Dr. Babasaheb Ambedkar Technological University

(Established as a University of Technology in the State of Maharashtra Under Maharashtra Government Act No.: XXXIX of 2014)

P. O. Lonere, Dist. Raigad, Pin 402 103, Maharashtra

Telephone and Fax: 02140-275142

www.dbatu.ac.in

Course structure and detailed syllabus

Third Year B. Tech. Programme in Information Technology

With effect from Academic Year 2019-20

Teaching and Evaluation Scheme Third Year B. Tech. Programme in Information Technology

Sr. No	Course Code	Course Title	Teaching Scheme			Evaluation Scheme					Total Marks	Credits	Total Hours
			L	T	P	MSE	CA		ESE				
							CA-I	CA-II	Internal	External			
		Semester V											
1	BTITC501	Database Management Systems	3	-	-	20	20		60		100	3	3
2	BTITC502	Design and Analysis of Algorithms	3	-	-	20	20		60		100	3	3
3	BTITC503	Software Engineering	3	-	-	20	20		60		100	3	3
4	BTITC504	Probability and Queuing Theory	3	-	-	20	20		60		100	3	3
5		**Elective III (Open)**	3	-	-	20	20		60		100	3	3
	BTITOE505A	Graph Theory											
	BTITOE505B	Human Computer Interaction											
	BTITOE505C	Game Theory											
6		**Elective IV**	3	-	-	20	20		60		100	3	3
	BTITPE506A	Embedded Systems											
	BTITPE506B	IT Service Management											
	BTITPE506C	Information Storage Management											
	BTITPE506D	Network Management											
	BTITPE506E	Data Visualisation											
7	BTHM501	Constitution of India	2	-	-	-	50	50	-	-	100	Audit	2
8	BTITL507	Programming Lab (R Programming)	-	-	2	-	15	15	10	10	50	1	2
9	BTITL508	Database Management Systems Lab	-	-	2	-	15	15	10	10	50	1	2
10	BTITL509	Design and Analysis of Algorithms Lab	-	-	2	-	15	15	10	10	50	1	2
11	BTITS510	Seminar	-	-	2	-	30		10	10	50	1	2
12	BTITF511	Field Training/Internship/ Industrial Training II Evaluation	-	-	-	-	-		50		50	1	-
Summary of Semester Assessment Marks, Credits & Hours			**20**	**-**	**8**	**120**	**340**		**490**		**950**	**23**	**28**
		Semester VI											
1	BTITC601	Operating Systems	3	-	-	20	20		60		100	3	3
2	BTITC602	Compiler Construction	3	-	-	20	20		60		100	3	3
3	BTITC603	Object Oriented Software and Web Engineering	3	-	-	20	20		60		100	3	3
4	BTITC604	Digital Image Processing	3	-	-	20	20		60		100	3	3
5		**Elective V (Open)**	3	-	-	20	20		60		100	3	3
	BTITOE605A	Enterprise Resource Planning											
	BTITOE605B	Decision Support Systems											
	BTITOE605C	Software Project Management											
6		**Elective VI**	3	-	-	20	20		60		100	3	3
	BTITPE606A	Software Testing											
	BTITPE606B	Data Storage Technologies & Networks											
	BTITPE606C	Service Oriented Architecture											
	BTITPE606D	Network Programming											
	BTITPE606E	Advanced Database Technology											
7	BTITL607	Operating Systems Lab	-	-	2	-	15	15	10	10	50	1	2
8	BTITL608	Digital Image Processing Lab	-	-	2	-	15	15	10	10	50	1	2

9	**Elective VI Lab**		-	-	2	-	15	15	10	10	50	1	2
	BTITPEL609A	Software Testing Lab											
	BTITPEL609B	Data Storage Technologies & Networks Lab											
	BTITPEL609C	Service Oriented Architecture Lab											
	BTITPEL609D	Network Programming Lab											
	BTITPEL609E	Advanced Database Technology Lab											
10	BTITP610	Mini Project	-	-	2	-	30		10	10	50	1	2
11	BTITF611	Field Training/Internship/ Industrial Training III (Minimum **four** weeks which can be completed partially in fifth semester and sixth semester or at one time)	-	-	-	-	-		-		-	*	-
Summary of Semester Assessment Marks, Credit s & Hours			**18**	**-**	**08**	**120**	**240**		**440**		**800**	**22**	**26**
*Evaluation in VII semester													

Course Title:	**Database Management Systems**	**Semester V**	
Course Code	**BTITC501**	**Course Type**	**Mandatory**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream	**Core**	**Credits**	**3**

Course Objectives:

1. To understand architecture and functioning of database management systems.
2. To learn relational mode.
3. To use structured query language (SQL) and its syntax, transactions, database recovery and techniques for query optimization.
4. To acquaint with various normalization forms and query processing.
5. To learn indexing methods.

Course Outcomes:

1. After learning the course the students should be able:
2. To explain need of database management.
3. To design and implement a database schema for a given problem-domain.
4. To normalize a database.
5. To create and query a database using SQL DML/DDL commands, stored procedures and functions.
6. To declare and enforce integrity constraints on a database.
7. To illustrate understanding of indexing methods.

Course Content:

UNIT I

Introduction: Basic concepts, Advantages of DBMS over file-processing systems, Data abstraction, Data models and data independence, Components of DBMS and overall structure of DBMS, Data modeling, Entity, Attributes, Relationships, Constraints, Keys E-R diagrams, Components of E-R Model.

UNIT II

Relational Model: Basic concepts, Attributes and domains, Concept of integrity and referential constraints, Schema diagram. Relational query languages, Relational Algebra and Relational Calculus: Tuple relational and domain relational calculus.

UNIT III

Structured Query Language-I: Introduction, Characteristics and advantages, Data types and literals, DDL, Tables: creating, modifying, deleting, Views: creating, dropping, Updation using views, DML, Operators, SQL DML queries, SELECT query and clauses.

UNIT IV

Structured Query Language- II: Set operations, Predicates and joins, Set membership, Tuple variables, Set comparison, Ordering of tuples, Aggregate functions, Nested queries, Database modification using SQL Insert, Update and Delete queries, Dynamic and SQL and concept of stored procedures, Query-by-example.

UNIT V

Relational Database Design: Notion of normalized relations, Functional dependency, Decomposition and properties of decomposition, Normalization using functional dependency, Multi-valued dependency and join dependency. Storage and File Systems: Secondary storage, RAID, File organization, Indices, Static and dynamic hashing, B-Trees and B+ Trees.

UNIT VI

Query Processing and Transaction Management: Measures of query cost, Selection operation, Sorting and join operation, Transaction concept, Components of transaction management, Concurrency and recovery sys-tem, Different concurrency control protocols such as timestamps and locking, Validation, Multiple granularity, Deadlock handling, Different crash recovery methods such as log-based recovery, Shadow-paging, Buffer management and Remote backup system.

Text Books

1. Abraham Silberschatz, Henry F. Korth, and S. Sudarshan, ***"Database System Concepts"***, McGraw Hill Education, 6th Edition, 2011.

2. Ramez Elmasri and Shamkant B. Navathe, ***"Fundamental Database Systems"***, Pearson Education, 7th Edition, 2015.

3. Raghu Ramkrishnan, Johannes Gehrke, ***"Database Management Systems"***, McGraw Hill Education, 3rd Edition, 2007.

Reference Books:

1. Carlos Coronel, Steven Morris ***"Database systems: Design Implementation and Management"***, Cengage Learning Press, 11th Edition, 2014.

2. J. Murach, ***"Murach's MySQL"***, Shroff Publication, 2nd Edition, 2016.

3. J. Murach, ***"Murach's Oracle SQL and PL/SQL: Works with All Versions Through 11g"***, Shroff Publication, 2008.

Course Title:	**Design and Analysis of Algorithms**	**Semester V**	
Course Code	**BTITC502**	**Course Type**	**Mandatory**
Pre-requisite	**Data Structures**	**L – T – P**	**3 – 0– 0**
Stream	**Core**	**Credits**	**3**

Course Objectives:

1. To learn fundamentals of algorithms design techniques.
2. To understand basic knowledge of computational complexity, approximation and randomized algorithms, selection of the best algorithm to solve a problem.
3. To analyze the performance of algorithms, to compare algorithms with respect to time and space complexity.
4. To develop proficiency in problem solving and programming.

Course Outcomes:

After learning the course the students should be able:

1. Develop efficient algorithms for simple computational tasks.
2. Gain understanding of concepts of time and space complexity, worst case, average case and best case complexities and the big-O notation.
3. Design standard algorithms such as sorting, searching, and problems involving graphs.
4. Compute complexity measures of algorithms, including recursive algorithms using recurrence relations.

Course Content:

UNIT I

Introduction: Instruction counts, Growth functions, Necessity of time and space analysis of algorithms, Order notations (O, Θ, Ω notations), Problem instance size, frequently occurring recurrence relations in analysis of algorithms.

UNIT II

Design Techniques-I: Divide and Conquer: Binary search, finding maximum and minimum, Merge sort, Quick sort, Strassen's matrix multiplication. Greedy Algorithms: Knapsack problem, Job sequencing with deadlines, optimal storage on tapes, Optimal merge pattern, Single source shortest paths.

UNIT III

Design Techniques-II: Dynamic Programming: Multistage graphs, All pairs shortest paths, 0/1 Knapsack, Travelling salesman problem.

UNIT IV

Design Techniques: Backtracking: 8-Queens Problems, Sum of subsets, Graph coloring. Branch-and-bound: Least cost (LC) search, Control abstractions for LC search, FIFO branch and bound, LC branch and bound.

UNIT V

Selected Algorithms from Various Areas: Graph Theory, Elementary Algorithms: DFS, BFS, Topological Sort, Minimum spanning trees (Kruskal and Prim's algorithms), Shortest Paths: Single source shortest paths, all pairs shortest paths, String Matching: The naive string-matching algorithm, The Robin-Karp algorithm, The Knuth-Morris-Pratt algorithm.

UNIT VI

Complexity Theory: Lower-bound arguments, NP-completeness: Introduction to NP-Complete, Reducibility (SAT, Independent Set, 3VC, Subset Sum and Partition, Hamiltonian Circuit).

