce presence; E-commerce Web site, site for mobiles; Web Store, Mobile Store; Social networking Store, Choosing right platform, Choosing hardware and software.

Unit 5: Payment System and Security issue in e-commerce **(7 Hrs)**

Types of Payment Systems: Card, Cash on Delivery; Online vs offline payment; Payment Gateway; Emerging trends in Payment system; Mobile Payment, Payment system in Nepal; Security threats in e-commerce environment and payment; Technology solutions, Management policies, business procedure and public law (focus on Nepalese law).

Unit 6: Digital Marketing **(9 Hrs)**

The Internet Audience and Consumer Behavior; Digital marketing and advertisement strategy; Digital marketing tools, Internet marketing technology; Social marketing, Mobile marketing and Local Marketing; Search Engine Optimization- focus on Google search engine; Mobile Marketing, E-marketing; Internet marketing strategy: content marketing; Web Analytics using Google, Online content and media; Social Medial analytics: data type and collection, structured and semi-structured data, social media metrics, social medial ROI, Social networks and social network analysis; Social media analytics with unstructured data: text mining, social customer relationship management, text mining for communication and reputation management; Big data, Internet of things

Unit 7: Ecommerce and digital marketing in Action **(5 Hrs)**

Nepalese E-Commerce and digital Market; Nepalese E-Commerce and digital marketing trend; Major players analysis; Opportunity and risk; Payment service providers in Nepal.

3.2 List of Tutorial **(16 Hrs)**

Unit-specific exercises focused on problem-solving

3.4 Laboratory Work **(32 Hrs)**

- Set up an online store using an e-commerce platform.
- Create a product catalog, manage inventory, and handle transactions.
- Develop and execute digital marketing campaigns for e-commerce.
- Analyze the performance of an e-commerce website and make data-driven improvements.
- Navigate challenges and emerging trends in the e-commerce industry.

4. Evaluation Scheme

Evaluation of student's performances is divided into two main parts: (1) Internal Evaluation and (2) Semester examination, with breakdowns as follows:

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal Evaluation	Class Attendance and Performance	5	20	10
	Assignment	10		
	Seminar/Project/Presentation	5		
	Practical/Lab Examination	20	20	10
	Internal Examination	**20**	**20**	**10**
	Total Internal Marks		**60**	**30**
Semester-End Examination			**40**	**20**

5. Recommended Books

- Laudon and Traver (2020), Ecommerce – Business, Technology and Society 16th Edition; Pearson
- Turban, Lee, King and Chung(2010) Electronic Commerce, 6th edition, Pearson
- Kalakota and Whinston (1996), Frontiers of Electronic Commerce, Pearson
- "E-commerce: Business, Technology, Society" by Paul T. Kidd and Paul W. Rayport:

Data Science

Course Code	BIT304	Year/Semester	III/V
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hrs/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

The course "Fundamentals of Data Science" aims to provide students with a comprehensive understanding of the field of Data Science. It cater students to covering a wide range of topics, including data manipulation, statistics, machine learning, and data visualization. Through theoretical concepts, practical exercises, and real-world projects, students will gain hands-on experience and knowledge essential for a successful career in Data Science. It is a laboratory course in which students will develop programming skills in loading, cleansing, transforming, modeling and visualizing data.

2. Course Objectives:

- Introduce students to the fundamental concepts and principles of Data Science.
- Provide a solid foundation in data manipulation and analysis using Python and other essential tools.
- Familiarize students with statistical concepts and their application in data analysis.
- Teach students various machine learning algorithms and techniques for predictive modeling.
- Enable students to work with big data using distributed computing frameworks.

3. Course Details

3.1 Theory **(48 Hrs)**

Unit 1: Fundamentals **(5 hrs)**

Python Programming and tools; Data Manipulation using pandas; Exploratory data analysis; Data Visualization using Matplotlib, pandas, visdom; Text processing with nltk.

Unit 2: Introduction to Statistics **(10 Hrs)**

Descriptive Statistics and Data visualization; Sampling and estimation; Hypothesis Testing; Analysis of Variance; Correlation Analysis.

Unit 3: Regression and Anova **(5 Hrs)**

Linear Regression; Multiple Linear Regression; ANOVA.

Unit 4: Data Analysis and Visualization **(10 Hrs)**

Exploratory Data Analysis; Deriving insights from Data with Python; Missing Value Analysis and Outlier Detection; Introduction to Tableau or Power BI; Data Visualization with Tableau or Power BI.

Unit 5: Machine Learning **(10 Hrs)**

Scikit-learn and PyTorch; Introduction to Supervised Learning; Linear Regression, Logistic

Regression; Naïve Bayes, Decision Tree, Random Forest; Support Vector Machine; Introduction to Neural Networks

Unit 6: Machine Learning in the Cloud **(8 Hrs)**

Introduction to Data Science and Machine Learning on Cloud Platforms (AWS, Microsoft Azure, Google Cloud); Deploying Machine Learning Models on the Cloud

3.2 List of Tutorials

Tutorial should cover the problem solving excise of each unit.

3.3 Laboratory Work

The practical component will include laboratory work corresponding to each lecture unit. Students will work on individual and group projects to demonstrate their ability to conceptualize, design, and develop real-time data science applications.

4. Evaluation Scheme

Evaluation of students' performance is divided into two main parts: (1) internal evaluation and (2) semester examination, with breakdown as follows.

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal Evaluation	Class Attendance and Performance	5	20	10
	Assignment	10		
	Seminar/Project/Presentation	5		
	Practical/Lab Examination	20	20	10
	Internal Examination	**20**	**20**	**10**
	Total Internal Marks		**60**	**30**
Semester Examination			**40**	**20**

5. Recommended Books:

- "Python for Data Analysis" by Wes McKinney
- https://wesmckinney.com/book/

Reference Books:

- "Introduction to Statistical Learning" by Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani
- "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow" by Aurélien Géron Geron, A. (2017),
- Hands-On Machine learning with Scikit-Learn & TensorFlow, O'Reilly. McKinney, W. (2013), Python for Data Analysis, O'Reilly.
- VanderPlas, J. (2016), Python Data Science Handbook: Essential Tools for Working with Data, O'Reilly.

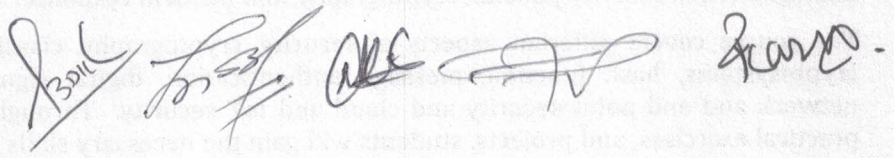
Journals and Magazines:

- IEEE Transactions on Knowledge and Data Engineering, IEEE

- ACM/IMS Transactions on Data Science, ACM
- Journal of Machine Learning Research (JMLR), Microtome
- Neural Networks, Elsevie

Best Learning Websites and Blogs:

- Python tutorials available online: https://docs.python.org/3/tutorial/
- Jupyter notebook tutorials available online: https://ipython.org/documentation.html
- Numpy tutorials available online:https://numpy.org/doc/stable/
- Pandas tutorials available online: https://pandas.pydata.org/docs/
- Nltk tutorials available online: https://www.nltk.org
- Matplotlib tutorials available online: https://matplotlib.org/contents.html
- Visdom tutorials available online: https://github.com/facebookresearch/visdom
- Scikit-learn tutorials available online: https://scikit-learn.org/stable/user_guide.html
- Pytorch tutorials available online: https://pytorch.org/tutorials/
- Kaggle (www.kaggle.com)
- Data Science Central (www.datasciencecentral.com



Information Security

Course Code	BIT305	Year/Semester	III/V
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hrs/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

This course introduces the fundamentals of information security, covering the protection of data, information systems, and network infrastructures. It explores various aspects of security, from risk management to security policies, cryptography, and incident response.

The course covers different aspects of security, cryptography, classical cryptosystems, modern cryptosystems, hash function, message authentication, digital signature, user authentication, network and end point security and cloud and IoT security. Through a combination of lectures, practical exercises, and projects, students will gain the necessary skills to secure computer systems using cryptographic algorithms and network security protocols.

2. Course Objectives

The course aims to achieve the following objectives:

- Introduce different concepts computer, information and network security.
- Gain insights about the principles of information security, including its importance and significance in modern organizations.
- Identify common security threats, vulnerabilities, and risks.
- Develop and implement security policies and practices.
- Utilize encryption and access control techniques to protect data and systems.
- Detect, respond to, and mitigate security incidents and breaches.
- Stay informed about emerging trends and technologies in information security.
- Provide concepts of classical ciphers
- Familiarize with private key cryptography
- Familiarize with public key cryptography
- Provide concepts of hash functions and message authentication codes
- Present concepts of digital signature for message authentication

3. Course Details

3.1 Theory (48 Hrs)

Unit 1: Introduction (5 Hrs)

Security: Computer Security, Information Security, Network Security; CIA Triad; Cryptography: Block Vs Stream, Public Vs Private; Cryptanalysis; Security Threats and Attacks; Security Services; Security Mechanisms, The importance of information security in organizations.
Key challenges and threats in the digital world.

Unit 2: Classical and Modern Ciphers **(9 Hrs)**

Classical Cryptosystems: Ceasar, Monoalphabetic, Playfair, Hill, Polyalphabetic, Rail Fence; Modern Ciphers: Data Encryption Standard (DES), Triple DES; Finite Fields: Groups, Rings, Fields, Galois Fields (GF(p) & $GF(2^n)$), Polynomial Arithmetic; Advanced Encryption Standard (AES); Number Theory: Prime Numbers, Fermat's and Euler's Theorem, Primality Testing, Euclidean Algorithm, Extended Euclidean Theorem; Discrete Logarithms; Diffie-Helman Key Exchange; RSA Algorithm

Unit 3: Hash Functions, Message Authentication Codes and Digital Signatures **(8 hr)**

Hash Functions; Message Digests: MD5; Secure Hash Algorithms: SHA-2, SHA-3; Message Authentication, Message Authentication Functions Message Authentication Codes, HMAC; Digital Signatures: Direct Digital Signatures, Arbitrated Digital Signature; Digital Signature Algorithm; RSA as a Digital Signature

Unit 4: User Authentication **(5 Hrs)**

Access Control and Authentication: Principles of access control, Authentication methods and technologies, Role-based access control (RBAC) and identity management, Authentication System; Password Based Authentication; Token Based Authentication; Biometric Authentication; Remote User Authentication; Kerberos Protocol

Unit 5: Network Security **(10 Hrs)**

Web Security; Transport Layer Security; HTTPS; Wireless Network Security: Wireless Security, Mobile Device Security, IEEE 802.11 Wireless LAN Overview, IEEE 802.11i Wireless LAN Security; E-mail Security: Email Architecture, E-mail Formats, Email Threats and Comprehensive Email Security, S/MIME, Pretty Good Privacy (PGP), DNSSEC; IP Security: IP Security Overview, IP Security Policy, Authentication Header, Encapsulating Security Payload, Security Associations

Unit 6: Network Endpoint Security **(5 Hrs)**

Firewall; Intrusion Detection System; Malicious Software; Denial of Service Attack; Distributed Denial of Service Attack

Unit 7: Cloud and IoT Security **(6 Hrs)**

Exploring the impact of cloud computing and IoT on security; The role of artificial intelligence and machine learning in security; Privacy and ethical considerations in the age of data; Cloud Computing; Cloud Security Concepts; Cloud Security Risks and Counter Measures; Cloud Security as a Service; Internet of Things; IoT Security Concepts and Objectives

3.2 List of Tutorial **(16 Hrs)**

Unit-specific exercises focused on problem-solving

3.3 Laboratory work **(32 Hrs)**

The practical work includes implementing the concepts of cryptographic algorithms, hash functions, digital signatures. The practical work extends implementation and simulation of Network Security Protocols, Intrusion Detection Systems, DDoS Attacks, Cloud Security and IoT Security Systems. Students are free to use any of the language, tools and platform as per the skills.

4. Evaluation Scheme

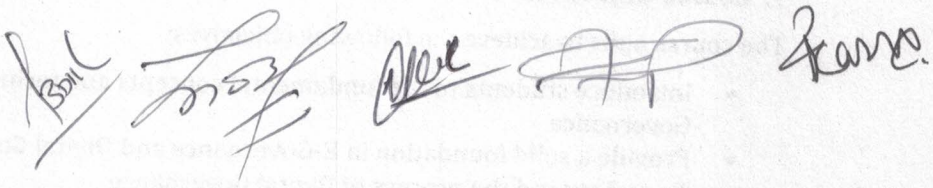
Evaluation of student's performances is divided into two main parts: (1) Internal Evaluation and (2) Semester examination, with breakdowns as follows:

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal Evaluation	Class Attendance and Performance	5	20	10
	Assignment	10		
	Seminar/Project/Presentation	5		
	Practical/Lab Examination	20	20	10
	Internal Examination	**20**	**20**	**10**
	Total Internal Marks		**60**	**30**
Semester-End Examination			**40**	**20**

5. Recommended Books (4-10 books):

- Cryptography and Network Security: Principles and Practice; William Stallings; Eighth Global Edition; Pearson
- Network Security Essentials: Applications and Standards; William Stallings; Sixth Edition; Pearson
- Cryptography & Network Security; Behrouz. A. Forouzan; First Edition; Mc Graw Hill Education
- Computer Security: Principles and Practice; William Stallings and Lawrie Brown; Fourth Edition; Pearson
- Network Security and Cryptography; Sarhan M. Musa; Mercury Learning and Information
- Computer Network Security Fundamentals; Joseph Migga Kizza; Fifth Edition, Springer

Year III Semester VI



Digital Governance

Course Code	BIT351	Year/Semester	III/VI
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hrs/wk
		Practical	-
		Total	64 hrs

6. Course Description

This course is designed to provide students with a comprehensive understanding of Digital Governance and the underlying technologies use in digital governance

The course covers various aspects of Digital Governance, Components of digital Governance, digital government Infrastructure development, Security for digital government, managing, implementing and applications of digital governance

Through a combination of lectures, case studies and projects, students will gain the necessary skills to implement digital governance project.

7. Course Objectives

The course aims to achieve the following objectives:

- Introduce students to the fundamental concepts and technologies that underpin the Digital Governance
- Provide a solid foundation in E-Governance and Digital Governance
- To understand the process of Digital Governance
- To conceptualize the Digital Governance Development Life Cycle.
- To familiarize Project Management framework and Tools use in Digital Governance
- To familiarize Security for Digital governance
- To design and develop the planning, scheduling, resource allocation, execution, tracking and delivery of Digital Governance
- To discuss the notation of risks and risk management process in digital Governance

8. Course Details

3.1 Theory **(48 Hrs)**

Unit 1: Introduction **(4 Hrs)**

Introduction to e-Governance, Digital-Governance , scope and importance of Digital Governance, issues and challenges of digital governance implementation, Digital governance evolution, global trends of growth in Digital Governance

Unit 2: Components of Digital Governance **(6 Hrs)**

Business process reengineering, Data Modelling, Metadata Management Data classifications & hierarchies, Naming & coding conventions Data Quality rules, policies & scorecards Master data

policies & workflows, E-Transformation, Whole of Government Perspectives ,Citizen Centric Approach, E-Inclusion, E-Leadership and Change Management

Unit 3: Technology and standards for Digital Governance **(5 Hrs)**

Enterprise Architecture, Enterprise Architecture Framework, Service Oriented Architecture, Enterprise Service Bus, service level agreement, Common Authentication Systems, Open Standards.

Unit 4: Digital Governance Infrastructure development **(6 Hrs)**

Network infrastructure, Computing Infrastructure, Data Centers, Digital Governance architecture, Interoperability Framework, Work Program in GOVIS 5.5 Impact and Improvement 5.6 ICT Implementation Challenges for good Governance

Unit 5: Security for Digital Governance **(6 Hrs)**

Challenges of Digital Governance security, an approach to security for Digital Governance, security management model, Digital Governance security architecture and standards, Secured Digital Identification

Unit 6: Managing and Implementing Digital Governance **(8 Hrs)**

Approaches to manage Digital Governance System, Key Performance Indicators for Digital Governmenance Systems, Management of Public Data, Digital Governance system life cycle, Assessment of Digital governance Projects, design of Digital Government system, Digital Governance Risk assessment and mitigation, Implementation and support, Public-Private Partnership

Unit 7: Emergent Technologies for Digital Governance **(5 Hrs)**

Cloud computing, Ubiquitous Governance, Enterprise Architecture, Service Oriented Architecture, Data warehousing and data mining

Unit 08: Case Study **(8 hrs)**

Case Study: Different areas of Digital Governance

3.2 Tutorial

Case studies: The case studies include related to digital governance of develop and developing countries.

4. Evaluation Scheme

Evaluation of student's performances is divided into two main parts: (1) Internal Evaluation and (2) Semester examination, with breakdowns as follows:

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal Evaluation	Class Attendance and Performance	5	40	20
	Assignment	15		
	Seminar/Project/Presentation	20		
	Practical/Lab Examination	-	-	-
	Internal Examination	20	20	10
	Total Internal Marks		**60**	**30**
Semester End Examination			**40**	**20**

5. Books

- Richard Heeks, Implementing and Managing e-Government
- C.S. R Prabhu, e-Governance: Concepts and Case studies, prentice hall of India Pvt. Ltd.
- J. Satyanarayana, e-Government, Prentice hall of India Pvt. Ltd
- Micheal E. Milakovich, Digital Governance, New Technologies for improving public service and participation,
- Yu-Che Chen, Managing Digital Governance: Issues, Challenges, and Solutions (ASPA Series in Public Administration and Public Policy) 1st Edition

Mobile Application Development

Course Code	BIT352	Year/Semester	III/VI
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hrs/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

This course is designed to provide students with a comprehensive understanding of Mobile development and the underlying technologies that provides the comprehensive knowledge related to Mobile programming, which encompasses integrated development environment, infrastructure, design, and development and testing of mobile applications, which communicate with databases to solve real word problems.

2. Course Objectives

The course aims to achieve the following objectives:

- Understand the fundamental concepts and principles of mobile development technology.
- The general objective of this course is to develop mobile applications that solve real word problems with use of current mobile technology.
- Apply security best practices in mobile development.
- Collaborate effectively in a team environment for mobile development projects.
- Analyze and troubleshoot common issues in mobile applications.

3. Course Details

3.1 Theory **(48 hrs)**

Unit -1: Introduction to Mobile and Mobile Programming **(2 hrs)**

Mobile Device (Features, Categories, History, Brands, Models and Platforms), Introduction to Mobile Programming.

Unit -2: Introduction to Android Programming **(2 hrs)**

Android Platform, History of Android, Environment Setup, Creating an android project, Laying out the user interface (The view hierarchy, widget attributes, creating string resources, previewing the layout), Creating a new class, Setting up the project, Running on the Emulator.

Unit -3: Designing the User Interface **(3 hrs)**

Android layout types (Linear, Relative, Table, Absolute, Constraint), Layout attributes, Android widgets (Textview, Edittext, Checkbox, Radiobutton, Spinner etc.) and its attributes, Event Handling, working with string, string array and colors, working with resources and drawable, adding icon to the project.

Unit -4: Android Activity **(3 hrs)**

The Activity life cycle, Creating multiple activities, Declaring activities in the manifest, Connecting activities with intents, Passing data between activities, Getting a result back from a child activity, Getting and setting data to/from the layout file.

Unit -5: UI Fragments, Menus and Dialogs **(4 hrs)**

The need for UI flexibility, Introduction to fragments, Lifecycle of fragment, Creating a UI fragment, Creating a fragment class, Wiring widgets in fragment, Introduction to fragment manager, Difference between Activity and Fragments.

Menus (Introduction, Types, Implementing menu in an application) Dialogs (Introduction, Creating a dialog fragment, Setting a dialog's content)

Unit -6: Listview, Gridview and Recyclerview **(4 hrs)**

Listview (Introduction, Features, Implementing listview in an application); Gridview (Introduction, Features, Implementing gridview in an application); Recyclerview (Introduction, Features, Implementing recyclerview in an application).

Unit -7: Advanced Android Concepts **(6 hrs)**

Local database with SQLite (Establishing connection, creating database and tables, data manipulation), Introduction to API, API Types, Introduction to JSON, Retrieving contents from remote server, Sending contents to remote server, Implementing Google Maps in android application, Procedure for publishing application on Google Play Store.

Unit -8: Introduction to ios Programming (8 hrs)

Introduction to ios and los programming, ios platform, Environment setup, Creating an Xcode project Building the interface, Making connections, Running on the simulator.

Introduction to Swift language, Views and the view hierarchy, Storyboard and view controllers, working with widgets and its attributes, Creating a simple ios application.

4. Practical

Laboratory Works

Laboratory works should be done covering all the topics listed above and a small project work should be carried out using the concept learnt in this course. Projects should be assigned on an individual basis.

5. Teaching Methods

The general teaching pedagogy includes class lectures, group discussions, case studies, guest lectures, research work, project work, assignments (theoretical and practical), and examinations (written and verbal), depending upon the nature of the topics. The teaching faculty will determine the choice of teaching pedagogy as per the need of the topics.

6. Tutorial

Unit specific programming practice exercise

7. Evaluation Scheme

Evaluation of student's performances is divided into two main parts: (1) Internal Evaluation and (2) Semester examination, with breakdowns as follows:

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal Evaluation	Class Attendance and Performance	5	20	10
	Assignment	10		
	Seminar/Project/Presentation	5		
	Practical/Lab Examination	20	20	10
	Internal Examination	**20**	**20**	**10**
	Total Internal Marks		**60**	**30**
Semester End Examination			**40**	**20**

8. Recommended Books:

1. Bill Phillips, Chris Stewart, Brian Hardy, and Kristin Marsicano, *Android Programming: The Big Nerd Ranch Guide,* Big Nerd Ranch LLC, 2nd edition, 2015.
2. 2. Christian Keur and Aaron Hillegass, *ios Programming: The Big Nerd Ranch Guide,* 5th edition, 2015.
3. Brian Fling, *Mobile Design and Development,* O'Reilly Media, Inc., 2009.
4. Maximiliano Firtman, *Programming the Mobile Web,* O'Reilly Media, Inc., 2nd ed., 2013.

Machine Learning

Course Code	BIT353	Year/Semester	III/VI
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hrs/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

This course is an advanced exploration of Machine Learning, designed to provide students with a deeper understanding of machine learning techniques and a wider variety of existing learning models. Building upon the foundational concepts covered in the Machine Learning course, students will delve into cutting-edge methods and applications in the field. Through a combination of lectures, practical exercises, and laboratory sessions, students will develop the skills and knowledge necessary to develop advanced machine learning applications and engage in research at a state-of-the-art level.

2. Course Objectives

The course builds on the content of Machine Learning, providing students with a deeper understanding of machine learning techniques and a wider variety of extant learning models. Students will be prepared to develop advanced machine learning applications and perform research at a state-of-the-art level.

3. Course Details

3.1 Theory

(48 hrs)

Unit 1: Introduction **(10 hrs)**

Introduction to machine Learning, Supervised Learning: Linear regression, logistic regression and generalized linear models, Generative probabilistic models, convex optimization and quadratic programming, support vector machines, Decision Trees and ensemble models, non-parametric methods.

Unit 2: Neural Network **(10 hrs)**

Perceptron's and inspiration from neuroscience, Multilayer neural networks and backpropagation, Optimization techniques, best practices, loss curve analysis, Inception modules, Residual Layers, squeeze and excitation, Detection models, semantic segmentation models, instance aware segmentation models.

Unit 3: Unsupervised Learning **(4 hrs)**

Clustering: k-means, Gaussian mixture models, principal component analysis, independent components analysis, Autoencoder.

Unit 4: Reinforcement Learning **(8 hrs)**

Markov decision processes and Bellman equations, Value iteration, policy iteration, Q-learning, Inductive transfer learning, Transudative transfer learning, Unsupervised transfer learning. Automated feature engineering, automated model selection and automated optimization algorithm selection.

Unit 5: Deep Unsupervised Learning **(8 hrs)**

Generative adversarial networks (GANs), Cycle GANs, Wasserstein GANs, Variational autoencoders, weight initialization, Dropout, Adam optimization, Batch normalization.

Unit 6: Time Series Processing **(4 hrs)**

Hidden Markov models (HMMs), Recurrent neural Networks (RNNs) and backpropagation through time, Word embedding for natural language processing, long short time memory (LSTM) units, Gated recurrent units (GRUs), Attention mechanisms for RNNs.

Unit 7: Deep Reinforcement Learning **(4 hrs)**

Policy gradients, Actor/critic methods, limitation learning, Exploration/exploitation, Meta learning, Monte Carlo methods.

3.2 Tutorials:

3.3 Laboratory Work

1. Speech recognition
2. Speech synthesis
3. Conversational agents
4. Recommendation systems
5. Anomaly detection
6. Computer vision system

4 Evaluation Scheme

Evaluation of student's performances is divided into two main parts: (1) Internal Evaluation and (2) Semester examination, with breakdowns as follows:

Evaluation	**Components**	**Weightage**	**Full Marks**	**Pass Marks**
Internal Evaluation	Class Attendance and Performance	5	20	10
	Assignment	10		
	Seminar/Project/Presentation	5		
	Practical/Lab Examination	20	20	10
	Internal Examination	**20**	**20**	**10**
	Total Internal Marks		**60**	**30**
Semester End Examination			**40**	**20**

5. Books

1. "Pattern Recognition and Machine Learning" by Christopher M. Bishop
2. "Deep Learning" by Ian Goodfellow, Yoshua Bengio, and Aaron Courville.

Cyber Security and Ethical Hacking

Course Code	BIT354	Year/Semester	III/VI
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hrs/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

This course introduces the fundamentals of information security, covering the protection of data, information systems, and network infrastructures. It explores various aspects of security, from risk management to security policies, cryptography, and incident response.
This course is designed to introduce students to the fundamentals of hacking and becoming an ethical hacker. The course focuses on the code of conduct and ethics of attacking systems. The course also teaches the mindset of the criminal hacker and evolution of the hacker.