Text Books:

1. Thomas Cormen, Charles Leiserson, Ronald Rivest and Cliford Stein, ***"Introduction to Algorithms"***, MIT Press, 3rd Edition, 2009.
2. E. Horowitz, S. Sahni and S. Rajsekaran, ***"Computer Algorithms"***, Silicon Press, 2nd Edition, 2008.

Reference Books:

1. B. K. Joshi, ***"Data Structures and Algorithms in C++"***, Tata McGraw Hill Education, 2010.
2. G. T. Heineman, Gary Pollice, Stanley Selkow, ***"Algorithms in a Nutshell"***, Shroff Publication, 1st Edition, 2008.
3. Kyle Loudon, ***"Mastering Algorithms with C"***, Shroff Publication, 1st Edition, 2008.

Course Title:	**Software Engineering**	**Semester V**	
Course Code	**BTITC503**	**Course Type**	**Core**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0– 0**
Stream	**Core**	**Credits**	**3**

Course Objectives:

1. To understand software lifecycle development models.
2. To understand and apply software requirements engineering techniques, software design principles, modeling and software testing techniques.
3. To understand the use of metrics in software engineering.
4. To understand software project management.

Course Outcomes:

After learning the course the students should be able:

1. To use the techniques, skills, and modern engineering tools necessary for engineering practice.
2. To design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability.
3. To identify, formulate and solve engineering problems.

Course Content:

UNIT I

Software Development Process: Software crisis and myths, Software process and development: Generic view of process, Software life cycle and models, Analysis and comparison of varies models, an agile view of process.

UNIT II

Requirement Engineering: Requirements engineering tasks, Initiating requirement engineering process, Eliciting requirement, developing use-cases, Building the analysis model, Negotiating and validating requirement, Building the analysis model.

UNIT III

System Design Overview: Design process and design quality, Design concepts, Design model, Pattern based software design, Architectural design, User interface design. UML: Different methods: Rambaugh / Booch / Jakobsons, Need for standardization. Developing diagrams in UML (Use CASE, Class, Interaction, State diagrams) CASE TOOLS.

UNIT IV

Validation and Testing: Strategic approach to Software testing, Strategic issues, Test strategies for conventional software, Validation testing, System testing, Debugging. White box testing and Black box testing.

UNIT V

Web Engineering: WebApps engineering layers, Web engineering processes planning for web engineering projects, Project management issue for web engineering. Metrics, Requirement analysis, Analysis models for web engineering design for WebApps, testing for WebApps.

UNIT VI

Planning and Management of Project: Project management, Metrics for process and projects, Estimation, Project scheduling, Risk management, Importance of software quality and measurements software engineering techniques for quality assurance, and Change management. ISO 9000 and CMM/PCMM.

Text Books

1. Roger S. Pressman, "***Software Engineering"***, Tata McGraw-Hill, 6th Edition, 2006.
2. G. Booch, J. Rambaugh, and I. Jacobson, ***"The Unified Modeling Language User Guide"***, Addison Wesley, 2nd Edition, 2005.

Reference Books:

1. Shari Pfleeger, ***"Software Engineering"***, Pearson Education, 3rd Edition, 2008.
2. Ian Sommerville, ***"Software Engineering"***, Pearson Higher Education, 10th Edition, 2016.
3. Pankaj Jalote, ***"An Integrated Approach to Software Engineering"***, Springer New York, 2nd Edition, 2013.

Course Title:	**Probability and Queuing Theory**	**Semester V**	
Course Code	**BTITC504**	**Course Type**	**Elective**
Pre-requisite	**Engineering Mathematics-III**	**L – T – P**	**3 – 0 – 0**
Stream	**Core**	**Credits**	**3**

Course Objectives:

1. Be through with probability concepts.
2. To acquire knowledge on Probability Distributions.
3. Get exposed to the testing of hypothesis using distributions.
4. Gain strong knowledge inn principles of Queuing theory.
5. Get exposed to Discrete time Markov chain.

Course Outcomes:

1. To acquire analytical ability in solving mathematical problems as applied to the respective branches of engineering.

Course Content:

UNIT I

Random Variables: Review of probability concepts, Types of Events, Axioms, Conditional probability, Multiplication theorem, Applications.

Discrete and continuous Random Variables – Discrete case, Probability Mass function, Cumulative distribution function, Applications, Characteristics of random variables – Continuous case, Probability density function, Cumulative distribution function, Applications, Expectation, Variance, Expectation, Variance, Moment Generating Function, Functions of Random Variable (One dimensional only).

Chebychev‟s Inequality – (Statement only). Applications of Chebychev's Inequality.

UNIT II

Theoretical Distributions:

Discrete Probability distribution: Binomial distribution – MGF, Mean, Variance, Applications of Binomial distribution, Fitting a Binomial distribution, Poisson distribution – MGF, Mean, Variance, Applications of Poisson distribution, Fitting a Poisson distribution, Geometric distribution – MGF, Mean, Variance, Memoryless Property , Applications of Geometric distribution, Continuous Probability Distributions: Uniform distribution – MGF, Mean, Variance & Applications, Exponential Distribution - MGF, Mean, Variance, Memoryless Property Applications of Exponential distribution, Normal distribution – Mean, Variance, Standard Normal distribution and Applications of Normal distribution

UNIT III

Testing of Hypothesis:

Introduction to Sampling Distributions, Population and Sample, Null Hypothesis and Alternative Hypothesis, Single and Two Tailed Test.

Testing of Hypothesis, Level of Significance, Critical Region, Procedure for Testing of Hypothesis

Large Sample Test- Test for Single Proportion, Two Sample Proportions.

Large Sample Test- Test for Single Mean, Two Sample Means.

Small Sample Tests – "t" Test For a Single Mean "t‟ Test For The Difference Of Means, Paired „t‟ Test

F Test – Test of Significance of the Difference between Two Population Variances.

Chi Square Test for Goodness of Fit, Independence of Attributes.

UNIT IV

Queuing Theory: Introduction to Markovian queuing models.
Single Server Model with Infinite system capacity, Characteristics of the Model (M/M/1): (∞/FIFO)
Problems on Model (M/M/1): (∞/FIFO), Problems on Model (M/M/1): (∞/FIFO), Single Server Model with Finite System Capacity, Characteristics of the Model (M/M/1): (K/FIFO), Problems on Model (M/M/1): (K/FIFO).

UNIT V

Markov Chains:
Introduction to Stochastic process, Markov process, Markov chain one step & n-step Transition Probability, TPM and Applications, Chapman Kolmogorov theorem (Statement only), Applications on Chapman Kolmogorov theorem.

UNIT VI

MARKOV CHAINS: Transition probability- Applications, Classification of states of a Markov chain, Classification of states of a Markov chain – Applications.

Text Books:

1. Veerarajan T., ***"Probability, Statistics and Random Processes"***, Tata McGraw Hill, 1st Reprint 2004.
2. S.C. Gupta and V.K. Kapoor, ***"Fundamentals of Mathematical Statistics"***, Sultan Chand & Sons, 9th extensively revised Eition,1999

Reference Books:

1. Trivedi K S, ***" Probability and Statistics with reliability, Queueing and Computer Science Applications***", Prentice Hall of India, New Delhi, 1984
2. Gross.D, Harris.C.M. , ***"Fundamentals of Queuing Theory"***, John Wiley and Sons, 1985.
3. Allen.A.O., ***"Probability Statistics and Queuing Theory"***, Academic Press, 1981

Course Title:	**Graph Theory**	**Semester V**	
Course Code	**BTITOE505A**	**Course Type**	**Elective**
Pre-requisite	**Discrete Structures and Applications**	**L – T – P**	**3 – 0 – 0**
Stream	**Departmental Elective**	**Credits**	**3**

Course Content:

UNIT I

Basics – Graphs, degree sequences, distance in graphs, complete, regular and bipartite graphs, basic properties.

UNIT II

Structure and Symmetry – Cut vertices, bridges and blocks, automorphism groups, reconstruction problem.

UNIT III

Trees and connectivity – Properties of trees, Arboricity, vertex and edge connectivity, Mengers theorem

UNIT IV

Eulerian and Hamiltonian graphs – Characterization of Eulerian graphs -Sufficient conditions for Hamiltonian graphs.

UNIT V

Colouring and planar graphs – vertex and edge colouring, perfect graphs, planar graphs, Euler's theorem, Kuratowski's theorem, Colouring of planar graphs, Crossing number and thickness.

UNIT VI

Matching, factors, decomposition and domination. Extremal Graph theory – Turan's theorem, Ramsay's theorem, Szemeredi's 97 regularity lemma, applications.

Text Books:

1. J. A. Bondy, U. S. R. Murthy, ***"Graph Theory"***, Springer Verlag, 2008.
2. D. B. West, ***"Introduction to Graph Theory"***, PHI, 2004.

Reference Books:

1. R. Diestel , ***"Graph Theory"***, Springer Verlag (Free Download available), 2003.

Course Title:	**Human Computer Interaction**	**Semester V**	
Course Code	**BTITOE505B**	**Course Type**	**Elective**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream	**Open Elective**	**Credits**	**3**

Course Content:

UNIT I

Introduction: The human, The computer, The interaction, Paradigms, Usability of Interactive Systems, Guidelines, Principles, and Theories.

UNIT II

Design Process- Interaction design basics, HCI in the software process, Design rules, Implementation support, Evaluation techniques, Universal design, User support.

UNIT III

Models and Theories, Cognitive models, Socio-organizational issues and stakeholder requirements, Communication and collaboration models, Task analysis, Dialogue notations and design, Models of the system, Modeling rich interaction.

UNIT IV

Interaction Styles- Direct Manipulation and Virtual Environments, Menu Selection, Form Filling and Dialog Boxes, Command and Natural Languages, Interaction Devices, Collaboration and Social Media Participation.

UNIT V

Design Issues- Quality of Service, Balancing Function and Fashion, User Documentation and Online Help, Information Search, Information Visualization.

UNIT VI

Outside the Box- Group ware, Ubiquitous computing and augmented realities, Hypertext, multimedia, and the World Wide Web.

Text Books:

1. Alan Dix, Janet Finlay, ***"Human Computer Interaction"***, Pearson Education, 2004.
2. Ben Shneiderman, ***"Designing the User Interface - Strategies for Effective Human Computer Interaction"***, Pearson Education, 2010.

Reference Books:

1. M. B. Rosson, J. M. Carroll ***"Usability Engineering: Scenario-Based Development of Human-Computer Interaction"***, Elsevier, 2002.
2. Alan Cooper, ***"The Essentials of Interaction Design"***, Wiley Publishing, 2007.
3. Nielsen, J. Morgan Kaufmann, San Francisco, ***"Usability Engineering"***, 1993.
4. Heim, S., ***"The Resonant Interface: HCI Foundations for Interaction Design"***, Addison-Wesley, 2007.

Course Title:	**Game Theory**	**Semester V**	
Course Code	**BTITOE505C**	**Course Type**	**Elective**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0– 0**
Stream	**Open Elective**	**Credits**	**3**

Course Objectives

1. The course is intended for students and teachers of institutions which offer undergraduate engineering programmes
2. The aim of the course is to provide an introduction to the study of game theory which has found wide applications in economics, political science, sociology, engineering apart from disciplines like mathematics and biology
3. The course would introduce to the fundamental tools of game theory, a few equilibrium concepts, apart from numerous exercises and applications
4. Knowledge of game theory would help students to understand and analyse real life situations such as market behaviour or voting in elections, apart from equipping them with analytical concepts which might be useful should they decide to pursue social sciences, engineering, sciences or managerial higher studies
5. This is an interdisciplinary course, hence not only social sciences but science and engineering departments of different universities can benefit from it
6. The six modules of the course are as follows,

a. Introduction to Game Theory
b. Strategic Games and Nash Equilibrium
c. Illustrations of Nash Equilibrium
d. Mixed Strategy Nash Equilibrium
e. Extensive Games and Nash Equilibrium
f. Illustrations of Extensive Games and Nash Equilibrium

Course Content:

UNIT I

Introduction to Game Theory

Concept of game theory, Theory of rational choice, Interacting decision makers

UNIT II

Strategic Games and Nash Equilibrium

Strategic games: examples, Nash equilibrium: concept and examples, Best response functions, Dominated Actions, Symmetric games and symmetric equilibrium.

UNIT III

Illustrations of Nash Equilibrium

Cournot’s model of duopoly market, Bertrand’s model of duopoly market, Electoral Competition, War of Attrition, Auctions, Accident Laws.

UNIT IV

Mixed Strategy Nash Equilibrium

Introduction, Strategic games with randomisation, Mixed strategy Nash equilibrium: concept and examples, Dominated Actions, Formation of Players’ beliefs.

UNIT V

Extensive Games and Nash Equilibrium

Introduction to extensive games, Strategies and outcomes, Nash equilibrium, Subgame perfect Nash equilibrium, Backward induction

UNIT VI

Illustrations of Extensive Games and Nash Equilibrium:

Stackelberg model of duopoly markets, Ultimatum game

References:

1. Osborne, M.J. , "***An Introduction to Game Theory***", Oxford University Press, 2004
2. Mas-Colell, A., M.D. Whinston and J.R. Green, "***Microeconomic Theory***", Oxford University Press, 1995
3. Gibbons, R., "***A Primer in Game Theory***", Pearson Education, 1992

Course Title:	**Embedded Systems**	**Semester V**	
Course Code	**BTITPE506A**	**Course Type**	**Elective**
Pre-requisite	**Microprocessor & Microcontroller**	**L – T – P**	**3 – 0 – 0**
Stream	**Software Application and Development**	**Credits**	**3**

Course Objectives:

1. To understand the fundamental concepts in Embedded Systems.
2. To learn Real Time Operating Systems.
3. To get acquainted with hardware & interfaces.
4. To know Embedded System Design Techniques.

Course Outcomes:

After learning the course the students should be able:

1. To demonstrate & explain embedded systems hardware & software components.
2. To define embedded systems using real time operating system – VxWorks/ µCOS II RTOS.
3. To design & develop embedded applications using C language.
4. To apply design techniques in real-life application.

Course Content:

UNIT I

Introduction: Introduction to embedded systems overview, design challenges, common design metrics, processor technology, IC technology, Design technology. Design productivity gap.

UNIT II

ARM Architecture: ARM 7 processor fundamentals, memory management, ARM processor family, Instruction set & interfacing. Introduction to ASIPS, Microcontrollers and DSP

UNIT III

Devices and Interfacing: Processor interfacing, Arbitration, Multilevel bus architecture. Basic protocol concepts: serial protocols, I2C,CAN, Firewire and USB, Parallel protocols, PCI bus, ARM bus, Wireless protocols: IrDA, Bluetooth, IEEE 802.11, Device Driver programming.

UNIT IV

Programming concepts: State m/c & concurrent process model, FSM m/c, FSMD, PSM model & concurrent process model, Scheduling process, Data flow model, Embedding programming in C++, JAVA and program modeling concepts.,

UNIT V

Real Time OS: OS services, Process management, Memory management device, File &IO subsystem management, Interrupt routines in RTOS, RTOS task scheduling models, Securities issues, RTOS mCOS-II & RTOS VxWorks.

UNIT VI

Design Examples and Case Studies: Personal Digital Assistants, Digital thermometer, Case Studies of digital camera, Smart card, Case study of coding for sending application layer byte stream on TCP/IP network using RTOS VxWorks.

Text Books

1. Frank Vahid/ Tony Givargis, ***"Embedded Systems Design"***, Wiley, 2002.
2. Raj Kamal, ***"Embedded Systems Architecture, and Programming"***, TMH Publication, 3rd Edition, 2015.
3. Andrew N. Sloss, ***"ARM System Developers Guide"***, ELSEVER Publication.

Reference Books:

1. Wayne Wolf, ***"Computer as Components – Principles of Embedded Computing System Design"***, Gulf Professional Publishing, 2nd Edition, 2008.
2. David E Simon, ***"An Embedded Software Primer"***, Addison Wesley Publication, 2004.

Course Title:	**IT Service Management**	**Semester V**	
Course Code	**BTITPE506B**	**Course Type**	**Elective**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream	**Infrastructure & Security Management**	**Credits**	**3**

Course Objectives:

1. To introduce practical implementation of Information Technology Service Management (ITSM).
2. To understand how an integrated ITSM framework can be utilized to achieve IT business integration, cost reductions and increased productivity.
3. To learn the best practices of ITSM methodology.

Course Outcomes:

After learning the course the students should be able:

1. To identify IT services as a means to provide functionality and value to customers.
2. To describe the needs and targets of the different stakeholders (service providers, customers, suppliers/partners) in the services value chain.
3. To demonstrate the value of a service management framework.
4. To explain the service management processes for given customers.
5. To select the appropriate tools to support a given designed service management solution.

Course Content:

UNIT I

IT Infrastructure: Introduction, Challenges in IT Infrastructure Management, Design Issues of IT Organizations and IT Infrastructure, IT System Management Process, IT Service Management Process, Information System Design Process.

UNIT II

Service Delivery Process: Service Level Management, Financial Management, IT Service Continuity Management, Capacity Management & Availability Management.

UNIT III

Service Support Process: Configuration Management, Incident Management, Problem Management, Change Management & Release Management.

UNIT IV

Storage Management: Storage, Backup, Archive and Retrieve, Disaster Recovery, Space Management, Database and Application Protection and Data Retention.

UNIT V

Security Management: Computer Security, Internet Security, Physical Security, Identity Management, Access Control System and Intrusion Detection.

UNIT VI

Case Studies on how IT Service Management and ITIL processes make IT efficient and save cost for organizations.

Text Books

1. Phalguni Gupta, Surya Prakash & Umarani Jayaraman, ***"IT Infrastructure & Its Management"***, Tata McGraw-Hill Education.

Reference Books:

1. W. Ronald Hudson, Ralph C. G. Haas, Waheed Uddin, ***"Infrastructure Management: Integrating Design, Construction, Maintenance, Rehabilitation, and Renovation"***, McGraw-Hill, 1997.
2. Anita Sengar, ***"IT Infrastructure Management"***, S.K. Kataria and Sons, 2nd Edition, 2009.

Course Title:	**Information Storage Management**	**Semester V**	
Course Code	**BTITPE506C**	**Course Type**	**Elective**
Pre-requisite	**Computer Architecture & Organization**	**L – T – P**	**3 – 0– 0**
Stream	**Information Management & Quality Control**	**Credits**	**3**

Course Objectives:

1. To evaluate storage architecture; understand logical and physical components of storage Infrastructure including storage subsystems.
2. To describe storage networking technologies such as FC-SAN, NAS, IP-SAN and data archival solution –CAS.
3. To identify different storage virtualization technologies and their benefits.
4. To understand and articulate business continuity solutions including, backup and recovery technologies, and local and remote replication solutions.
5. To define information security, and storage security domains and Identify parameters of managing and monitoring storage infrastructure and describe common storage management activities and solutions.

Course Outcomes:

After learning the course the students should be able:

1. To describe and apply storage technologies.
2. To identify leading storage technologies that provides cost-effective IT solutions for medium to large scale businesses and data centers.
3. To describe important storage technologies' features such as availability, replication, scalability and performance.
4. To design, analyze and manage clusters of resources.

Course Content:

UNIT I

Introduction to Information Storage Management - Intelligent Storage System (ISS) and its components Implementation of ISS as high-end and midrange storage-arrays, Direct Attached -Storage - Introduction to SCSI.

UNIT II

Introduction to parallel SCSI, SCSI Command Model – Storage Area Networks - Fiber Channel Connectivity, Login types, Topologies.

UNIT III

Storage networking technologies: Network-Attached Storage- General purpose servers vs. NAS Devices - Benefits of NAS, NAS File I/O – NAS Components, Implementation, File Sharing protocols, I/O operations – IPSAN-ISCSI, Components of ISCSI- Content-Addressed Storage.

UNIT IV

STORAGE VIRTUALIZATION: Fixed Content and Archives, Types, Features, Benefits, CAS Architecture, object storage and Retrieval, examples - Storage Virtualization-forms of virtualization, SNIA Taxonomy – Storage virtualization configurations, challenges, Types of storage virtualization - Business Continuity-Overview of emerging technologies such as Cloud storage, Virtual provisioning, Unified Storage, FCOE, FAST.