2. Course Objectives

The course aims to achieve the following objectives:
The course objectives for Cyber security and Ethical Hacking programs can vary by institution, but generally, they include the following key goals:

- Grasp the core concepts of cyber security, including threats, vulnerabilities, and risk management
- Learn how to secure networks by implementing firewalls, intrusion detection/prevention systems, and secure configurations.
- Understand the methodologies used by ethical hackers, including reconnaissance, scanning, exploitation, and post-exploitation.
- Develop skills to assess and identify vulnerabilities in systems and networks.
- Familiarize with the legal and regulatory aspects of cyber security, including data protection laws and compliance standards.
- Learn how to effectively respond to security incidents and manage breaches, including forensics and recovery strategies.
- Understand best practices for developing secure applications and the importance of integrating security into the software development lifecycle.
- Evaluate and manage cyber security risks through proper assessment techniques and mitigation strategies.
- Gain practical experience through labs, simulations, and real-world scenarios to apply the knowledge learned.
- Stay updated with current trends, emerging threats, and new technologies in the field of cyber security.

3. Course Details

3.1 Theory (48 hrs.)

Unit 1: Introduction to Cyber Security (3 hrs)

Introduction to Cyber security and its history, Cyber security goals and principles, Types of cyber-attacks and attackers, cyber security technologies, cyber security policy and standards, Information

Security policies: Risk Management & Incident Management, Information Security controls, Data Management

Unit 2: Concept of Cryptography **(3 hrs)**

Introduction to Cryptography Concepts, Types of Cryptography, Encryption Algorithms: Ciphers, Data Encryption Standard, Advanced Encryption Standard, RC4, RC5, and RC6 Algorithms, Email, Disk Encryption and Cryptanalysis, Digital Signature, SSL, TLS, Cryptography Toolkit & Disk Encryption, Cryptography Attacks: Brute-Force Attack, Meet-in-the-Middle Attack on Digital Signature Schemes, Side Channel Attack, Hash Collision Attack, DUHK Attack, Rainbow Table Attack

Unit 3: Ethical Hacking Overview **(5 hrs)**

Ethical Hacking Overviews: Hacking concept and terminologies used in hacking, Need of Ethical hacking, Cases of Hacking in Nepal, Asia and across the globe

Principles of Ethical hacking: Basic Principle and Commandments of Ethical Hacking

Hacking Methodologies: Types of Hacking, Building the foundation for Ethical hacking, Phases of Hacking

Role of Ethical Hacker: Types of Hackers, Roles and Responsibilities,

Scope & limitations of hacking: Advantages & scope for hacking, Drawbacks & Limitation of hacking

Overview of PT / VA: Concept of Penetration testing, types of Penetration testing, Phases of Penetration testing, Vulnerability Assessment by Penetration testing

Unit 4: Malwares **(4 hrs.)**

Malware Overviews: Introduction to Malware, Different Ways a Malware can Get into a System, Common Techniques Attackers Use to Distribute Malware on the Web, Components of Malware

Virus Worm & Trojan Concepts: Introduction to Viruses, Worms & Trojan, Types of Virus, Worms &Trojan, Fake Antiviruses, How Did Antivirus Works.

Malware Analysis: Introduction to Malware Analysis, Malware Analysis Procedure, Malware Detection Method

Unit 5: Digital Foot Print **(8 hrs.)**

Foot printing, Enumeration, Scanning, Sniffing, Social Engineering; Foot printing through Search Engines, Web Services: Information Gathering Using Google Advanced Search and Image Search, VoIP and VPN Foot printing through Google, Hacking Database; Foot printing through Web Services: Finding Company's Top-level Domains (TLDs) and Sub-domains, Finding Location, People search, Professional search

Enumeration: Techniques for Enumeration, Services and Ports to Enumerate, NetBIOS, SNMP, LDAP Enumeration, Information Gathering Using Groups, Forums, and Blogs.

Scanning targets: Network Scanning Concepts, Scanning Tools and Techniques, Scanning Pen Testing, Port Scanning & Countermeasures; Sniffing Network: Sniffing Concepts & Techniques, Wire Shark installing & concept, Sniffing Detection Techniques; Social Engineering: Social Engineering Concepts, Social Engineering Techniques, Insider Threat/ Insider Attack, Social Engineering Countermeasures.

Unit 6: Cyber Attacks **(6 hrs.)**

SQL Injection, DOS Attacks, Session hijacking and System hacking: SQL Injection: Basics to the SQL queries, How Injection Can be done, Cross Site Scripting Attacks; DOS Attacks: DoS/DDoS Attack Techniques, DDoS Case Study, DoS/DDoS Attack Tools; Session Hijacking: Session Hijacking Concepts, Network Level Session Hijacking, Client side Hijacking; System Hacking: System Hacking; Concepts, Cracking Passwords, Escalating Privileges, Hiding Files and Covering tracks.

Unit 7: Web Application Attack and Counter Measures **(8 hrs.)**

Web Applications and Web Servers attacking methodology and Counter measures; Basics to Web application & threats: Basics of Web Application and Technology stack, OWASP Top 10 Application Security Risks – 2017, OWASP Top 10 Application Security Risks – 2017, Injection Flaws, File Injection Attack, Broken Authentication, Sensitive Data Exposure, XML External Entity (XXE), Broken Access Control, Security Misconfiguration, Cross-Site Scripting (XSS) Attacks, Insecure Deserialization, using Components with Known Vulnerabilities; Web Hacking methodology and tools: Web App Hacking Methodology and its foot printing, Password Attacks: Password Functionality Exploits, Password Guessing and Brute-forcing; Introduction to Web Server: Web Server Concepts, Web server attacks, Web Server Attack Methodology, Web Server Attack Tools ; Counter measures: Detecting Web Server Hacking Attempts, Patch Management, Web Server Security Tools

Unit 8: Cyber Security Measures **(6 hrs.)**

Firewall, IDS and honeypot evasion techniques, evasion tools, and countermeasures; Basics of Firewalls, IDS, Honeypots: What is Firewall & Examples, IDS & examples, Honeypots & examples, Case study of Bypassing Firewall, IDS, Firewall; IDS/Firewall/Honeypot Evasion Techniques: Packet Fragmentation and Source Routing, Working with SNORT [Detection Tool], IP address Decoy and Spoofing, IP Spoofing Detection Techniques: Direct TTL Probes, IP Spoofing Detection Techniques: IP Identification Number, IP Spoofing Detection Techniques: TCP Flow Control Method, IP Spoofing Countermeasures.

Unit 9: Wireless Network Attacks and Counter Measures **(5hrs.)**

Mobile & Wireless Networks Hacking and Counter measures; Mobile Hacking: Introduction to the Mobile architecture, working principle for mobile OS, Security of the application used in Mobiles, Hacking Methodology for mobile with Metasploit; Wireless Overview: Introduction to Terminology, Networks, Standards, Types of Wireless Authentication and Encryption, WEP (Wired Equivalent Privacy) Encryption, WPA (Wi-Fi Protected Access) Encryption, WPA2 (Wi-Fi Protected Access 2) Encryption; Wireless Hacking Methodology: Wi-Fi Discovery, GPS Mapping, Traffic canalizing, Launch Wireless attacks, Crack Wi-Fi Encryption

3.1 Tutorials

Unit-specific exercises focused on problem-solving, case study and particularly analysis of problems.

3.3 Laboratory work

The practical work includes implementing the concepts of cryptographic algorithms, hash functions, digital signatures. The practical work extends implementation and simulation of Network Security Protocols, Intrusion Detection Systems, DDoS Attacks, Cloud Security and IoT Security Systems. Students are free to use any of the language, tools and platform as per the skills.

List of Experiments:

1. List the tools for Ethical Hacking.
2. Implement Foot printing and Reconnaissance using tools 3d Traceroute, Alchemy Eye, DNS Tools and Network Solution Whois.
3. Implement Network Scanning using tools Advanced Port Scanner, Colasoft Ping Tool, Hide Your IP Address, Nessus and Nmap.

4. Implement Enumeration using tools Default Password List, Default Password List, OpUtil Network Monitoring Tool and OpUtil Network Monitoring Tool.
5. Implement system hacking using tools Actual spy, Alchemy Remote Executor, Armor Tool and FSecure BlackLight.
6. Implement Trojan and Backdoors using tools Absolute Startup Manager, Absolute Startup Manager, Netwirx Services Monitor and StartEd Lite.
7. Implement Viruses and Worms using tools Anubis Analyzing UnknownBinaries, Filterbit, Sunbelt CWSandbox and ThreatExpert.

4. Evaluation Scheme

Evaluation of students' performance is divided into two main parts: (1) Internal evaluation and (2) Semester examination, with breakdown as follows.

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal Evaluation	Class Attendance and Performance	5	20	10
	Assignment	10		
	Seminar/Project/Presentation	5		
	Practical/Lab Examination	20	20	10
	Internal Examination	**20**	**20**	**10**
	Total Internal Marks		**60**	**30**
Semester Examination			**40**	**20**

5. Books

- Hacking: Beginner's Guide to Computer Hacking, Basic Security, Penetration Testing (Hacking, How to Hack, Penetration Testing, Basic security, Computer Hacking)
- Penetration Testing: A Hands-On Introduction to Hacking by Georgia Weidman
- Gray Hat Hacking: The Ethical Hacker's Handbook, Fifth Edition by Allen Harper
- Cryptography and Network Security: Principles and Practice; William Stallings; Eighth Global Edition; Pearson
- Cryptography & Network Security; Behrouz. A. Forouzan; First Edition; Mc Graw Hill Education
- Computer Security: Principles and Practice; William Stallings and Lawrie Brown; Fourth Edition; Pearson
- Network Security and Cryptography; Sarhan M. Musa; Mercury Learning and Information
- Computer Network Security Fundamentals; Joseph Migga Kizza; Fifth Edition, Springer
- The Tangled Web: A Guide to Securing Modern Web Applications by Michal Zalewski latest edition

Year IV Semester VII

IT Project Management

Course Code	BIT401	Year/Semester	IV/VII
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hrs/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

This course provides students with the knowledge and skills necessary to effectively plan, execute, and manage information technology (IT) projects. It covers project management principles, tools, and techniques specific to IT projects, emphasizing successful project delivery within scope, time, and budget constraints.

2. Course Objectives

- Understand the fundamentals of project management in an IT context.
- Define project scope, objectives, and requirements.
- Develop comprehensive project plans and schedules.
- Manage project resources, including human resources and technology.
- Identify, assess, and mitigate project risks.
- Monitor project progress, control changes, and ensure quality.
- Effectively communicate with project stakeholders.
- Apply various project management methodologies and frameworks.

3. Course Details

3.1 Theory **(48 Hrs)**

Unit 1: Project Management and ICT for Development **(6 Hrs)**

Introduction to IT Project Management: Importance, Project, IT Projects Vs Non-IT Projects, ICT4D Project, Project Management, Contract Management, and technical project management, Activities, Plan, Methods, and methodologies, some ways to categorizing software projects, management, Step wise – an overview of project planning: Introduction to stepwise project planning, select project, identify project scope and objectives, identify project infrastructure, analyze project characteristics, identify project products and activities, Programme management and project evaluation: Programme management, managing the allocation of resources within programmes, strategic programme management , evaluation of individual projects, technical assessment, cost benefit analysis, cash flow forecasting, cost benefit evaluation techniques, risk evaluation

Unit 2: Selection of IT Project **(10 Hrs)**

Selection of an appropriate project approach Choosing technologies, technical plan contents list, choice of process models, structure versus speed of delivery, the waterfall model, V-process model, the spiral model, prototypes, incremental delivery

Software efforts estimation, Where Estimate done? Problems with over and under estimates, the basis for software estimating, software effort estimation techniques, Expert judgment, estimating by analogy, Albrecht function point analysis, function points mark II, COSMIC full function points, COCOMO : a parametric model

Unit 3: Activity Planning **(9 Hrs)**

Introduction, Objectives of Activity Planning, When to Plan, Project Schedules, Projects and Activities, Sequencing and Scheduling Activities, Network Planning Models, Formulating a Network Model, Adding the Time Dimension, The Forward Pass, Backward Pass, Identifying the Critical Path, Activity Float, Shortening the Project Duration, Identifying Critical Activities, Activity-on-Arrow Networks.
Risk Management: Introduction, Risk, Categories of Risk, Risk Management Approaches, A Framework for Dealing with Risk, Risk Identification, Risk Assessment, Risk Planning, Risk Management, Evaluating Risks to the Schedule, Boehm"s Top 10 Risks and Counter, Measures, Applying the PERT Technique, Monte Carlo Simulation, Critical Chain Concepts.
Resource Allocation: Introduction, Nature of Resources, Identifying Resource Requirements, Scheduling Resources, Creating Critical Paths, Counting the Cost, Being Specific, Publishing the Resource Schedule, Cost Schedules, Scheduling Sequence.

Unit 4: Monitoring and Control **(6 Hrs)**

Introduction, Creating the Framework, Collecting the Data, Review, Visualizing Progress, Cost Monitoring, Earned Value Analysis, Prioritizing Monitoring, Getting the Project Back to Target, Change Control, Software Configuration Management (SCM). Managing Contracts: Introduction, Types of Contract, Stages in Contract Placement, Typical Terms of a Contract, Contract Management, Acceptance. Managing People in Software Environments: Introduction, Understanding Behaviour, Organizational Behaviour: A Background, Selecting the Right Person for the Job, Instruction in the Best Methods, Motivation, The Oldham–Hackman Job Characteristics Model, Stress, Stress Management, Health and Safety, Some Ethical and Professional Concerns.