UNIT V

BUSINESS CONTINUITY AND RECOVERY: Information Availability, BC Terminology, Life cycle, Failure analysis - Backup and Recovery- Backup purpose, considerations, Backup Granularity, Recovery

considerations- Backup methods, process, backup and restore operations , Overview of emerging technologies - duplication, offsite backup.

UNIT VI

STORAGE SECURITY AND MANAGEMENT: Storage security framework, Securing the Storage infrastructure Risk triad - Managing the storage infrastructure, Monitoring the storage infrastructure, identify key parameters and components to monitor in a storage infrastructure List key management activities and examples Define storage management standards and initiative-Industry trend.

Text Books

1. EMC Corporation, ***"Information Storage and Management"***, Wiley India, 1st Edition, 2009.

Reference Books:

1. IBM, ***"Introduction to Storage Area Networks and System Networking"***, 5th edition, November 2012.
2. Robert Spalding, ***"Storage Networks: The Complete Reference"***, Tata McGraw Hill, Osborne, 6th reprint 2003.
3. Marc Farley, ***"Building Storage Networks"***, Tata McGraw Hill, Osborne, 1st Edition, 2001.
4. Tom Clark, ***"Designing Storage Area Networks -A Practical Reference for Implementing Fiber Channel and IP SANs",*** Tata McGraw Hill 2003, 2nd edition.

Course Title:	**Network Management**	**Semester V**	
Course Code	**BTITPE506D**	**Course Type**	**Elective**
Pre-requisite	**Internetworking Protocols**	**L – T – P**	**3 – 0– 0**
Stream	**Network**	**Credits**	**3**

Course Objectives:

1. To understand the principles of network management, different standards and protocols used in managing complex networks.
2. To understand the automation of network management operations and making use of readily available network management systems.

Course Outcomes:

After learning the course, the students should be able:

1. To acquire the knowledge about network management standards (OSI and TCP/IP).
2. To acquire the knowledge about various network management tools and the skill to use them in monitoring a network.
3. To analyze the challenges faced by Network managers.
4. To evaluate various commercial network management systems and open network management systems.
5. To analyze and interpret the data provided by an NMS and take suitable actions.

Course Content:

UNIT I

Data communication and network management overview: Analogy of Telephone Network Management, Communications protocols and Standards, Case Histories of Networking and Management, Challenges of Information Technology Managers, Network Management: Goals, Organization, and Functions, Network and System Management, Network Management System Platform, Current Status and future of Network Management.

UNIT II

SNMPV1 network management, Managed network: Organization and Information Models. Managed network: Case Histories and Examples, The History of SNMP Management, The SNMP Model, The Organization Model, System Overview, The Information Model.

UNIT III

SNMPV1 Network Management: Communication and Functional Models, The SNMP Communication Model, Functional model. SNMP MANAGEMENT: SNMPv2 Major Changes in SNMPv2, SNMPv2 System architecture, SNMPv2 Structure of Management Information, The SNMPv2 Management Information Base, SNMPv2 Protocol, Compatibility with SNMPv1.

SNMP MANAGEMENT: RMON: What is Remote Monitoring? RMON SMI and MIB, RMON1, RMON2, ATM Remote Monitoring, A Case Study of Internet Traffic Using RMON

UNIT IV

Telecommunication management network: Why TMN? Operations Systems, TMN Conceptual Model, TMN Standards, TMN Architecture, TMN Management Service Architecture, An Integrated View of TMN, Implementation Issues.

UNIT V

Network management tools and systems: Network Management Tools, Network Statistics Measurement Systems, History of Enterprise Management, Network Management systems, Commercial Network Management Systems, System Management and Enterprise Management Solutions.

UNIT VI

Web-Based Management: NMS with Web Interface and Web-Based Management, Web Interface to SNMP Management, Embedded Web-Based Management, Desktop management Interface, Web-Based Enterprise Management, WBEM: Windows Management Instrumentation, Java management Extensions, Management of a Storage Area Network, Future Directions. Case Studies:

Text Books:

1. Mani Subrahmanian, ***"Network Management Principles and Practice"***, Pearson Education, 2nd Edition, 2010.

Reference Books:

1. Morris, ***"Network management"***, Pearson Education, 1st Edition, 2008.
2. Mark Burges, ***"Principles of Network System Administration"***, Wiley DreamTech, 1st Edition, 2008.

Course Title:	**Data Visualisation**	**Semester V**	
Course Code	**BTITPE506E**	**Course Type**	**Elective**
Pre-requisite	**Database Management Systems**	**L – T – P**	**3 – 0– 0**
Stream	**Data Science**	**Credits**	**3**

Course Objectives:

1. Learn and understand the importance of data visualization.
2. Learn what is user experience in data visualization and its importance.
3. Learn about basic and advance chart types used in data visualization.
4. Learn the psychology of visualization with Gestalt Principles.

Course Outcomes:

After learning the course, the student will be able to:

1. Get a solid understanding of how people work in data visualization project.

Course Content:

UNIT I

The seven stages of Data Visualization: Why data display requires planning, An example, Iteration and Combination, Principles.
Getting Started with Processing: Sketching with processing, Example and Distributing your work, Examples and references, Functions, Sketching and Scripting
Mapping: Drawing a Map, Locations on map, Data on Map, Using your own data, Next step.

UNIT II

Time series:

Milk, Tea, and Coffee (Acquire and parse), Cleaning the table (Filter and Mine), A simple plot (Represent and refine), Labeling the current data set (Refine and Interact), Drawing Axis labels(Refine), Choosing a proper representation (Represent and refine), Using rollovers to Highlights points(Interact), Ways to connect points(refine), Text labels as tabbed panes(Interact), Interpolation between data sets(Interact).

UNIT III

Connections and Correlations:

Changing data sources, Problem statement, Preprocessing, Using the processed data (Acquire, Parse Filter and Mine), Displaying the results(Represent), Returning to the questions(Refine), Sophisticated sorting: Using salary as a Tiebreaker(Mine), Moving to multiple days(Interact), Smoothing out Interaction(Refine), Deployment Consideration (Acquire, Parse, filter).

UNIT IV

Scatterplot Maps: ++Preprocessing, Loading the data (Acquire and Parse), Drawing a scatterplot of Zip codes (Mine and represent), Highlighting Points while typing (Refine and Interact), Show the currently selected points(refine), Progressively Dimming and Brightening points(Refine), Zooming in (Interact), Changing How Points are Drawn when Zooming (Refine), Development issues (Acquire and Refine)

UNIT V

Trees, Hierarchies, and Recursion: Using recursion to build a Directory Tree, Using a Queue to Load Asynchronously (Interact), An improving the TreeMaps Display (Refine), Flying through files(Interact).

Networks and Graphs: A simple graph Demo, a more complicated Graph, Approaching Network Problem, Advanced graph example, Mining additional example.

UNIT VI

Acquiring Data: Where to find data, Tools for Acquiring data from Internet, Loading files for use with processing, Loading text data, Dealing with files and folders, Listing files in folders, Asynchronous Image download, Using openStream() As a bridge to Java, Dealing with Byte arrays, Advanced web techniques, Using Databases, Dealing with large number of files.

Parsing Data**:** Levels of efforts, Tools for gathering clues, Text is Best, Text Markup language, Regular expressions(regexps), Grammars and BNF Notations, Compressed Data, Vectors and Geometry, Binary data formats, Advanced detective work.

Text Books:

1. Ben Fry, ***"Visualizing Data: Exploring and Explaining data with Processing Environment"***, Shroff/O'Reilly Media, 2016

Reference Books:

1. Scott Murray, ***"Interactive Data Visualization for the web"***, Shroff/O'Reilly Media, 2016.
2. Julia Steele, Noah Lliinsky, ***"Designing Data Visualizations"***, Shroff/O'Reilly Media, 2012.
3. Kyran Dale, ***"Data Visualization with Python and JavaScript: Scrape, Clean, Explore & Transform your data"***, Shroff/O'Reilly Media, 2016.
4. Julia Steele, Noah Lliinsky, ***"Beautiful Visualization"***, Shroff/O'Reilly Media, 2016.

Course Title:	**Constitution of India**	**Semester V**	
Course Code	**BTHM501**	**Course Type**	**Audit**
Pre-requisite	**Nil**	**L – T – P**	**2 – 0– 0**
Stream		**Credits**	**0**

Course Content:

UNIT I

Introduction to Indian Constitution

Historical background, Philosophy of Indian Constitution, Preamble of Constitution- its forms and Importance, Features of Indian Constitution, The nature of Indian Federation

UNIT II

Fundamental Rights and Directive Principles

Fundamental Rights- its forms and importance, Fundamental rights in Constitution, Evaluation of Fundamental rights, Fundamental duties, Directive Principles of State Policies(Meaning, Objectives and Source), Classification of Directive Principles, Implementation of Directive Principles

UNIT III

Composition and Structure of Parliament

Function of Parliament, Law making Procedure, Executive Council structure and Role, State assembly, Changing Trends of Parliament

UNIT IV

Judiciary and Election Commission

Forms of Judiciary, Power, Function and Role of Supreme Court, Judicial Review, Judicial Activism, Structure, Function and Role of Election Commission, Electoral System and Reforms in it

UNIT V

Socialism of Constitution

Provision for Women Empowerment, Protection of Rights of Backward Class, Special Provision for Scheduled Tribes, Protection of Rights of workers, Socialistic democracy

UNIT VI

Democracy in India: Challenges

Constitutional Institutions and their role, Lokpal and Lokayukt, State Central Relation, Important Amendments, Nationalism, Criminalisation of Politics

Text Books:

1. D. D. Basu, "*Introduction to the Constitution of India*", LexisNexis Publishers, 23rd Edition, 2018.
2. B. Shiva Rao (Editor), "*Framing of Indian's Constitution*, *Select Documents*", Vol. 1, 2015.

Reference Books:

1. T. K. Tope, "*Constitutional Law of* India", Sujata V. Manohar (Editor), Eastern Book Company, 3rd Edition, 2010.

2. Sir Ivor Jennings, "*Some Characteristics of Indian Constitution*", Geoffrey Cumberlege Publishers, 1953.

Course Title:	**Programming Lab (R programming)**	**Semester V**	
Course Code	**BTITL507**	**Course Type**	**Mandatory**
Pre-requisite	**Nil**	**L – T – P**	**0 – 0 – 2**
Stream	**Core**	**Credits**	**1**

Lab Experiments Objective:

1. To learn R programming.

Lab Experiments List:

1. Download R programming language SDK and setup to run programs.
2. Develop and write a program to declare R variables, constants, operators and reserved words and understand the operator precedence.
3. Write a program to declare and understand the functioning of all the decision and loop constructs like If-Else, While, Break-Next and Repeat.
4. Execute all R functions.
5. Execute program to demonstrate Vectors, Matrix, data frame and factor.
6. Execute programs to test R Objects and Class.
7. Write a program to use and display various graphs and charts in R.
8. Execute programs to use plot in R.