Unit 5: IT Project Team **(9 Hrs)**

Working in Teams: Introduction, becoming a Team, Decision Making, Organization and Team Structures, Coordination Dependencies, Dispersed and Virtual Teams, Communication Genres, Communication Plans, Leadership.

Unit 6: Software Quality **(8 Hrs)**

Introduction, The Place of Software Quality in Project Planning, Importance of Software Quality, Defining Software Quality, Software Quality Models, ISO 9126, Product and Process Metrics, Product versus Process Quality Management, Quality Management Systems, Process Capability Models, Techniques to Help Enhance Software Quality, Testing, Software Reliability, Quality Plans.
Project Closeout: Introduction, Reasons for Project Closure, Project Closure Process, Performing a Financial Closure, Project Closeout Report.

3.6 Tutorials

Unit-specific exercises focused on problem-solving.

3.3 Laboratory Works

The practical work includes implementing the concepts of cryptographic algorithms, hash functions, digital signatures. The practical work extends implementation and simulation of Network Security Protocols, Intrusion Detection Systems, DDoS Attacks, Cloud Security and IoT Security Systems. Students are free to use any of the language, tools and platform as per the skills.

4. Evaluation Scheme

Evaluation of students' performance is divided into two main parts: (1) Internal evaluation and (2) Semester examination, with breakdown as follows.

Evaluation	**Components**	**Weightage**	**Full Marks**	**Pass Marks**
Internal Evaluation	Class Attendance and Performance	5	20	10
	Assignment	10		
	Seminar/Project/Presentation	5		
	Practical/Lab Examination	20	20	10
	Internal Examination	**20**	**20**	**10**
	Total Internal Marks		**60**	**30**
Semester-End Examination			**40**	**20**

5. Books:

- Bob Hughes, Mike Cotterell, Rajib Mall. *Software Project Management*. TMH, 6th edition, 2018.
- Shailesh Mehta. *Project Management and Tools & Technologies – An Overview*. SPD, 1st edition, 2017.
- Walker Royce. *Software Project Management*. Pearson, 2005.

Embedded Systems & Internet of Things (IoT)

Course Code	BIT402	Year/Semester	IV/VII
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hrs/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

This course includes design and implementation of embedded systems and Internet of Things (IoT). Students will gain hands-on experience with programming microcontrollers, designing embedded applications, and interfacing with various peripherals. This course is further designed to provide students with a comprehensive understanding of IoT and the essential technologies. The course covers various aspects of IoT system components, IoT Devices and IoT Gateways. It further includes fundamental information of IoT devices and platform for developing IoT Systems, Design methodology, Data Analytics for IoT, Servers and IoT System Security.

2. Course Objectives

- Introduce students to the fundamental concepts and technologies that outline the Embedded System and IoT System.
- Introduce the students about the principles, techniques, development and applications of Embedded System and IoT System.
- Familiarize students with the computers and network system, its components, data representation and to get them familiar with various important features of procedure oriented programming.
- Develop the ability to work collaboratively in a team environment, enabling students to effectively contribute to Embedded as well as IOT related projects, communicate ideas, and coordinate tasks.
- Develop students' skills in identifying and resolving common issues in Embedded and IOT applications through problem-solving techniques.
- Familiarize students with various interfacing techniques, including input/output devices.
- Develop problem-solving skills in designing and troubleshooting embedded systems.

3. Course Details

3.1 Theory **(48 hrs.)**

Unit 1: Characteristics of Embedded System and Hardware **(3 hrs)**

Ideal Characteristics, operational and non-operational quality attributes. Hardware: memory map, i/o map, interrupt map, external peripherals, memory –RAM, ROM, Flash memory, Interfacing.

Unit 2: Design and Development of Embedded System **(6 hrs)**

Embedded system development environment- IDE, Types of file generated on cross compilation, disassembler, decompiler, simulator, emulator and debugging, embedded product development life cycle, trends in embedded industry, Embedded System development environment - IDE, Types of file generated on cross compilation, disassembler / decompiler, simulator, emulator and debugging,

Unit 3: IC Technology for Embedded System **(3 hrs)**

Full-custom (VLSI) IC technology, Semi-custom (ASIC) IC technology, Programming Logic Device (PLD) IC technology, embedded product development life-cycle, Trends in embedded industry

Unit 4: IOT Introduction, Data Acquisition & Platforms **(6 hrs)**

Concepts and Definitions of IoT, IoT enabling technologies, Physical and Logical Design of IoT, IoT Architecture, Major component of IOT, Communication and networking protocols, IoT services and applications, IoT Standards, Connectivity, IoT and M2M, Platforms for IoT, Micro Controllers (Arduino uno/mega2560, Rasberry-Pi, ARM), Interfacing LED, push button and buzzer with Arduino, Interfacing Arduino with LCD, IoT System for Weather Monitor, Real-time systems, and embedded software, OS and Drivers (End Device Program), Hardware & Software Requirements, IOT Cloud platforms

Unit 6: Data Communication, Storage and Retrieval **(5 hrs)**

Transfer data by Wireless / Wired connectivity, Ipv4/Ipv6, Ethernet/GigE, MIPI, M-PHY, UniPro, SPMI, BIF, SuperSpeed USB Inter-Chip (SSIC), Mobile PCIe (M-PCIe) and SPI GSM , 2G-5G IEEE 802.15.4, IEEE 802.15.4e, 802.11ah, Target Wake Time (TWT), Posting sensor(s) data to web server, Role of Storage in Cloud / Server /Inhouse Storage, Databases Connectivity with IOT and uses

Unit 7: Data Analytics & Visualization **(5 hrs)**

Big data tools and technology, Edge Streaming Analytics, Network Analytics, Hadoop, Spark & Cassandra, Prototyping with TI CC3200 physical computing dev board, Programming WiFi Communication with Energia IDE & C Language, Using Cloud Services to Visualize live Data Streams, Analysis of data using the Ipython Module, Visualization and interpretation of Data, Data Cleaning in IoT

Unit 8: IoT Security **(4 hrs)**

Attack, Defense and Network Robustness of IoT, Malware Propagation and Control, Privacy Preservation Data Dissemination, Computational Security, Security Protocols, Security Testing, IoT Security Practices and Challenges, Symmetric Encryption and Non Symmetric Encryption Standards, Trust and Trust Models

3.2 Tutorials **(16 hrs)**

Unit-specific exercises focused on problem-solving.

3.3 Laboratory Works **(32 hrs)**

- Configure timer for delay function, I/O interfacing with led's, seven segments, sensors. Device control (motor control), Waveform generation (Square wave, sine wave).
- Implement the concept mentioned in the course using Python programming language, Arduino or Raspberry pi as a System board. All sensors mentioned in course should be implemented in a single project or separately to observe their working mechanism.

4. Evaluation Scheme

Evaluation of students' performance is divided into two main parts: (1) Internal evaluation and (2) Semester examination, with breakdown as follows.

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal Evaluation	Class Attendance and Performance	5	20	10
	Assignment	10		
	Seminar/Project/Presentation	5		
	Practical/Lab Examination	20	20	10
	Internal Examination	**20**	**20**	**10**
	Total Internal Marks		**60**	**30**
Semester-End Examination			**40**	**20**

5. Books:

- Simone Cirani, Gianluigi Ferrari, Marco Picone, Luca Veltri. Internet of Things: Architectures, Protocols and Standards. Wiley, 2019. p.383. ISBN 9781119359678.
- Perry Xiao. Designing Embedded Systems and the Internet of Things (IoT) with the ARM® Mbed™. Wiley, 2019. p.316. ISBN 9781119363996.
- ArshdeepBahga, Vijay Madisetti, " Internet of Things (A Hands-on-Approach)", University Press India Pvt. Ltd., 2015.
- David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Ro be rt Barton, Jerome Henry,"IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things", Pearson Education (Cisco Press Indian Reprint).
- Raj Kamal, "Internet of Things: Architecture and Design Principles", McGraw Hill Education, 2017. Mazidi,E. and Mazidi,F., The 8051 Microcontroller and Embedded Systems, Prentice-Hall of India (2004) 2nd ed.
- Frank Vahid, Tony Givargis, " Embedded System Design", John Wiley & Sons, 2008.

Block Chain Technology

Course Code	BIT403	Year/Semester	IV/VII
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hrs/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

This course provides a comprehensive introduction to Blockchain Technology, a revolutionary decentralized ledger system that underpins cryptocurrencies like Bitcoin and Ethereum. Students will delve into the foundational concepts of blockchain, its architecture, and the mechanisms that ensure security and immutability. The course covers various real-world applications of blockchain beyond cryptocurrencies, including supply chain management, voting systems, smart contracts, and decentralized finance (DeFi). Through practical examples, case studies, and hands-on exercises, students will gain a deep understanding of blockchain's potential, challenges, and the future impact it may have on various industries.

2. Course Objectives

- Understand the fundamental concepts of blockchain technology, including its history, principles, and components.
- Explore the architecture and working mechanisms of different blockchain systems.
- Analyze the security features of blockchain networks and evaluate their strengths and weaknesses.
- Examine the consensus algorithms used in blockchain networks and their impact on scalability and decentralization.
- Investigate real-world applications of blockchain in various sectors such as finance, healthcare, supply chain, and more.
- Gain practical experience in setting up and interacting with blockchain networks.
- Identify potential use cases for blockchain technology and evaluate its advantages and limitations in different scenarios.
- Discuss the legal, ethical, and regulatory challenges associated with blockchain implementation.
- Collaborate in groups to develop innovative blockchain-based solutions to real-world problems.
- Stay informed about the latest trends and developments in the ever-evolving field of blockchain technology.

3. Course Details

3.1 Theory (48 hrs)

Unit 1: Introduction to Blockchain (3 hrs)

Definition and characteristics of blockchain technology; Brief history and evolution of blockchain; Decentralization, transparency, and immutability.

Unit 2: Cryptography and Consensus Mechanisms (5 hrs)

Cryptographic principles used in blockchain (hashing, digital signatures, etc.); Public and private key cryptography; Consensus algorithms (Proof of Work, Proof of Stake, etc.).

Unit 3: Blockchain Architecture and Data Structure (4 hrs)

Understanding the blockchain data structure; Blocks, transactions, and Merkle trees; Blockchain nodes and their roles (miners, validators, etc.).

Unit 4: Ethereum and Smart Contracts **(5 hrs)**
Introduction to Ethereum blockchain; Ethereum Virtual Machine (EVM) and smart contract execution; Writing, deploying, and interacting with smart contracts.

Unit 5: Solidity Programming **(5 hrs.)**
Solidity basics and syntax; Handling data and variables in Solidity; Implementing control structures and functions.

Unit 6: Advanced Smart Contracts **(5 hrs.)**
Using OpenZeppelin for secure smart contract development; Design patterns for smart contracts; Developing complex smart contracts (e.g., ERC-20 token, multisig wallet).

Unit 7: Decentralized Applications (DApps) Development **(5 hrs.)**
Frontend development with web3.js or similar libraries; Integrating DApps with smart contracts; User interactions and Metamask integration.

Unit 8: Blockchain Interoperability **(4 hrs.)**
Understanding cross-chain communication; Interoperability protocols and solutions; Building interoperable applications.

Unit 9: Security and Privacy in Blockchain **(3 hrs.)**
Common security threats in blockchain networks; Best practices for secure smart contract development; Privacy considerations and solutions in blockchain.

Unit 10: Scalability and Performance **(3 hrs.)**
Challenges of blockchain scalability; Layer 2 scaling solutions (state channels, sidechains); Exploring sharding and other scalability approaches.

Unit 11: Real-world Blockchain Applications **(3 hrs.)**
Use cases in finance, supply chain, healthcare, and more; Case studies of successful blockchain implementations.

Unit 12: Future Trends and Emerging Technologies **(3 hrs.)**
Ethereum 2.0 and its impact on the blockchain ecosystem; DeFi (Decentralized Finance) and NFTs (Non-Fungible Tokens); Exploring emerging blockchain technologies.

3.7 Tutorials

Unit-specific exercises focused on problem-solving.

3.8 Laboratory Works

Lab 1: Setting Up the Development Environment
Objective: Install and configure the necessary tools and libraries required for blockchain development.
Tasks:

- Install a blockchain client (e.g., Geth, Parity, or Besu) on your local machine.
- Set up an Ethereum development environment using Remix IDE or Truffle.
- Configure MetaMask and connect it to your local development network.
- Create a new Ethereum account using MetaMask and fund it with test Ether.

Lab 2: Writing and Deploying Smart Contracts
Objective: Learn the basics of writing and deploying smart contracts on the Ethereum blockchain.
Tasks:

- Write a simple smart contract in Solidity that stores and retrieves data.
- Compile the smart contract using the Ethereum development environment.
- Deploy the smart contract to your local development network using Remix or Truffle.
- Interact with the deployed smart contract through Remix or a JavaScript frontend.

Lab 3: Creating a Decentralized Application (DApp)
Objective: Build a decentralized application (DApp) that interacts with a deployed smart contract.
Tasks:

- Design a simple DApp using HTML, CSS, and JavaScript.
- Integrate web3.js or ethers.js library to interact with the deployed smart contract.
- Implement functionalities such as reading data from the smart contract and sending transactions.
- Test the DApp on your local development network using MetaMask.