Course Title:	**Database Management Systems Lab**	**Semester V**	
Course Code	**BTITL508**	**Course Type**	**Mandatory**
Pre-requisite	**Nil**	**L – T – P**	**0 – 0 – 2**
Stream	**Core**	**Credits**	**1**

Lab Experiments Objective:

1. To design a database adopting the principles of relational database model.
2. To practice and master DDL and DML through SQL.
3. To learn building efficient queries to interact with a database.

Lab Experiments List:

1. Creation of databases and use of SQL commands (DDL, DML and DCL).
2. Suitable exercises to practice SQL commands may be given for Insert, Update and Delete.
3. Write SQL procedure for an application which uses exception handling.
4. Write SQL procedure for an application with cursors.
5. Write SQL for implementing Nested Queries.
6. Write SQL for implementing Join Queries.
7. Write a DBMS program to prepare reports for an application using functions.
8. Write SQL block containing triggers.
9. Write SQL block containing stored procedures.
10. Develop a menu driven, GUI-based database application in any one of the domains such as Banking, Billing, Library management, Payroll, Insurance, Inventory, Healthcare etc. integrating all the features specified in the above exercises.

Course Title:	**Design and Analysis of Algorithms Lab**	**Semester V**	
Course Code	**BTITL509**	**Course Type**	**Mandatory**
Pre-requisite	**Nil**	**L – T – P**	**0 – 0 – 2**
Stream	**Core**	**Credits**	**1**

Lab Experiments Objective:

1. To design and develop various algorithms and analyze its efficiency to a specific problem.

Lab Experiments List:

1. Sort a given set of n integer elements using Quick Sort method and compute its time complexity. Run the program for varied values of n and record the time taken to sort. Plot a graph of the time taken versus non graph sheet. The elements can be read from a file or can be generated using the random number generator. Demonstrate how the divide and conquer method works along with its time complexity analysis: worst case, average case and best case.
2. Implement the 0/1 Knapsack problem using (a) Dynamic Programming method (b) Greedy method.
3. From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra's algorithm. Write the program.
4. Find Minimum Cost Spanning Tree of a given connected undirected graph using Kruskal's algorithm. Use Union-Find algorithms in your program.
5. Find Minimum Cost Spanning Tree of a given connected undirected graph using Prim's algorithm.
6. Write programs to (a) Implement All-Pairs Shortest Paths problem using Floyd's algorithm
   (b) Implement Travelling Sales Person problem using Dynamic programming.
7. Design and implement a program to find a subset of a given set S = Sl, S2,.....,Sn of n positive integers whose SUM is equal to a given positive integer d. For example, if S = 1, 2, 5, 6, 8 and d = 9, there are two solutions 1, 2,6and 1, 8. Display a suitable message, if the given problem instance doesn't have a solution.
8. Design and implement a program to find all Hamiltonian Cycles in a connected undirected Graph G of n vertices using backtracking principle.

Course Title:	**Seminar**	**Semester V**	
Course Code	**BTITS510**	**Course Type**	**Mandatory**
Pre-requisite	**Nil**	**L – T – P**	**0 – 0 – 2**
Stream	**Core**	**Credits**	**1**

Seminar topic is included to enable the students to apply their knowledge to understand advanced technologies, designs etc. Literature survey may help to select such topics which are invaluable to an engineer in an Information Technology industry. It will encourage students to develop their presentation skills, good communication skills and skills of collecting the correct information regarding the technical topic.

The students will be able to deliver seminar with useful information. He/she should under-stand the technologies, designs and skills of writing technical report, to do literature survey and to attempt the queries from examiner.

Course Title:	**Field Training/ Internship/ Industrial Training**	**Semester V**	
Course Code	**BTITF511**	**Course Type**	**Elective**
Pre-requisite		**L – T – P**	**0 – 0 – 0**
Stream		**Credit**	**1**

Assessment of Field Training/ Internship/ Industrial Training completed in 3rd and 4th semester or at one time.

Course Title:	**Operating Systems**	**Semester VI**	
Course Code	**BTITC601**	**Course Type**	**Mandatory**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream	**Core**	**Credits**	**3**

Course Objectives:

1. To study the basic concepts and functions of operating systems.
2. To understand the structure and functions of OS.
3. To learn about Processes, Threads and Scheduling algorithms.
4. To understand the principles of concurrency and Deadlocks.
5. To learn various memory management schemes.
6. To study I/O management and File systems.

Course Outcomes:

After learning the course, the students should be able:

1. To design various Scheduling algorithms.
2. To apply the principles of concurrency.
3. To design deadlock, prevention and avoidance algorithms.
4. To compare and contrast various memory management schemes.
5. To design and Implement a prototype file system.

Course Content:

UNIT I

Operating System Structures: Definition, Types of operating system, Real time operating system, System components, Sys-tem services, Systems calls, System programs, System structure, Virtual machines, System design and implementation.

UNIT II

Processes and CPU scheduling: Process concept, Process scheduling, Operation on a process, Co-operating processes, Threads, Interprocess communication, Scheduling criteria, Scheduling algorithms, Multiple-processor scheduling, Real-time scheduling, Scheduling algorithms and performance evaluation.

UNIT III

Process Synchronization: The critical-section problem, Critical regions, Synchronization hardware, Semaphores, Classical problems of synchronization, Monitors.

UNIT IV

Deadlocks: Systems model, Deadlock characterization, Methods for handling deadlocks, Deadlock prevention, Deadlock avoidance, Deadlock detection, Recovery from deadlock, Combined approach to deadlock handling.

UNIT V

Memory Management and Virtual Memory: Logical versus physical address space, Swapping, Contiguous allocation, Paging, Segmentation with paging, Demand paging, Page replacement algorithms, Thrashing.

UNIT VI

File Management: File system and secondary storage devices, Real-time operating systems.

Text Books

1. A. Silberschatz, P. Galvin, ***"Operating System Concepts"***, Wiley Publication, 9th Edition, 2013.
2. A. S. Tanenbaum, H. Bos, ***"Modern Operating Systems"***, Pearson Education, 4th Edition, 2015.

Reference Books:

1. D.M. Dhamdhere, ***"Systems Programming and Operating Systems"***, Tata McGraw Hill Publication, 2nd Edition, 2001.
2. G. Nutt, ***"Operating Systems Concepts"***, Addison Wesley Publication, 3rd Edition.
3. H. M. Deitel, ***"An Introduction to Operating Systems"***, Addison Wesley Publication, 1990.

Course Title:	**Compiler Construction**	**Semester VI**	
Course Code	**BTITC602**	**Course Type**	**Elective**
Pre-requisite	**Data Structures**	**L – T – P**	**3 – 0– 0**
Stream	**Core**	**Credits**	**3**

Course Objectives:

1. To introduce the major concept areas of language translation and compiler design.
2. To develop an awareness of the function and complexity of modern compilers.
3. To provide practical, hands on experience in compiler design.

Course Outcomes:

After learning the course, the students should be able:

1. To understand the major concept areas of language translation and compiler design.
2. To develop an awareness of the function and complexity of compilers.
3. To identify the similarities and differences among various parsing techniques and grammar transformation techniques.

Course Content:

UNIT I

Introduction to Automata Theory and Context Free Grammar

Definition of deterministic finite automata, Non-deterministic finite automata, Regular expressions, Recursive definition, NFA with e-moves, Regular expression and FA, Production rules, Ambiguous grammar, Removal of ambiguity, Chomsky hierarchy, Context Free Grammar (CFG) – definition, Simplification of CFG. Pushdown Automata- Definition, Non-deterministic PDA, Multi-stack PDA, Definition: Turing Machines, TM construction

UNIT II

Introduction to Compiling and Lexical Analysis:

Definition, analysis of the source program, the phases of a compiler, the grouping of phases, Compiler-Construction tools, Role of the Lexical analyzer, Input buffering, Specification of Tokens, A Language for Specifying Lexical Analyzers, Design of a Lexical Analyzer generator

UNIT III

Syntax Analysis:

The role of the Parser, Context-free grammars, Writing a Grammar, Top-Down Parsing, Bottom- Up Parsing, Operator-precedence Parsing, LR-Parsers, Using Ambiguous Grammars, Parser Generators.

UNIT IV

Syntax-Directed Translation:

Definitions, Construction of Syntax Trees, Bottom-Up Evaluation of S Attributed definitions, Top-Down Translation, Bottom-Up Evaluation of Inherited attributes.

UNIT V

Intermediate Code Generation:

Variants of syntax trees, Three address code, Types and declarations, Type checking, Control Flow and Backpatching.

UNIT VI

Code Generation and Code Optimization:

Code Generation: Issues in the design of code generation, The target language, Addresses in the target code, Different forms of object codes, Register allocation and assignments

Code Optimization: Need of code optimization, Principal sources of optimization, Optimization of basic blocks, Peephole optimization

Text Books:

1. Aho, Sethi, Ullman, ***"Compilers-Tools and Techniques"***, Pearson, 2nd Edition, 2011.
2. Tremblay, Sorenson, ***"Theory and Practice of Compiler Writing"***, McGraw Hill Publication.
3. Hopcroft, ***"Introduction to Automata Theory, Languages and Computation"***, Pearson Publication.

Reference Books:

1. Paul G. Sorenson, ***"Compiler Writing"***, Tata McGraw Hill.
2. Robin Hunter, ***"The Essence of Compilers"***, Pearson Publication, 1998.

Course Title:	**Object Oriented Software and Web Engineering**	**Semester VI**	
Course Code	**BTITC603**	**Course Type**	**Mandatory**
Pre-requisite	**Object Oriented Paradigm with C++**	**L – T – P**	**3 – 0 – 0**
Stream	**Core**	**Credits**	**3**

Course Objectives:

1. To learn the concept of Object Oriented Software Development Process.
2. To get acquainted with UML Diagrams.
3. To understand Object Oriented Analysis Processes.
4. Understand the characteristics of web application.
5. Learn to Model web applications.
6. Be aware of Systematic methods.
7. Be familiar with the testing techniques for web applications.