Lab 4: Implementing Smart Contract Security
Objective: Identify and fix common vulnerabilities in smart contracts.
Tasks:

- Explore common smart contract vulnerabilities (e.g., reentrancy, integer overflow, etc.).
- Analyze a vulnerable smart contract and identify its weaknesses.
- Implement security fixes to address the identified vulnerabilities.
- Re-deploy the updated smart contract and test its security.

Lab 5: Exploring Blockchain Interoperability
Objective: Learn about cross-chain communication and interoperability between different blockchain networks.
Tasks:

- Set up a test environment with two blockchain networks (e.g., Ethereum and Binance Smart Chain).
- Create a bridge between the two networks using a cross-chain communication protocol.
- Transfer assets (e.g., tokens) from one network to another using the bridge.
- Verify the successful transfer of assets and ensure consistency across both networks.

Lab 6: Blockchain Scalability and Layer 2 Solutions
Objective: Explore different Layer 2 scaling solutions and their impact on blockchain scalability.
Tasks:

- Investigate the scalability challenges faced by the Ethereum blockchain.
- Learn about Layer 2 solutions such as state channels or sidechains.
- Implement a simple state channel for off-chain transactions.
- Measure the performance improvement in terms of transaction throughput and gas fees.

Lab 7: Building an Ethereum Token
Objective: Create a custom ERC-20 token and deploy it on the Ethereum blockchain.
Tasks:

- Design the token contract in Solidity following the ERC-20 standard.
- Deploy the token contract to the Ethereum blockchain.
- Interact with the token contract using Remix or a DApp frontend.
- Test token transfers and verify token balances using MetaMask.

Lab 8: Exploring DeFi (Decentralized Finance) Applications
Objective: Explore and interact with decentralized finance (DeFi) applications on the Ethereum blockchain.
Tasks:

- Learn about popular DeFi protocols like Uniswap, Compound, or Aave.
- Connect to the DeFi protocols using MetaMask.
- Engage in token swaps, lending, or borrowing activities within the DeFi applications.
- Analyze the risks and benefits of using DeFi applications.

Lab 9: Ethereum Smart Contract Upgrades
Objective: Understand how to upgrade smart contracts while preserving data and user interactions.
Tasks:

- Study the challenges of upgrading smart contracts.
- Implement a smart contract with upgradable features using a proxy pattern.
- Perform upgrades on the upgradable smart contract, ensuring data and functionality persistence.
- Test the upgraded contract with various scenarios.

Lab 10: Final Project
Objective: Apply the knowledge gained throughout the laboratory work to develop a full-fledged blockchain application or smart contract project.
Tasks:

- Choose a suitable project idea related to blockchain technology (e.g., NFT marketplace, decentralized Digital voting system, Transparent Budgeting, Record Management, Healthcare, Media and Entertainment, Real State, supply chain tracking, Finance, Energy).
- Design the architecture and functionalities of the project.
- Implement the project using best practices and security measures.
- Present the final project to the class, explaining its features and demonstrating its functionality.

Note: The laboratory work can be adjusted based on the specific blockchain platform or framework being used in the course. Additionally, students may work in teams for certain tasks and collaborate on larger projects for the final assignment.

4. Evaluation Scheme

Evaluation of students' performance is divided into two main parts: (1) Internal evaluation and (2) Semester examination, with breakdown as follows.

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal Evaluation	Class Attendance and Performance	5	20	10
	Assignment	10		
	Seminar/Project/Presentation	5		
	Practical/Lab Examination	20	20	10
	Internal Examination	**20**	**20**	**10**
	Total Internal Marks		**60**	**30**
Semester Examination			**40**	**20**

5. Recommended Books

Textbooks:

- "Mastering Blockchain: A Hands-On Approach to Distributed Ledger Technology" by Imran Bashir
- "Blockchain Basics: A Non-Technical Introduction in 25 Steps" by Daniel Drescher

Reference Books:

- "Blockchain Revolution: How the Technology Behind Bitcoin Is Changing Money, Business, and the World" by Don Tapscott and Alex Tapscott

- "Blockchain for Dummies" by Tiana Laurence
- "Blockchain: Blueprint for a New Economy" by Melanie Swan

Learning Websites:

- (https://www.coursera.org/browse/computer-science/blockchain)
- (https://www.edx.org/learn/blockchain)
- (https://academy.b9lab.com/)
- (https://www.blockchain-council.org/)
- (https://developer.ibm.com/technologies/blockchain/)

Capstone Project

Course Code	BIT445	Year/Semester	IV/VII
Credit Weightage	5	Class Load	10 hrs/wk
		Practical	10 hrs/wk
		Total	160 hrs

1. Course Description

This is a fully practical course that expects students to implement concepts learned during the first three years of their study. The system should not be limited to the basic CRUD operations only, they are highly recommended to implement appropriate algorithms relevant to the project. The project should include precise system analysis, design, implementation, and result analysis. The students can choose appropriate language and technologies that they have learn till seven semester.

2. Course Objectives

The objective of this project work is to make the student able to:

- Lead a software project development
- Implement appropriate algorithms in the project
- Work in a team
- Use CASE tools
- Write programs and improve programming skills
- Write test cases for software testing and improve QA skills
- Enhance problem-solving, report writing, and presentation skills

3. Nature of Project

- It is an academic project focused on development of a desktop based, web based, or mobile based application using any programming language and technology of their expertise and comfort
- Students can develop the application not limited in CRUD operations but also have implement appropriate algorithm.
- Student should use appropriate CASE tools during SDLC
- While implementing the project, students should be encouraged to write their own modules rather than relying on APIs or Plugins (except in some unavoidable circumstances)
- Emphasis on problem-solving, system design, and implementation
- Project group size: Maximum 2 students in a group but highly recommended to do individually.

4. Focus of the Study

Each student in a group (in case of group) should have equal participation in every phase of the project. The students should focus on the each phases of SDLC in detail.

5. Phases of the Project Work

5.4. Proposal Submission and defense
- Timeline: 3rd - 4th week of the 7th Semester

5.5. Mid-Term Progress Report Evaluation and Defense
- Timeline: 11th –12th week of the 7th Semester

5.6. Final Submission and Defense
- Final report submission: At least 10 days before the final defense date

- Defense includes viva, presentation, project demonstration, and report evaluation

6. Supervision

- Supervised by a regular faculty member
- One supervisor may guide up to two groups or four individual

7. Evaluation Criteria

Total Marks: 100

Evaluation Phase	Evaluators	Marks
Proposal Defense	HoD/Coordinator (4), Supervisor (12), Internal Examiner (4)	20
Mid-Term	HoD/Coordinator (8), Supervisor (24), Internal Examiner (8)	40
Final Defense	HoD/Coordinator (5), Supervisor (15), Internal Examiner (5), External Examiner (15)	40

8. Evaluation Focus on

- Presentation Skills
- Technical Work Quality
- Documentation Quality
- Project Report
- Viva Performance
- Project Demo
- Level of work understanding

9. Roles & Responsibilities

Role	Responsibilities
HoD/Coordinator	Oversee project schedule, arrange defenses, evaluate at all stages
Supervisor	Guide, monitor, and evaluate the project work throughout
Internal Examiner	Evaluate at proposal, midterm, and final defense
External Examiner	Evaluate final presentation, viva, and demo of project
Student	Active in project development, documentation, and defense. Keep supervision logs

10. Report Contents

13.1. Prescribed Content Flow for the Project Proposal

8. Introduction
9. Problem Statement
10. Objectives
11. Methodology
    a. Requirement Identification

- Defense includes viva, presentation, project demonstration, and report evaluation

6. Supervision

- Supervised by a regular faculty member
- One supervisor may guide up to two groups or four individual

7. Evaluation Criteria

Total Marks: 100

Evaluation Phase	Evaluators	Marks
Proposal Defense	HoD/Coordinator (4), Supervisor (12), Internal Examiner (4)	20
Mid-Term	HoD/Coordinator (8), Supervisor (24), Internal Examiner (8)	40
Final Defense	HoD/Coordinator (5), Supervisor (15), Internal Examiner (5), External Examiner (15)	40

8. Evaluation Focus on

- Presentation Skills
- Technical Work Quality
- Documentation Quality
- Project Report
- Viva Performance
- Project Demo
- Level of work understanding

9. Roles & Responsibilities

Role	Responsibilities
HoD/Coordinator	Oversee project schedule, arrange defenses, evaluate at all stages
Supervisor	Guide, monitor, and evaluate the project work throughout
Internal Examiner	Evaluate at proposal, midterm, and final defense
External Examiner	Evaluate final presentation, viva, and demo of project
Student	Active in project development, documentation, and defense. Keep supervision logs

10. Report Contents

13.1. Prescribed Content Flow for the Project Proposal

8. Introduction
9. Problem Statement
10. Objectives
11. Methodology
    a. Requirement Identification

3.2.3. System Modelling
3.2.3.1. Object Modelling: Class and Object Diagram
3.2.3.2. Dynamic Modelling: State and Sequence Diagram
3.2.3.3. Process Modelling: Activity Diagram
3.3. System Design
3.3.1. Refinement of Class, Object, State, Sequence, and Activity Diagrams
3.3.2. Component Diagrams
3.3.3. Deployment Diagrams
3.4. Algorithm Details

Chapter 4: Implementation and Testing

4.3. Implementation
4.3.1. Tools Used (CASE tools, Programming Language, Database Platforms)
4.3.2. Implementation Details of Modules (Description of classes, procedures, functions, methods, algorithms)
4.4. Testing
4.4.1. Test Case for Unit Testing
4.4.2. Test Case for System Testing
4.5. Result Analysis

Chapter 5: Conclusion and Future Recommendations

5.4. Lesson Learnt and Outcome
5.5. Conclusion
5.6. Future Recommendations

Note: *While writing above chapters students should avoid basic definitions. They should relate and contextualize the above mentioned concepts with their project work.*

14. Citation and Referencing

The listing of references should be listed in the references section. The references contain the list of articles, books, url of primary source that are cited in the document. The books, articles, and others that are studied during the literature review and project work but are not cited in the document can be listed in the bibliography section.
The citation and referencing standard should be **IEEE** referencing standard.

15. Report Format Standards

- **Page size:** A4
- **Font:** Times New Roman, 12pt (body), headings bold (16pt/14pt/12pt).
- **Spacing:** 1.5 line, justified.
- **Margins:** Top/Bottom/Right = 1", Left = 1.25".
- **Page Numbers:** Roman (i, ii...) for preliminaries, numeric (1, 2...) from Chapter 1.
- **Headings:** Chapter – 16pt (Bold and center align), Section – 14pt (Bold and left align), Sub-section – 12pt (Bold and left align), Paragraph content – 12pt (Regular)
- **Figures/Tables:** Center-align, Captions – 12pt (Bold)

16. Binding & Submission

- Copies: 3 (Dean Office, College Library, Self)
- Binding: Golden Embracing with Black Book Binding
- Final Copy: Signed final copy should be submitted to the Dean Office and respective department

(Sample of Cover and Title Page)

Lumbini Technological University

Institute of Engineering and Information Technology

TITLE OF PROJECT REPORT

A PRPJECT REPORT

Submitted to

Department of B.Tech in Information Technology

Name of the Collage

In partial fulfillment of the requirement for the B.Tech in Information Technology

Submitted by

Name:

Registration No.:

Symbol No.:

Month, Year

Under the Supervision of

Supervisor Name

Designation

(Sample of Certificate)

Lumbini Technological University

Institute of Engineering and Information Technology

College Name

Supervisor's Recommendation

I, hereby recommend that this project is completed under my supervision by NAME OF THE STUDENT (Year IV Semester: VII entitled **"TITLE OF THE PROJECT"** in partial fulfillment of the requirements for the degree of B.Tech in Information Technology is recommended for the final evaluation.

.....
Signature
Name of Supervisor
Designation of Supervisor
Department Name
Address of the college

(Sample of Approval)

Lumbini Technological University

Institute of Engineering and Information Technology

College Name

LETTER OF APPROVAL

This is to certify that this project is completed by NAME OF THE STUDENT (Year IV Semester: VII) entitled "**TITLE OF THE PROJECT**" in partial fulfillment of the requirements for the degree of B.Tech in Information Technology has been evaluated. In our opinion it is satisfactory in the scope and quality as a project for the required degree.

.....
Signature
Name of Supervisor
Academic Designation
Department Name
Address of the college

.....
Signature
Name of HOD/Program Coordinator
Academic Designation
Department Name
Address of the college

.....
Signature
Name of Internal Examiner
Academic Designation

.....
Signature
Name of External Examiner
Academic Designation

Year IV Semester VIII

IT Entrepreneurship

Course Code	BIT451	Year/Semester	IV/VIII
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	2 hrs/wk
		Practical	N/A
		Total	80 hrs

1. Course Description

This course contains the fundamental concept of entrepreneurship, growth of entrepreneurship, entrepreneurial opportunities, entrepreneurship competencies development, business plan concept and elements of business and business plan evaluation.