Course Outcomes:

After learning the course, the students should be able to:

1. Understand Object Oriented Software Development Process.
2. Gain exposure to Object Oriented Methodologies & UML Diagrams.
3. Apply Object Oriented Analysis Processes for projects.
4. Apply the characteristics of web applications.
5. Model web applications.
6. Design web applications.
7. Test web applications.

Course Content:

UNIT I

Object Basics, Object oriented philosophy, objects, classes, attributes, object behavior and methods, encapsulation and information hiding, class hierarchy, polymorphism, object relationships and associations, aggregations and object containment, case study, object identity, persistence, Object oriented systems development life cycle: Software development process, building high quality software, use- case driven approach, reusability.

UNIT II

Object Oriented Methodologies: Rumbaugh et al.’s object modeling technique, Booch methodology, Jacobson et al methodologies, patterns, frameworks, and the unified approach. Unified modeling language: Static and dynamic models, UML diagrams, UML class diagrams, use-case diagrams, UML dynamic modeling, packages, UML extensibility and UML Meta model.

UNIT III

Object Oriented Analysis Process: Business object analysis, use-case driven object oriented analysis, business process modeling, use-case model, developing effective documentation, case study. Classification: Classification theory, noun phrase approach, common class patterns approach, use-case driven approach, classes, responsibilities, and collaborators, naming classes.

UNIT IV

Identifying Object Relationships, Attributes and Methods: Association, super-subclass relationships, a-part of relationships, case study, class responsibility, Defining attributes for vianet bank objects, object responsibility, defining methods for vianet bank objects Design process and design axioms: Corollaries, design patterns.

Designing Classes: UML object constraint languages, designing classes, class visibility, refining attributes for the vianet bank objects, designing methods and protocols, designing methods for the vianet bank objects, packages and managing classes. Designing access layer, Designing view layer, macro level process.

UNIT V

Introduction to Web Engineering and requirement engineering: Motivation, Categories of Web Applications, Characteristics of Web Applications, Product-related Characteristics, Usage related Characteristics, Development-related Characteristic,

Web Application Architecture and Modelling Web Applications: Introduction- Categorizing Architectures, Specifics of Web Application Architectures, Components of a Generic Web Application Architecture, Layered Architectures, 2-Layer Architectures, N-Layer Architectures Data-aspect Architectures, Database-centric Architectures, Architectures for Web Document Management, Architectures for Multimedia Data Modeling Specifics in Web Engineering, Levels, Aspects, Phases Customization, Modeling Requirements, Hypertext Modeling,

UNIT VI

Web Application Design: Introduction, Web Design from an Evolutionary Perspective, Information Design, Software Design: A Programming Activity, Merging Information Design and Software Design, Problems and Restrictions in Integrated Web Design, A Proposed Structural Approach, Presentation Design, Presentation of Nodes and Meshes,

Testing Web Applications: Introduction, Fundamentals, Terminology, Quality Characteristics, Test Objectives, Test Levels, Role of the Tester, Test Specifics in Web Engineering, Test Approaches, Conventional Approaches, Agile Approaches, Test Scheme, Three Test Dimensions, Applying the Scheme to Web Applications, Test Methods and Techniques, Link Testing, Browser Testing, Usability Testing, Load, Stress, and Continuous Testing, Testing Security, Test-driven Development, Test Automation.

Text Books

1. Ali Bahrami, ***"Object Oriented Systems Development using the Unified Modeling Language"***, McGraw Hill, Reprint, 2009.
2. Craig Larman, ***"Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development"***, Pearson Education, 3rd Edition, 2005.
3. Gerti Kappel, Birgit Proll, ***"Web Engineering"***, John Wiley and Sons Ltd, 2006.
4. Roger S. Pressman, David Lowe, ***"Web Engineering"***, Tata McGraw Hill Publication, 2007.
5. Guy W. Lecky-Thompson, ***"Web Programming"***, Cengage Learning, 2008.

Reference Books:

1. Bernd Oestereich, ***"Developing Software with UML, Object-Oriented Analysis and Design in Practice"***, Addison-Wesley, 2000.
2. James Rumbaugh, Ivar Jacobson, Grady Booch, ***"The Unified Modeling Language Reference Manual"***, Addision Wesley, 2nd Edition, 2005
3. Simon Bennett, Steve Mc Robb and Ray Farmer, ***"Object Oriented Systems Analysis and Design Using UML"***, McGraw Hill Education, 4th Edition, 2010.

4. Erich Gamma, and Richard Helm, Ralph Johnson, John Vlissides, ***"Design Patterns: Elements of Reusable Object-Oriented Software"***, Addison-Wesley, 1995.
5. Chris Bates, **"Web Programming: Building Internet Applications"**, Third Edition, Wiley India Edition, 2007.
6. John Paul Mueller, **"Web Development with Microsoft Visual Studio 2005"**, Wiley Dream tech, 2006.

Course Title:	**Digital Image Processing**	**Semester VI**	
Course Code	**BTITC604**	**Course Type**	**Elective**
Pre-requisite		**L – T – P**	**3 – 0 – 0**
Stream	**Core**	**Credits**	**3**

Course Objectives:

1. To cover the fundamentals and mathematical models in digital image and video processing.
2. To develop time and frequency domain techniques for image enhancement.
3. To expose the students to current technologies and issues in image and video processing.
4. To develop image and video processing applications in practice.

Course Outcomes:

At the end of this course, students will be able to:

1. Understand theory and models in Image and Video Processing.
2. Interpret and analyze 2D signals in frequency domain through image transforms.
3. Apply quantitative models of image and video processing for various engineering applications.
4. Develop innovative design for practical applications in various fields.

Course Content:

UNIT I

Image fundamentals: Image acquisition, sampling and quantization, image resolution, basic relationship between pixels, color images, RGB, HSI and other models.

UNIT II

Two dimensional transforms: 2D-Discrete fourier transform, discrete cosine transform, Walsh Hadamard transform, Haar transform, KL transform, and discrete wavelet transform.

UNIT III

Spatial domain Processing: Point processing such as digital negative, contrast stretching, thresholding, gray level slicing, bit plane slicing, log transform and power law transform, neighbourhood processing such as averaging filters, order statistics filters, high pass filters and high boost filters, histogram equalization and histogram specification, frequency domain such as DFT for filtering, ideal, Gaussian and butterwort filters for smoothening and sharpening, and homomorphic filters.

UNIT IV

Image segmentation and morphology: Point, line and edge detection, edge linking using Hough transform and graph theoretic approach, thresholding, and region based segmentation, dilation, erosion, opening, closing, hit or miss transform, thinning and thickening, and boundary extraction on binary images.

UNIT V

Degradation model, noise models, estimation of degradation function by modelling, restoration using Weiner filters and inverse filters.

UNIT VI

Video formation, perception and representation: Digital video sampling, video frame classifications, I, P and B frames, notation, ITU-RBT 601 digital video formats, digital video quality measure, video capture and display: principle of colour video camera, video camera, digital video, sampling of video signals: required sampling

rates, sampling in two dimensions and three dimensions, progressive virus interlaced scans, two dimensional motion estimation, block matching algorithms.

Text Books:

1. Gonzales and Woods, "*Digital Image Processing*", Pearson Education, India, Third Edition.
2. Anil K.Jain, "*Fundamentals of Image Processing*", Prentice Hall of India, First Edition, 1989.

Reference Books:

1. Ze-Nian Li and Mark S. Drew, "*Fundamentals of Multimedia*", PHI 2011.
2. Murat Tekalp, "*Digital Video Processing*", Pearson, 2010.
3. John W. Woods, "*Multidimensional Signal, Image and Video Processing*", Academic Press 2012.
6. A.I.Bovik, "*Handbook on Image and Video Processing*", Academic Press.

Course Title:	**Enterprise Resource Planning**	**Semester VI**	
Course Code	**BTITOE605A**	**Course Type**	**Elective**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0– 0**
Stream	**Open Elective**	**Credits**	**3**

Course Objectives:

1. To introduce to enterprise systems and show how organizations use enterprise systems to run their operations more efficiently and effectively.
2. To learn about the critical success factors and implementation strategies that lead to enterprise system success.
3. To learn about the informational, knowledge, and decision-making opportunities afforded by enterprise systems.
4. To examine typical Enterprise Systems modules: materials management (MM), supply chain management (SCM), customer relationship management (CRM), financials, projects, human resource management (HRM).

Course Outcomes:

After learning the course, the students should be able:

1. To demonstrate a good understanding of basic issues in Enterprise Systems.
2. To explain the scope of common Enterprise Systems (e.g., MM, SCM, CRM, HRM, procurement).
3. To explain the challenges associated with implementing enterprise systems and their impacts on organizations.
4. To describe the selection, acquisition and implementation of enterprise systems.
5. To use one of the popular ERP packages to support business operations and decision-making.
6. To communicate and assess an organization's readiness for enterprise system implementation with a professional approach in written form.
7. To demonstrate an ability to work independently and in a group.

Course Content:

UNIT I

Enterprise Resource Planning: Introduction, Disadvantages of non-ERP systems, What Is ERP? Need of ERP, Advantage of ERP, Risks of ERP, Growth of ERP.

UNIT II

ERP Modules: Finance, Production Planning, Control and Management, Sales and Distribution, Human Resource Management, Inventory Control System, Quality Management, Plant Maintenance.

UNIT III

ERP Implementation: ERP Implementation (Transition) strategies, ERP Implementation Life Cycle, Implementation Methodologies, Evaluation and selection of ERP package, ERP Project Team: Vendors, Employees, Consultants, Training & Education, Project management & Monitoring, Post Implementation Activities, Operation & maintenance of ERP system, Measuring the Performance of ERP System, Success & failure factors of an ERP, Implementation.

UNIT IV

ERP Market and Vendors: ERP Marketplace and Marketplace Dynamics, Comparison of Current ERP Packages and Vendors, like; SAP, Oracle, PeopleSoft, BAAN etc.