2. Course Objectives

- To understand how emerging trends in information technology and innovation affect business processes and potentially create value (Emerging Technologies).
- To utilize the fundamentals of business process innovation and how to manage business process innovation initiatives and process configurations to impact business agility

3. Course Details

3.1 Theory **(48 hrs.)**

Unit 1: Introduction of Business process and Information System **(7 hrs)**

Organization process, Flows in business process, Monitor process performance, Application Infrastructure, Information system and business process, Importance of information system (data and information, Functional information system), Functional organizational structure (Delay in execution the process, excess inventory, lack of visibility across processes)

Unit 2: Enterprise System **(7 hrs)**

Role of enterprise system in organization, Execute the process, Capture and store process data, Stand-alone mainframe systems, Client server architecture, Service Oriented Architecture, Types of Enterprise system, Types of Data in enterprise system (transaction data, master data, organizational data)

Unit 3: IT and Strategy **(7 hrs)**

Information revolution, Business and Strategy, Information Technology Strategy, Strategies and success, Design parameters and Strategic positioning, Evolution and development of strategy, Strategic planning and IT strategies, Evolving a dynamic nature of the Business

Unit 4: Managing IT **(7 hrs)**

IT management and Its Role, IT governance and infrastructure, IT Governance and Strategy, Technology Management Process, Steps in Technology Management, Strategic Aspects of IT and Positioning the company for change, IT and business alignment, Risk management, Implementing and Exploiting IT capabilities, Using IT in a Strategic Manner, Measuring IT with Performance measures and Balanced Score card, Implementing change in IT management

Unit 5: E – Strategy **(5 hrs)**

Introduction, E business and E-strategy, Developing an E-strategy, E-business objectives, E-commerce and E-business, Business –model and E-business model, Making E-strategy and E-economy, Best practice and Competitive advantage

Unit 6: IT strategy for IT companies **(5 hrs)**

Strategic aspects for an IT product companies and IT strategic development, IT strategy and innovation driving factor of start-up product companies, IT strategies for product life cycle and dealing with chasm, Project life cycle ad Strategies at various stages, Technology Selection and IT strategic aspects, Technology change management

Unit 7: Emerging Entrepreneurship in Nepal **(10 hrs)**

Micro, small and medium size enterprise: concept and characteristics, importance and contribution to employment generation and resource mobilization in the economic development of Nepal. Indigenous knowledge and major ethnic entrepreneurship- its problems and prospects; The emerging entrepreneurship in Nepal: Rural entrepreneurship, Tourism entrepreneurship, Agri-entrepreneurship, Transport entrepreneurship.

3.2 Project Work:

- Develop a business Strategy for a small IT company which focuses on building a software product and services and develop a IT strategy for a company
- Develop a risk management Plan
- Preparation of e- business model for an ISP company
- Develop a E strategy for a company for E commerce site
- Case study for knowledge management

4. Evaluation Scheme

Evaluation of students' performance is divided into two main parts: (1) Internal evaluation and (2) Semester examination, with breakdown as follows.

Evaluation	Components	Weightage	Full Marks	Pass Marks
-Internal Evaluation	Class Attendance and Performance	5	40	20
	Assignment	15		
	Seminar/Project/Presentation	20		
	Practical/Lab Examination	-	-	-
	Internal Examination	**20**	**20**	**10**
	Total Internal Marks		**60**	**30**
Semester-End Examination			**40**	**20**

5. Books

- Kulkarni Parag,Chande k.pradip, IT strategy for Business ,Oxford-2008
- Barringer, B.R. and Ireland, R. D. Entrepreneurship: Successfully Launching New Business. Pearson, Delhi, India.

Charantimath, P.M. Entrepreneurship Development and Small Business Enterprises. Pearson, Delhi, India

Data Warehouse and Data Mining

Course Code	BIT452	Year/Semester	IV/VIII
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hrs/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

This course contains the fundamental concept of data mining and data warehousing. This course covers various functionalities, applications, issues and architectures of data mining. Data mining algorithms used to solve various real life problems, are covered in this course. Along with mining, Data warehousing is also important, course content will also focus on models, architectures, technologies used to design data warehouse.

2. Course Objectives

- To understand Data Warehouse fundamentals, Data Mining Principles
- To design data warehouse with dimensional modelling and apply OLAP operations
- To identify appropriate data mining algorithms to solve real world problems
- To impart the knowledge on various data mining concepts and techniques that can be applied to text mining, web mining etc.
- To develop the knowledge for application of data mining and social impacts of data mining.
- To compare and evaluate different data mining techniques

3. Course Outcome

Upon completion of the course the student will be able to

- Interpret the contribution of data warehousing and data mining to the decision-support systems.
- Construct the data needed for data mining using pre-processing techniques.
- Discover interesting patterns from large amounts of data using Association Rule Mining.
- Extract useful information from the labelled data using various classifiers and Compile unlabelled data into clusters applying various clustering algorithms.
- Demonstrate capacity to perform a self-directed piece of practical work that requires the application of data mining techniques.

4. Course Details

4.1. Theory (48 hrs)

Unit 1: Introduction to Data Warehousing (2 hrs)

Definition & Characteristics; Online Transaction Processing (OLTP) vs Online Analytical Processing (OLAP) systems; Subject-orientation, Time-variant, Non-volatility, Integration; Role of data warehousing in Business Intelligence (BI); Historical evolution of Data Warehouse concepts.

Unit 2: Data Warehouse Design (3 hrs)

Dimensional Modelling Basics: Facts and Dimensions; additive, semi-additive, and non-additive facts; Schema Types: Star schema: structure and example, Snowflake schema: normalization in dimensions, Fact constellation schema (galaxy schema); Dimension Table Design: Attributes, hierarchies, surrogate keys; Fact Table Types: Slowly Changing Dimensions (SCD), Data Warehouse Design Process.

Unit 3: ETL (Extract, Transform, Load) Processes **(6 hrs)**

ETL Overview: Role of ETL in DW pipelines, ETL vs ELT; Data Extraction: Types of data sources, Full vs incremental extraction; Data Transformation: Cleaning: handling missing values, smooth noisy data, identify or remove outliers, resolve inconsistencies, detect and remove redundancies; Transformation: normalization, standardization, discretization, data mapping; Reduction: dimensionality reduction, numerosity reduction; Data Loading: Bulk vs trickle load, Initial load vs delta/incremental load, Load validation and reconciliation; Error handling and logging.

Unit 4: Data Warehouse Architecture and Components **(4 hrs)**

Architectural Models: Top-down (Inmon) vs Bottom-up (Kimball) approach, Hybrid architecture; Core Components: Staging area, Data warehouse, Metadata repository, Data marts (dependent/independent); Technical Architecture Layers: Data acquisition, Data storage, Data presentation; Operational Aspects: Refresh cycles, Job scheduling, Backup and recovery; Real-Time Data Warehousing (RTDW): Challenges and techniques

Unit 5: Introduction to Data Mining **(7 hrs)**

Introduction to data mining - Data mining functionalities - Steps in data mining process- Classification of data mining systems - Major issues in data mining; Data Preprocessing: An overview - Data cleaning - Data integration -Data reduction - Data transformation.

Unit 6: Frequent Pattern Mining **(7 hrs)**

Frequent Pattern Mining: Basic Concepts and a Road Map - Efficient and scalable frequent item set mining methods: Apriori algorithm, FP-Growth algorithm - Mining frequent item sets using vertical data format.

Unit 7: Classification Techniques **(7 hrs)**

General approach to classification - Classification by decision tree induction - Bayes classification methods - Model evaluation and selection - Techniques to improve classification accuracy - advanced classification methods: Bayesian belief networks- Lazy learners.

Unit 8: Cluster Analysis **(6 hrs)**

Types of data in cluster analysis - Partitioning methods - K Medoid Clustering - Density based methods - Grid based methods - Outlier analysis.

Unit 9: Data Mining Trends and Research Frontiers **(6 hrs)**

Overview of Web mining-Temporal and Spatial mining-Other methodologies of data mining: Statistical data mining- Data mining applications, Contemporary Issues in data mining.

5. Tutorials

Unit specific problem solving exercise.

6. Practical Work

Experiments

- Designing Star/Snowflake schemas using case studies
- SQL queries on fact and dimension tables
- ETL process using open-source tools (e.g. Apache NiFi)
- OLAP Cube Construction, and OLAP operations using open-source tools (e.g. Apache Kylin)
- Introduction to exploratory data analysis using R.
- Demonstrate the Descriptive Statistics for a sample data like mean, median, variance and correlation etc.,
- Demonstrate Missing value analysis using sample data.
- Demo of Apriori algorithm on various data sets with varying confidence and support.
- Demo of FP Growth algorithm on various data sets with varying confidence and support.

- Demo on Classification Techniques such as Decision Tree (ID3 / CART), Bayesian etc., and using sample data.
- Demonstration of Clustering Techniques K-Medoid and Hierarchical.
- Demonstration on Document Similarity Techniques and measurements.
- Simulation of Page Rank Algorithm.
- Data Aggregation and Materialized Views (e.g., in PostgreSQL)
- Applying Data Mining algorithms
- Generating reports and dashboards (e.g., Apache Superset, Power BI, Tableau etc.)
- Demonstration on Hubs and Authorities.

7. Evaluation Scheme

Evaluation of student's performances is divided into two main parts: (1) Internal Evaluation and (2) Semester examination, with breakdowns as follows:

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal Evaluation	**Class Attendance and Performance**	5	20	10
	Assignment	10		
	Seminar/Project/Presentation	5		
	Practical/Lab Examination	20	20	10
	Internal Examination	**20**	**20**	**10**
	Total Internal Marks		**60**	**30**
Semester-End Examination			**40**	**20**

8. Recommended books

Text Book(s)

1. Jiawei Han and Micheline Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann Publishers, third edition, 2013.

Reference Books

2. Parteek Bhatia, Data Mining and Data Warehousing: Principles and Practical Techniques, Cambridge University Press, 2019.
3. Pang-Ning Tan, Michael Steinbach, Anuj Karpatne, Vipin Kumar, Introduction to Data Mining, Pearson, 2nd Edition, 2019.

Elective I

Year III Semester VI

Image Processing

Course Code	Elective I	Year/Semester	III/VI
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hrs/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

This course covers the investigation, creation and manipulation of digital images by computer. The course consists of theoretical material introducing the mathematics of images and imaging. Topics include representation of two-dimensional data, time and frequency domain representations, filtering and enhancement, the Fourier transform, convolution, interpolation. The student will become familiar with Image Enhancement, Image Restoration, Image Compression, Morphological Image Processing, Image Segmentation, Representation and Description, and Object Recognition.

2. Course Objectives: The objective of this course is to make students able to:

- To develop a theoretical foundation of Digital Image Processing concepts.
- To provide mathematical foundations for digital manipulation of images; image acquisition; preprocessing; segmentation; Fourier domain processing; and compression.
- To gain experience and practical techniques to write programs for digital manipulation of images; image acquisition; pre-processing; segmentation; Fourier domain processing; and compression.

3. Course Details

3.1 Theory **(48 hrs.)**

Unit 1: Introduction **(5 hrs)**

Digital image representation; Fundamental steps in digital image processing; Components of an image processing system; Elements of visual perception: structure of human eye, Image formation in eyes; Image sensing and acquisition: image acquisition using single sensor; Basic concept of sampling and quantization; Relationships between pixels: Neighbor of pixel, adjacency, connectivity, regions and boundaries.

Unit 2: Intensity Transformation **(5 hrs)**

Basic intensity transformation functions: Negative transformation, Log transformation, Gamma transformation; Histogram processing: Histogram equalization, Use of histogram statistics for image enhancement.

Unit 3: Image filtering in Spatial Domain **(6 hrs)**

Spatial filtering: Spatial correlation, convolution and padding, generating spatial filter mask; Smoothing spatial filters: Linear filters, nonlinear filters; Sharpening spatial filters: Unsharp masking and high-boost filtering, Using first order derivative (Gradient), Using second order derivative (Laplacian).

Unit 4: Image Filtering in Frequency Domain **(8 hrs)**

Concept of frequency of an image, Fourier series, Fourier transformation: 1-D and 2-D DFT, Properties of DFT, Frequency domain filters: Smoothing frequency domain filter (Ideal low pass filter, Butterworth low pass filter, Gaussian low pass filter), Sharpening frequency domain filter (Ideal high pass filter, Butterworth high pass filter, Gaussian high pass filter); Fast Fourier Transformation: Derivation of 1-D FFT, Conversion of data from spatial domain to frequency domain using FFT, conversion of data form frequency domain to spatial domain using IFFT; Haar transformation, Hadamard transformation.

Unit 5: Image Restoration and Reconstruction **(6 hrs)**

Introduction to image restoration, Noise Models (Gaussian, Rayleigh, Impulse, Gamma, Uniform, and Exponential), Restoration filters: Mean Filters (Arithmetic, Geometric, and Harmonic), Order Statistics

Filters (Median, Max and Min, Midpoint, and Alpha trimmed mean), Band pass and Band Reject Filters (Ideal, Gaussian, and Butterworth).

Unit 6: Image Compression **(5 hrs)**

Introduction to image compression, Compression ratio, Relative data redundancy, Average code length, Redundancies in Image: Coding redundancy (Huffman code), Inter-pixel redundancy (Run length coding) and Psychovisual redundancy; Image compression model.

Unit 7: Morphological Image Processing **(3 hrs)**

Introduction to mathematical morphology, Basic concept of set theory, Logic operation in binary image, Hit, Fit and Miss, Erosion and Dilation, Opening and Closing.

Unit 8: Image Segmentation **(6 hrs)**

Introduction to image segmentation; Similarity based approach (Thresholding: Global, Local and Adaptive); Discontinuity based approach (Point, Line and Edge Detection); Edge linking and boundary detection; Hough transform; Region based segmentation: Region growing algorithm, Region splitting and merging algorithm.

Unit 9: Representation, Description and Object Recognition **(4 hrs)**

Representation and Description: Chain codes, Shape numbers, and Signature; Object Recognition: Pattern and Pattern class, Recognition based on decision-theoretic methods (Matching, Neural Networks).

3.2 List of Tutorials

Unit specific exercise focused on problem solving.

3.3 Laboratory Works

Students are required to develop programs in related topics using suitable programming languages such as MatLab or Python or other similar programming languages.

4. Evaluation Scheme

Evaluation of student's performances is divided into two main parts: (1) Internal Evaluation and (2) Semester examination, with breakdowns as follows:

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal Evaluation	Class Attendance and Performance	5	20	10
	Assignment	10		
	Seminar/Project/Presentation	5		
	Practical/Lab Examination	20	20	10
	Internal Examination	**20**	**20**	**10**
	Total Internal Marks		**60**	**30**
Semester Examination			**40**	**20**

5.1 Text Books

- Rafael C. Gonzalez and Richard E. Woods, "Digital Image Processing", Pearson Edition, Latest Edition.

5.2 Reference Books

- I. Pitas, "Digital Image Processing Algorithms", Prentice Hall, Latest Edition.
- A. K. Jain, "Fundamental of Digital Image processing", Prentice Hall of India Pvt. Ltd., Latest Edition.
- K. Castlemann, "Digital image processing", Prentice Hall of India Pvt. Ltd., Latest Edition.
- P. Monique and M. Dekker, "Fundamentals of Pattern recognition", Latest Edition.

Integrated GIS and Remote Sensing

Course Code	Elective I	Year/Semester	III/VI
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hr/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

This integrated course covers fundamental principles, technologies, and applications of Geographic Information Systems (GIS) and Remote Sensing (RS). Students will learn spatial data acquisition, analysis, and visualization techniques, with hands-on experience in solving real-world environmental, urban, and resource management challenges.

2. Course Objectives

By the end of this course, students should be able to:

- Understand core concepts of GIS and RS, including data models, platforms, and sensors.
- Process and analyze geospatial data using appropriate techniques.
- Integrate GIS and RS for applications like land-use change detection and disaster management.
- Design effective cartographic outputs and spatial decision-support systems.

3. Course Details

(48 hrs.)

3.1 Theory

Unit 1: Foundations of Geospatial Technologies **(6 hrs)**

GIS Concepts: Definitions, history, components (hardware, software, data, people); RS Fundamentals: Electromagnetic spectrum, energy-matter interaction, resolutions; Integrated Applications: Smart cities, precision agriculture, climate change studies.

Unit 2: Data Models and Acquisition **(6 hrs)**

Types of data, Geographical data, major sources of GIS data; Spatial and non-spatial data, Data linkage; GIS Data Models: Vector (spaghetti, topological), raster, TIN; RS Data Sources: Satellites (Landsat, Sentinel), UAVs, LiDAR; Data Quality: Accuracy, precision, metadata (ISO 19115).

Unit 3: Coordinate Systems and Georeferencing **(7 hrs)**

Projections: Mapping concepts, coordinate systems, UTM, Lambert Conformal Conic; datum transformations; Georeferencing: Ground control points, Rectification; GPS Integration: GPS basics, components of GPS, GPS segments, Differential GPS, RTK surveys.

Unit 4: Spatial Databases **(7 hrs)**

Data concept and design, database design, data processing system, Database Table; Database Design: Hierarchical database, network database, relational database (PostgreSQL/PostGIS), NoSQL for geospatial; Normalization: 1NF to 3NF for spatial data and data security; Spatial SQL: Queries, indexing.

Unit 5: Remote Sensing Processing **(8 hrs)**

Image Preprocessing: Radiometric/atmospheric correction; Classification: Supervised (SVM), unsupervised (K-means); Change Detection: Time-series analysis (NDVI, NDWI, NDBI etc, trends).

Unit 6: Advanced Geospatial Analysis **(8 hrs)**

GIS Analysis: Network routing, hydrological modelling; RS Analysis: Spectral indices (NDVI, NDWI), object-based image analysis (OBIA); Integration: RS-to-GIS data conversion (e.g., raster to vector).

Unit 7: Map output and Information Dissemination **(6 hrs)**

Cartography and GIS; Map Layout Elements; Map Design: Color theory, visual hierarchy; Types of Maps; Cartographic output Format.

3.2 List of Tutorials:

The various tutorial activities that suits this course should cover all the content of this course to give student a space to engage more actively with the course content in the presence of instructor. Students should submit tutorials as assignments or class-works to the instructor for evaluation.

A. Discussion-based Tutorials:

1. GIS vs. Remote Sensing: Compare/contrast applications in environmental monitoring.
2. Vector vs. Raster Data Models: Debate advantages for urban planning vs. forestry.
3. Open-Source vs. Proprietary Geospatial Tools: Ethical and technical implications.
4. AI in Geospatial Analysis: Discuss ML applications in satellite image classification.

B. Problem-solving-based Tutorials:

5. Coordinate Transformations: Convert WGS84 to UTM coordinates for given points.
6. SQL for Spatial Queries: Solve queries (e.g., "Find hospitals within 1 km of flood zones").
7. Spectral Indices Calculation: Compute vegetation indices from sample spectral bands.
8. Topology Error Fixing: Correct gaps/overlaps in polygon datasets.
9. Network Analysis: Optimize emergency routes using Dijkstra's algorithm.
10. Image Classification: Classify a Landsat scene using maximum likelihood.
11. Cartographic Design: Critique map layouts based on visual hierarchy principles.

3.3 Laboratory Works

- Fundamentals of Geospatial Data Handling
- Coordinate Systems and Georeferencing
- Spatial Database Creation
- Remote Sensing Image Processing
- Vector-Based Spatial Analysis
- Raster-Based Terrain Analysis
- Integrated GIS-RS Application
- Cartographic Design and Visualization

4. Evaluation Scheme

Evaluation of students' performance is divided into two main parts: (1) Internal evaluation and (2) Semester examination, with breakdown as follows.

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal Evaluation	Class Attendance and Performance	5	20	10
	Assignment	10		
	Seminar/Project/Presentation	5		
	Practical/Lab Examination	20	20	10
	Internal Examination	**20**	**20**	**10**
	Total Internal Marks		**60**	**30**
Semester-End Examination			**40**	**20**

5. Books

1. Chang, K. (2019). Introduction to Geographic Information Systems (9th edition). New York: McGraw-Hill Education, Penn Plaza.
2. Bolstad, P. (2016). GIS Fundamentals: A First Text on Geographic Information Systems (5th edition). United States: Eider Press.
3. Jensen, J. R., Jensen, R. R. (2013). Introductory Geographic Information Systems. Boston: Pearson.
4. Lillesand, T., Kiefer, R. W., & Chipman, J. (2015). Remote Sensing and Image Interpretation. Wiley.
5. Jensen, J. R. (2015). Introduction to Remote Sensing. CRC Press.
6. Campbell, J. B., & Wynne, R. H. (2011). Introduction to Remote Sensing. Guilford Press.
7. Good child, Loiigley, Maguire, "Geographic Information System and Science"
8. A complete coiu'se manual in Geogi'aphic Information System, nec
9. ICIMOD "Training manual on GIS application and Remote sensing"

Elective II

Year IV Semester VIII

Data Visualization and Interpretation

Course Code	Elective II	Year/Semester	IV/VIII
Credit Weightage	3	Lecture	2 hrs/wk
		Tutorial	1 hrs/wk
		Practical	2 hrs/wk
		Total	80 hrs

1. Course Description

This course introduces the principles and techniques of data visualization and interpretation within the context of data science. It aims to equip students with the knowledge and skills necessary to transform data into meaningful visual narratives that facilitate analytical reasoning and decision-making. Through hands-on projects and case studies, students will learn to apply visualization tools and software to real-world data sets, gaining insights into the role of visualization in data science.

2. Course Objectives

Upon completion of this course, students will be able to:

- Select appropriate visualization techniques for various types of data and analysis goals with understanding of principles and aesthetics of efficient visualization
- Use leading data visualization tools and software to create impactful visual representations of data for different use cases
- Communicate complex data insights through compelling visual narratives.
- Apply ethical considerations in the visualization and interpretation of data.

3. Learning Outcomes (Follow Bloom's Taxonomy to the extent possible. See Annex A)

Upon completion of this course, students will be able to:

- Explain the importance of data visualization and Distinguish between exploratory and explanatory analysis.
- Identify and use different elements of data visualization that are used in procedural and declarative data visualization
- use tools such as matplotlib, vega-altair or ggplot, to plot different types of data plot as need
- apply Gestalt principles of visual perception for designing different visualization without producing chart junk
- Conduct univariate, bivariate and multivariate analysis using different types of plots
- Create exploratory visualization and appropriate narrative for storytelling with data
- develop interactive visualization in dashboards using certain dashboarding tools such as looker studio, Power BI, Streamlit, or other tools
- analyze and recognize bias in data and visualization and use appropriate guidelines for ethical data visualization designs.

4. Course Details

(32 hrs)

Unit 1: Introduction to Data Visualization **(2hrs)**

Importance of Data Visualization; Multiple interpretations of Data and Visualization; Exploratory vs Explanatory Analysis.

Unit 2: Visual Encoding and Graphical Foundations **(6 hrs)**

Data types; Elements of Visual Graphics; Pipeline of Graphic Generation; Procedural Graphic System: Matplotlib; Declarative system based on Grammar of Graphics: ggplot, Vega-Altair.

Unit 3: Visual Aesthetics Theory **(8 hrs)**

Design thinking in data visualization; Layout, colour, and typography; Gestalt Principles of Visual Perception; Data ink-ratio & Chart Junk.

Unit 4: Exploratory Data Analysis **(8 hrs)**

Univariate Analysis: Bar, Pie, Dot & Jitter, Histogram, KDE, Rank Order Plot, Pareto chart, CDF plot, Quantile Plot, Box-whisker Plots, Violin Plot; Bivariate Analysis: CrossTab, Scatter plot, Trend Lines, Correlations, Heatmap, Bivariate KDE, Violin; Multivariate Analysis: Faceting, Joint Plots, Plot Matrix, Non-positional encoding; Dimensionality reduction for data visualization (PCA, t-SNE).

Unit 5: Storytelling with Data **(4 hrs)**

Explanatory Analysis; Creating a Narrative for storytelling with data; Interactive Visualization; Dashboard Design Best Practices.

Unit 6: Responsible and Trustworthy Data Visualization **(4 hrs)**

Ethical considerations in Data Visualization (eg of Lying with Data); Types and source of Bias in Data and Visualization; Recognizing and avoiding bias; Case studies: Misleading visualizations and their impact; Guidelines for ethical visualization design.

5. Tutorial

Unit specific problem solving exercise.

6. Practical

1. Procedural Graphics Tool : Matlab and Seaborn
2. Declarative Graphics Tool: ggplot or Vega-Altair
3. Exploratory Data Analysis:
    a. Univariate Analysis
    b. Bivariate Analysis
    c. Multivariate Analysis
4. Interactive Dashboarding with Apache Superset or Google Looker Studio or MS Power BI or streamlit or any other dashboarding tool
5. Advance Data Visualization Techniques
    a. Time series visualization with any visualization tool
    b. Geospatial Visualization with Geoplot and geopandas
    c. Network visualization with networkxx

7. Evaluation Scheme

Evaluation of student's performances is divided into two main parts: (1) Internal Evaluation and (2) Semester examination, with breakdowns as follows:

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal Evaluation	Class Attendance and Performance	5	20	10
	Assignment	10		
	Seminar/Project/Presentation	5		
	Practical/Lab Examination	20	20	10
	Internal Examination	**20**	**20**	**10**
	Total Internal Marks		**60**	**30**
Semester-End Examination			**40**	**20**

8. Recommended Books

1. "The Visual Display of Quantitative Information" by Edward R. Tufte, Graphic Press 2nd Edition, 2007
2. "Storytelling with Data: A Data Visualization Guide for Business Professionals" by Cole Nussbaumer Knaflic, Wiley, 2015
3. "Data Points: Visualization That Means Something" by Nathan Yau
4. "Interactive Data Visualization for the Web" by Scott Murray (for those interested in web technologies)
5. "The Grammar of Graphics" by Leland Wilkinson, 1999

Application of GIS and Remote Sensing using Python

Course Code	Elective II	Year/Semester	IV/VIII
Credit Weightage	3	Lecture	2 hrs/wk
		Tutorial	N/A
		Practical	3 hrs/wk
		Total	80 hrs

1. Course Description

This course provides a comprehensive introduction to Python programming and its application in Geographic Information Systems (GIS). Students will learn fundamental programming concepts and how to use Python to process, analyze, and visualize spatial data. The course focuses on real-world geospatial applications including terrain modeling, spatial queries, coordinate transformations, and watershed analysis using libraries such as GeoPandas, Rasterio, Shapely, and GDAL.