UNIT V

ERP and related technologies: Business Process Re-Engineering (BPR), Information Systems -Management Information, System (MIS), Decision Support System (DSS), Executive Support System (ESS) Data Warehousing, Data Mining, On-Line Analytical Processing (OLAP), Supply Chain Management, Customer Relationship Management

UNIT VI

ERP Case Studies: ERP systems implemented in: TISCO, SKF Automotive Bearings Co. Ltd, Qualcomm CDMA, California, Post Implementation review of ERP packages in Manufacturing, Services and Others Organizations, Customization of ERP for different types of Industries.

Text Books

1. Alexis Leon, ***"ERP Demystified"***, TMH New Delhi, 2nd Edition.
2. V. K. Garg & N. K. Venkita Krishnan, ***"ERP Ware: ERP Implementation Framework"***, PHI.

Reference Books:

1. V. K. Garg & N. K. Venkita Krishna, ***"ERP Concepts & Planning"***, PHI, 2nd Edition.

Course Title:	**Decision Support Systems**	**Semester VI**	
Course Code	**BTITOE605B**	**Course Type**	**Elective**
Pre-requisite	**Database Management Systems**	**L – T – P**	**3 – 0 – 0**
Stream	**Open Elective**	**Credits**	**3**

Course Objectives:

1. To select appropriate modeling techniques for supporting semi-structured business decision making.
2. To identify and select appropriate decision support systems for generating innovative business solutions.
3. To design and implement decision support systems for generating innovative business solutions.

Course Outcomes:

After learning the course, the students should be able:

1. To recognize the relationship between business information needs and decision making.
2. To appraise the general nature and range of decision support systems.
3. To appraise issues related to the development of DSS.
4. To select appropriate modeling techniques.
5. To analyze, design and implement a DSS.

Course Content:

UNIT I

Basic Concepts: Decision making systems, Modeling and support, Basics and definition Systems models, Modeling process, Decision making, Intelligence phase, Design phase Choice phase, Evaluation, Implementation phase, Alternative decision making models, Decision support systems, Decision makers, Case applications.

UNIT II

Decision Support System Development: Decision support system development, Basics, Life cycle, Methodologies, Prototype, Technology levels and tools, Development platforms, Tool selection, Developing DSS, Enterprise systems, Concepts and definition, Evolution of information systems, Information needs, Characteristics and capabilities, Comparing and integrating EIS and DSS, EIS data access, Data warehouse, OLAP, Multidimensional analysis, Presentation and the Web, Including soft information enterprise on systems, Organizational DSS, Supply and value chains, Decision support, Supply chain problems and solutions, Computerized systems. MRP, ERP, SCM, Frontline decision support systems.

UNIT III

Knowledge Management: Organizational learning and memory, Knowledge management, Development Methods, Technologies and tools, Success , Knowledge management and artificial intelligence, Electronic Document Management, Knowledge Acquisition and Validation, Knowledge Engineering – Scope, Acquisition Methods, Interviews, Tracking Methods, Observation and other Methods, Grid Analysis, Machine Learning, Rule Induction, Case-Based Reasoning, Neural Computing, Intelligent Agents, Selection of an appropriate Knowledge Acquisition Methods, Multiple Experts, Validation and Verification of the Knowledge Base-Analysis, Coding, Documenting, and Diagramming, Numeric and Documented.

UNIT IV

Knowledge Acquisition, Knowledge Acquisition and the Internet/Intranets, Knowledge Representation Basics, Representation in Logic and other Schemas, Semantic Networks, Production Rules, Frames, Multiple Knowledge Representation, Experimental Knowledge Representations, Representing Uncertainty. Intelligent

System Development: Inference Techniques, Reasoning in Artificial Intelligence, Inference with Rules, Inference Tree, Inference with Frames, Model Based and Case Based Reasoning, Explanation and Meta Knowledge, Inference with Uncertainty, Representing Uncertainty, Probabilities and Related Approaches, Theory of Certainty, Approximate Reasoning using Fuzzy Logic

UNIT V

Intelligent Systems Development, Prototyping, Project Initialization, System Analysis and Design, Software Classification, Building Expert Systems with Tools, Shells and Environments, Software Selection, Hardware, Rapid Prototyping and a Demonstration Prototype, System Development, Implementation, Post Implementation.

UNIT VI

Management Support Systems: Implementing and Integrating Management Support Systems, Implementation, Major Issues, Strategies, System Integration, Generic Models MSS, DSS–ES, Integrating EIS, DSS and ES, Global Integration, Intelligent DSS, Intelligent Modeling and Model Management, Examples of Integrated Systems, Problems and Issues in Integration.

Text Books

1. Efrain Turban and Jay E. Aronson, "***Decision Support Systems and Intelligent Systems"***, Pearson Education, 6th Edition,2001.

Reference Books:

1. Ganesh Natarajan and Sandhya Shekhar, ***"Knowledge Management Enabling Business Growth"***, Tata McGraw Hill, 2002.
2. George M. Marakas, ***"Decision Support System"***, Prentice Hall, India, 2003.
3. Efrem A. Mallach, ***"Decision Support and Data Warehouse Systems"***, Tata McGraw, Hill, 2002.
4. Kimiz Dalkir, ***"Knowledge Management: Theory and Practice"***, Elsevier Science, 2005.
5. Becerra Fernandez and Laidener, ***"Knowledge Management: An Evolutionary View"***, PHI, 2009.

Course Title:	**Software Project Management**	**Semester VI**	
Course Code	**BTITOE605C**	**Course Type**	**Elective**
Pre-requisite	**Software Engineering**	**L – T – P**	**3 – 0 – 0**
Stream	**Open Elective**	**Credits**	**3**

UNIT I

Project Evaluation and Planning - Activities in Software Project Management, Overview of Project Planning, Stepwise planning, contract management, Software processes and process models.

UNIT II

Cost Benefit Analysis, Cash Flow Forecasting, Cost-Benefit Evaluation Techniques, Risk Evaluation. Project costing, COCOMO 2, Staffing pattern, Effect of schedule compression, Putnam's equation, Capers Jones estimating rules of thumb, Project Sequencing and Scheduling Activities, Scheduling resources, Critical path analysis, Network Planning, Risk Management, Nature and Types of Risks, Managing Risks, Hazard Identification, Hazard Analysis, Risk Planning and Control, PERT and Monte Carlo Simulation techniques.

UNIT III

Monitoring and Control- Collecting Data, Visualizing Progress, Cost Monitoring, review techniques, project termination review, Earned Value analysis, Change Control

UNIT IV

Software Configuration Management (SCM), Managing Contracts, Types of Contracts, Stages In Contract Placement, Typical Terms of A Contract, Contract Management and Acceptance.

UNIT V

Quality Management and People Management- Introduction, Understanding Behavior, Organizational Behaviour, Selecting The Right Person for The Job, Motivation, The Oldman – Hackman Job Characteristics Model, working in Groups, Organization and team structures, Decision Making, Leadership, Organizational Structures, Stress, Health and Safety. ISO and CMMI models, Testing, and Software reliability, test automation.

UNIT VI

Overview of project management tools.

Text Books:

1. Bob Hughes, Mike Cotterell, ***"Software Project Management"***, Tata McGraw Hill, 2009.

Reference Books:

1. Royce, ***"Software Project Management"***, Pearson Education, 2005.
2. Robert K. Wysocki, ***"Effective Software Project Management"***, Wiley, 2006.

Course Title:	**Software Testing**	**Semester VI**	
Course Code	**BTITPE606A**	**Course Type**	**Elective**
Pre-requisite	**Software Engineering**	**L – T – P**	**3 – 0 – 0**
Stream	**Software Application & Development**	**Credits**	**3**

Course Objectives:

1. To study fundamental concepts in software testing, including software testing objectives, processes, criteria, strategies, and methods.
2. To learn planning of a test project, designing test cases and test data, conducting test operations, managing software problems and defects, and generating a test report.
3. To develop an understanding of the meaning and importance of quality in relation to software systems and the software development process.
4. To study issues and techniques for implementing and managing software quality assurance processes and procedures.

Course Outcomes:

After learning the course, the students should be able:

1. To apply software testing knowledge and its processes to software applications.
2. To identify various software testing problems.
3. To solve software testing problems by designing and selecting software test models, criteria, strategies and methods.
4. To apply the techniques learned to improve the quality of software development.
5. To prepare a software quality plan for a software project.

Course Content:

UNIT I

Principles of Testing Software development life cycle model: Phases of software project, Quality, Quality assurance and quality control, Testing, Verification and validation, Process models to represent various phases, Life cycle models, Software testing life cycle.

UNIT II

White Box Testing (WBT) and Black Box Testing: Static testing, Structural testing, Challenges in WBT. Black box testing: Black box testing process.

UNIT III

Integration Testing: Definition, As a type of testing: Top-down integration, Bottom-up integration, Bi-directional integration, System integration, Choosing integration method, As a phase of testing, Scenario testing: System scenarios, Use case scenarios, Defect bash.

UNIT IV

System and Acceptance Testing, Functional Vs non Functional, Functional system testing, Non- functional system testing, Acceptance testing.

UNIT V

Performance testing, Regression testing, Internationalization testing, Adhoc testing. Factors governing performance of testing, Methodology, tools and process for performance testing. Regression Testing: Introduction, Types of Regression testing, Regression testing process. Adhoc testing: Introduction, Buddy testing, Pair testing, exploratory testing, Iterative testing, Agile and Extreme testing, XP work flow, Defect seeding.

UNIT VI

Testing Object Oriented Software: Introduction, Comparison of object oriented and procedural software, System testing example, Unit testing of classes, Tools for testing object oriented software, Testing web applications.

Text Books

1. Srinivasan Desikan, Gopalaswamy Ramesh, ***"Software Testing: Principles and Practices"***, Pearson publication, 2nd Edition, 2006.

Reference Books:

1. Loise Tamres, ***"Introducing Software Testing"***, Pearson publication, 2002.
2. Boris Beizer, ***"Software Testing Techniques"***, Dreamtech press, 2nd Edition, 2014

Course Title:	**Data Storage Technologies & Networks**	**Semester VI**	
Course Code	**BTITPE606B**	**Course Type**	**Elective**
Pre-requisite	**Internetworking Protocols, Operating Systems**	**L – T – P**	**3 – 0 – 0**
Stream	**Infrastructure & Security Management**	**Credits**	**3**

Course Objectives:

1. To gain knowledge and understand the design of a Data Centre.
2. To understand the best practice of design in the Data Centre.
3. To learn the options in the running of an efficient Data Centre.
4. To understand the value of data to a business, Information Lifecycle.
5. To understand the challenges in data storage and data management.
6. To learn solutions available for data storage.