2. Course Objectives

By the end of this course, students should be able to:

- Write and debug Python programs for data analysis and spatial processing.
- Apply spatial data structures and functions to manipulate and analyze geospatial data.
- Implement geoprocessing operations and raster analysis using Python libraries.
- Develop spatial workflows for advanced GIS applications such as watershed modeling, terrain analysis, and spatial statistics.
- Visualize spatial data using libraries such as Matplotlib, Seaborn, and Plotly

3. Learning Outcomes (Follow Bloom's Taxonomy to the extent possible. See Annex A)

Upon completing this course, Students will be able to:

- Remember Python syntax and core data structures (variables, lists, dictionaries, etc.).
- Understand spatial concepts such as projections, coordinate transformations, and geoprocessing.
- Apply spatial and raster analysis techniques using Python.
- Analyze terrain models, DEM data, and land-use patterns with statistical methods.
- Evaluate and optimize spatial workflows using coding best practices.
- Create GIS-integrated projects to solve real-world spatial problems (e.g., watershed delineation, urban heat island detection).

4. Course Details

4.1. Theory (32 hrs)

Unit 1: Introduction to Python Programming (4 hrs)

Variables, operators, expressions, input/output; Interactive vs. script mode, debugging; Data types: int, float, str, list, dict, conversions; String operations and basic file handling.

Unit 2: Functions, Control Flow, and Data Structures (4 hrs)

Defining functions, parameters, return types, recursion; Conditional statements (if, else), loops, list comprehensions; Exception handling; Core data structures: strings, lists, dictionaries, sets.

Unit 3: Advanced Python Programming (5 hrs)

File operations: open, read, write, file modes; Context managers (with statement); OOP: Classes, methods, inheritance, encapsulation, polymorphism.

Unit 4: Coordinate Transformations and Image Handling **(6 hrs)**
Affine and perspective transformations using matrices; Interpolation techniques; Raster data: reading bands, metadata, histograms; Libraries: NumPy, Rasterio, GDAL.

Unit 5: Spatial Data Processing and Geoprocessing **(6 hrs)**
Vector data models: points, lines, polygons; CRS, projections, spatial joins, and geometric operations; Geoprocessing tasks: buffer, clip, merge, dissolve; Libraries: GeoPandas, Shapely.

Unit 6: Spatial Analysis and Applications **(7 hrs)**
Raster processing, terrain analysis, slope/aspect/hillshade; Spatial interpolation techniques (IDW, Kriging - basic); Watershed modeling: DEM preprocessing, flow direction; Spatial statistics and network analysis (e.g., shortest path using OSMnx).

5. Tutorial
N/A

6. Practical

Lab #	Question Description
1	Basic Python scripting: variables, loops, functions
2	Vector data processing with GeoPandas (e.g., land-use shapefiles)
3	Raster analysis: DEM visualization and histogram analysis
4	Coordinate transformations and projection handling
5	Geoprocessing operations: buffer, clip, union, merge
6	Raster terrain analysis (slope, hillshade, aspect)
7	Watershed delineation using DEM and flow direction modeling
8	Spatial visualization using Matplotlib, Seaborn, Plotl

For each lab work, students are supposed to implement the algorithm with performance analysis.

7. Evaluation Scheme

Evaluation of students' performance is divided into two main parts: (1) Internal evaluation and (2) Semester examination, with breakdown as follows.

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal Evaluation	Class Attendance and Performance	5	20	10
	Assignment	10		
	Seminar/Project/Presentation	5		
	Practical/Lab Examination	20	20	10
	Internal Examination	**20**	**20**	**10**
	Total Internal Marks		**60**	**30**
Semester-End Examination			**40**	**20**

8. Recommended Books

1. Bolstad, P. (2016). GIS Fundamentals: A First Text on Geographic Information Systems.
2. Campbell, J.B., & Wynne, R.H. (2011). Introduction to Remote Sensing.
3. DeMers, M.N. (2018). Fundamentals of Geographic Information Systems.
4. Jensen, J.R. (2015). Introductory Remote Sensing: Principles and Concepts.
5. Lillesand, T., Kiefer, R.W., & Chipman, J. (2015). Remote Sensing and Image Interpretation.
6. Mather, P.M. (2016). Computer Processing of Remotely-Sensed Images: An Introduction.
7. Richards, J.A., & Jia, X. (2013). Remote Sensing Digital Image Analysis: An Introduction.
8. Zhang, Y., & Xu, B. (2022). Big Data and Machine Learning in Geospatial Analysis.

Social Media Mining

Course Code	Elective II	Year/Semester	IV/VIII
Credit Weightage	3	Lecture	3 hrs/wk
		Tutorial	1 hrs/wk
		Practical	2 hrs/wk
		Total	96 hrs

1. Course Description

This course introduces students to the growing field of web and social media data analytics. With the increasing prevalence of online platforms, valuable data is constantly generated by users in structured and unstructured formats. Students will explore techniques to acquire, preprocess, and analyze such data using statistical and machine learning approaches. The course emphasizes the use of APIs to collect data from platforms like Twitter, YouTube, and Reddit, and applies methods such as regression, classification, and sentiment analysis to uncover insights. Ethical considerations, including privacy and responsible use of data, are embedded throughout the course to encourage socially responsible data science practice. The course culminates in a project involving real-world social media datasets, encouraging innovation and research.

2. Course Objectives

By the end of this course, students will be able to:

- Understand the nature, characteristics, and limitations of web and social media data sources.
- Acquire structured and unstructured data through APIs and web scraping techniques.
- Preprocess and transform web data for statistical and machine learning analysis.
- Apply core analytical techniques, including correlation, regression, classification, and sentiment analysis.
- Evaluate ethical and legal implications of using publicly available data.
- Explore and analyze real-time streaming and web data using appropriate tools and libraries.

3. Course Outcomes

After completing this course, students will be able to:

- Implement and use social media APIs (e.g., Twitter, YouTube, Reddit) to programmatically collect data.
- Apply data mining techniques to derive patterns and trends from social media data.
- Conduct sentiment analysis on user-generated content using natural language processing (NLP) tools.
- Develop basic real-time web data processing pipelines.
- Demonstrate understanding of ethical issues such as user consent, bias, misinformation, and data privacy.
- Utilize modern tools (such as Python, pandas, scikit-learn, Tweepy, TextBlob, and Streamlit) to design and present data-driven insights.s.

4. Course Details

4.1 Theory **(48 hrs)**

Unit-1: Fundamentals of Social Media **(10 hrs)**

Key concepts of social media mining; Good data versus bad data; Understanding sentiments; Scherer's typology of emotions; Sentiment polarity, data and classification; Supervised social media mining; Lexicon-based sentiment; Supervised social media mining; Naive Bayes classifiers; Unsupervised social

media mining; Item Response Theory for text scaling; Social Computing Tasks; Importance of Nodes; Strengths of Ties; Influence Modelling.

Unit-2: Social Media Data **(5 hrs)**

The nature of social media data; Traditional versus non-traditional social data; Measurement and inferential challenges; Opinion mining made difficult; Sentiment and its measurement; Big Data; Human Sensor and Honest Signals; Quantitative approaches and its Challenges.

Unit-3: Communities and Social Media **(5 hrs)**

Types of Communities; Node-Centric Community Detection; Group-Centric Community Detection; Network-Centric Community Detection; Hierarchy Centric Community Detection.

Uint-4: Patterns and Classification in Social Media **(10 hrs)**

Pattern Evolution; A Naïve Approach to Studying Community Evolution; Community Evolution in Smoothly Evolving Networks; Segment-based Clustering with Evolving Networks; Classification with Network Data; Collective Classification; Community-based Learning.

Unit-5: R Fundamentals **(10 hrs)**

Introduction to R; Assignment and arithmetic basics, Functions, Arguments, Vectors, Sequences, and Combining vectors; creating data frames and importing files; Visualization in R; Style and workflow.

Unit-6: Case Study: Mining Twitter Data **(8 hrs)**

Twitter API; Search and Extraction of Tweets, Retweets; Graph Visualization of Tweets, Retweets; Tag Clouds; Harvesting Friends and Followers; Analysis of Relationships, Cliques; Geodata.

5. List of Tutorials

Unit specific problem solving exercise.

6. Practical

- Scraping data from social media using appropriate tools
- Social media data visualization using R

7. Evaluation Scheme

Evaluation of student's performances is divided into two main parts: (1) Internal Evaluation and (2) Semester examination, with breakdowns as follows:

Evaluation	Components	Weightage	Full Marks	Pass Marks
Internal Evaluation	Class Attendance and Performance	5	20	10
	Assignment	10		
	Seminar/Project/Presentation	5		
	Practical/Lab Examination	20	20	10
	Internal Examination	**20**	**20**	**10**
	Total Internal Marks		**60**	**30**
Semester-End Examination			**40**	**20**

8. Recommended Books

Text Books:

1. John Sammons, The Basics of Digital Forensics, Elsevier
2. John Vacca, Computer Forensics: Computer Crime Scene Investigation, Laxmi Publications.

Reference Books:

1. Richard Heimann, Nathan Danneman, "Social Media Mining with R", Packt Publishing, March 2014.
2. Mining the Social Web, 2nd Edition Data Mining Face book, Twitter, LinkedIn, Google+, GitHub, and More By Matthew A. Russell Publisher: O'Reilly Media.

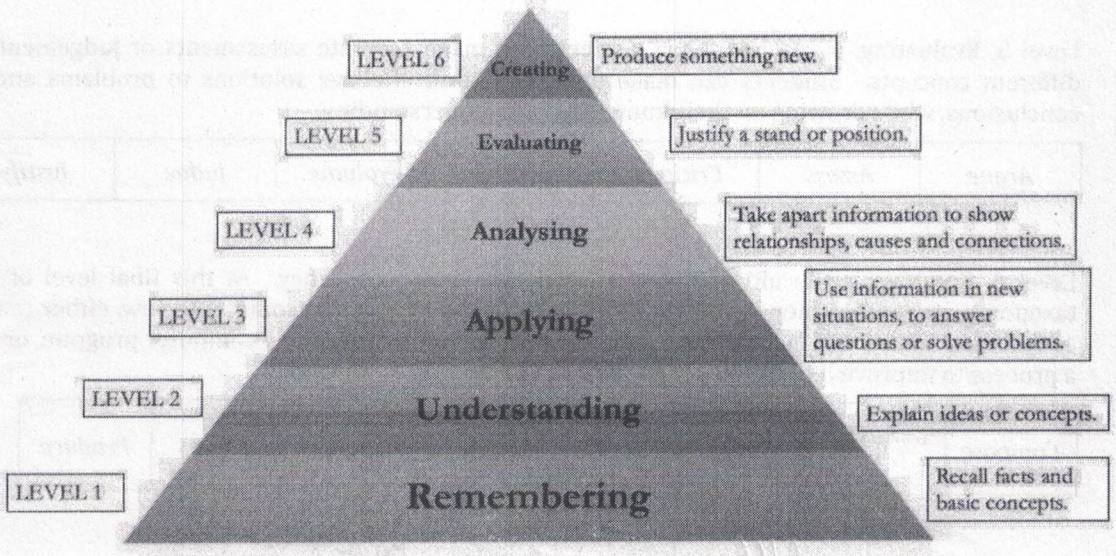
Web References:

1. Lei Tang, Huan Liu "Community Detection and Mining in Social Media", Morgan and Claypool Publishers,1st Edition, 2010.
2. Matthew A. Russell, "21 Recipes for Mining Twitter", O'Reilly Media, January 2011.
3. Yangchang Zhao, "R and Data Mining, Examples and Case Studies", Academic Press; 1st Edition , 2012

Annex A

Bloom's Taxonomy action verbs:

As Bloom's taxonomy is a hierarchy of progressive processes ranging from the simple to the complex, in which it is necessary to first master those lower down the pyramid before being able to master those higher up, the framework promotes what Bloom termed 'mastery learning'. In other words, by moving up the taxonomy, students become more knowledgeable, more skilled and develop an improved understanding of the content they are learning. Thus, by creating lesson plans and tasks, using the examples of verbs (in italics) provided, teachers can align with the different levels of the taxonomy.



Level 1, Remembering, is the most basic, requiring the least amount of cognitive rigour. This is about students recalling key information, for example, the meaning of a word.

Arrange	*Define*	*Describe*	*List*	*Match*	*Name*	*Order*	*Recall*	*Reproduce*

Level 2, Understanding, is to do with students demonstrating an understanding of the facts remembered. At this level, the student who recalls the definition of a word, for example, would also be able to show understanding of the word by using it in the context of different sentences.

Classify	*Discuss*	*Explain*	*Identify*	*Report*	*Summarise*

Level 3, Applying, is concerned with how students can take their knowledge and understanding, applying it to different situations. This usually involves students answering questions or solving problems.

Apply	*Calculate*	*Demonstrate*	*Interpret*	*Show*	*Solve*	*Suggest*

Level 4, Analysing, is about students being able to draw connections between ideas, thinking critically, to break down information into the sum of its parts.

Analyse	*Appraise*	*Compare*	*Contrast*	*Distinguish*	*Explore*	*Infer*	*Investigate*

Level 5, Evaluating, is reached when students can make accurate assessments or judgements about different concepts. Students can make inferences, find effective solutions to problems and justify conclusions, while drawing on their knowledge and understanding.

Argue	*Assess*	*Critique*	*Defend*	*Evaluate*	*Judge*	*Justify*

Level 6, Creating, is the ultimate aim of students' learning journey. At this final level of Bloom's taxonomy, students demonstrate what they have learnt by creating something new, either tangible or conceptual. This might include, for example, writing a report, creating a computer program, or revising a process to improve its results.

Compose	*Construct*	*Create*	*Devise*	*Generate*	*Organise*	*Plan*	*Produce*