Course Outcomes:

After learning the course, the students should be able:

1. To explain the design of a data center and storage requirements.
2. To discuss the various types of storage and their properties.
3. To explain physical and virtualization of storage.
4. To explain the backup, archiving with regard to recovery and business continuity.

Course Content:

UNIT I

Data Centre: Introduction, Site Selection and Environmental Considerations, Hierarchical or Layered Architecture, Architect Roles, Goals and Skills, Architecture Precursors.

UNIT II

Data Centre Design: Architecture Design and Standards Recommendations, Raised Access Floor and Design Best Practices, connecting the infrastructure with copper and fiber. IT Hardware, Cooling System Options and Environmental Control, Electrical Power Systems, Room Layout, Fire Protection and Security Systems, Building Automation and Energy Management Systems, Commissioning and Handover.

UNIT III

Storage Management: Introduction to Storage Technology, Storage Systems Architecture, Physical and logical components of a connectivity environment, Major physical components of a disk drive and their functions, Concept of RAID and its components, Different RAID levels and their suitability for different application environments: RAID 0, RAID 1, RAID 3, RAID 4, RAID 5, RAID 0+1, RAID 1+0, RAID 6, Integrated and Modular storage systems, high-level architecture and working of an intelligent storage systems.

UNIT IV

Networked Storage: Evolution of networked storage, Architecture, components, and topologies of FC-SAN, NAS, and IP-SAN, Benefits of the different networked storage options, Need for long-term archiving solutions and describe how CAS fulfill the need, Appropriateness of the different networked storage options for different application environments.

UNIT V

Managing Data Center: Reasons for planned/unplanned outages, Impact of downtime, Difference between business continuity (BC) and disaster recovery (DR), RTO and RPO, Identification of single points of failure in a storage infrastructure and solutions to mitigate these failures, Architecture of backup/recovery and the

different backup/recovery topologies, replication technologies and their role in ensuring information availability and business continuity Remote replication technologies and their role in providing disaster recovery and business continuity capabilities, Key areas to monitor in a data center, Industry standards for data center monitoring and Management Key metrics to monitor storage infrastructure.

UNIT VI

Securing Storage and Storage Virtualization: Information Security, Critical security attributes for information systems, Storage security domains, Analyze the common threats in, each domain, Storage Virtualization: Forms, Configurations and Challenges, Types of Storage Virtualization: Block-level and File-Level.

Text Books

1. Mauricio Arregoces, ***"Data Center Fundamentals"***, Cisco Press, 1st edition, 2003.
2. Robert Spalding, ***"Storage Networks: The Complete Reference"***, Tata McGraw Hill, Osborne, 2003.
3. Marc Farley, ***"Building Storage Networks"***, Tata McGraw Hill, Osborne. 2001.
4. Meeta Gupta, ***"Storage Area Network Fundamentals"***, Pearson Education Limited, 2002

Reference Books:

1. G. Somasundaram, Alok Shrivastava, ***"Information Storage and Management"***, EMC Education Series, Wiley Publishing Inc., 2011.
2. Gustavo Santana, ***"Data Center Virtualization Fundamentals: Understanding Techniques and Designs for Highly Efficient Data Centers with Cisco Nexus, UCS, MDS, and Beyond"***, Cisco Press, 1st Edition, 2013

Course Title:	**Service Oriented Architecture**	**Semester VI**	
Course Code	**BTITPE606C**	**Course Type**	**Elective**
Pre-requisite	**Nil**	**L – T – P**	**3 – 0 – 0**
Stream	**Information Management & Quality Control**	**Credits**	**3**

Course Objectives:

1. To gain understanding of the basic principles of service orientation.
2. To learn service oriented analysis techniques.
3. To learn technology underlying the service design.
4. To learn advanced concepts such as service composition, orchestration and Choreography.
5. To know about various WS specification standards.

Course Outcomes:

After learning the course, the students should be able:

1. Build applications based on XML.
2. Develop web services using technology elements.
3. Build SOA-based applications for intra-enterprise and inter-enterprise applications.

Course Content:

UNIT I

Introducing SOA: Fundamental SOA: Common Misperceptions about SOA, Common tangible benefits of SOA, Common pitfalls of adopting SOA, The Evolution of SOA:-from XML to Web services to SOA, The continuing evolution of SOA, The roots of SOA. Web Services and Primitive SOA: The Web services framework-Services, Service descriptions, messaging with SOAP.

UNIT II

Web Services and Contemporary SOA: Message exchange patterns- Service activity-coordination-Atomic transactions-Business activities-Orchestration-Choreography- Web Services and Contemporary SOA: Addressing- Reliable messaging-Correlation- Policies- Metadata exchange- Security- Notification and eventing.SOA and Service-Orientation: Principles of Service - Anatomy of a service-oriented architecture-Common principle of service orientation-Service Layers –Service orientation.

UNIT III

Building SOA: SOA Delivery Strategies- SOA delivery lifecycle phases. Service-Oriented Analysis: Introduction to service-oriented Analysis-Benefits of a business-centric SOA- Deriving business services-Service-Oriented Analysis: Service modeling, Service modeling guidelines- Classifying service model logic-Contrasting service modeling approaches.

UNIT IV

Service-Oriented Design: Introduction to service-oriented design- WSDL-related XML Schema language basics- WSDL language basics- SOAP language basics- Service interface, design tools. SOA Composition Guidelines: Steps to composing SO Considerations for choosing service layers and SOA standards, positioning of cores and SOA extensions.

UNIT V

SOA Service Design: - Overview-Service design of business service, application service, task centric service and guidelines. SOA Business Process Design: WS-BPEL language basics-WS Coordination.

UNIT VI

SOA support in J2EE – Java API for XML-based web services (JAX-WS) - Java architecture for XML binding (JAXB) – Java API for XML Registries (JAXR) - Java API for XML based RPC (JAX-RPC)- Web Services Interoperability Technologies (WSIT)

Text Books

1. Thomas Erl, ***"Service-Oriented Architecture: Concepts, Technology, and Design"***, Pearson Education, 2006.
2. Frank. P. Coyle, ***"XML, Web Services and The Data Revolution"***, Pearson Education, 2002.
3. Sandeep Chatterjee, James Webber, ***"Developing Enterprise Web Services. An Architect's Guide"***, Pearson Education, 2005.
4. Eric Newcomer, Greg Lomow, ***"Understanding SOA with Web Services"***, Pearson Education, 2005.
5. Ron Schmelzer et al. ***"XML and Web Services"***, Pearson Education, 2002

Reference Books:

1. Dan woods and Thomas Mattern, ***"Enterprise SOA designing IT for Business Innovation"***, O'REILLY, 1st Edition, 2006.
2. James McGovern, Sameer Tyagi, Michael E. Stevens, Sunil Mathew, ***"Java Web. Services Architecture"***, Morgan Kaufmann Publishers, 2003.
3. Atul Kahate, ***"XML and Related technologies"***, Pearson Education, 2008.
4. Kennard Scibner and Mark C. Stiver, ***"Understanding SOAP"***, SAMS publishing.
5. B. V. Kumar, S. V. Subrahmanya, ***"Web Services: An Introduction"***, TMH India, 2nd Edition, 2012.

Course Title:	**Network Programming**	**Semester VI**	
Course Code	**BTITPE606D**	**Course Type**	**Elective**
Pre-requisite	**Internetworking Protocols, Operating Systems**	**L – T – P**	**3 – 0 – 0**
Stream	**Network**	**Credits**	**3**

Course Objectives:

1. To learn the basics of socket programming using TCP Sockets.
2. To learn about Socket Options.
3. To learn to develop Macros for including Objects in MIB Structure.
4. To understand SNMPv1, v2 and v3 protocols & practical issues.

Course Outcomes:

After learning the course, the students should be able:

1. To analyze the requirements of a networked programming environment and identify the issues to be solved;
2. To create conceptual solutions to those issues and implement a programming solution;
3. To understand the key protocols that support the Internet;
4. To apply several common programming interfaces to network communication;
5. To understand the use of TCP/UDP Sockets
6. To apply advanced programming techniques such as Broadcasting, Multicasting.

Course Content:

UNIT I

Socket and Application Devlopment: Introduction to Socket Programming - System Calls - Address conversion functions - POSIX Signal Handling - Server with multiple clients - Boundary conditions - Server process Crashes, Server host Crashes, Server Crashes and reboots, Server Shutdown - I/O Multiplexing - I/O Models -TCP echo client/server with I/O Multiplexing

UNIT II

Socket Optian: Socket options - getsockopt and setsockopt functions - Generic socket options - IP socket options -ICMP socket options - TCP socket options - Multiplexing TCP and UDP sockets - SCTP Sockets -SCTP Client/server - Streaming Example - Domain name system - gethostbyname, gethostbyaddr, getservbyname and getservbyport functions - Protocol Independent functions in TCP Client/Server Scenario

UNIT III

Advanced Socket: IPv4 and IPv6 interoperability - Threaded servers - Thread creation and termination - TCP echo server using threads - Mutex - Condition variables - Raw sockets - Raw socket creation - Raw socket output - Raw socket input - ping program - traceroute program

UNIT IV

Simple Network Management: SNMP network management concepts - SNMPv1 - Management information - MIB Structure – Object syntax - Standard MIB’s - MIB-II Groups - SNMPv1 protocol and Practical issues.

UNIT V

SNMP V2, V3 and RMO: Introduction to SNMPv2 - SMI for SNMPV2 - Protocol - SNMPv3 - Architecture and applications -Security and access control model - Overview of RMON.

UNIT VI

Protocols, Sessions, State, and Implementing Custom Protocols State vs. Stateless, Methods for Maintaining State, What Is a Protocol? Designing a Custom Protocol, Our Chat Protocol, Protocol Registration

Elementary Name, Address Conversions and design decisions Domain Name System, gethostbyname Function, RES_USE_INET6 Resolver Option, gethostbyname2 Function and IPv6 Support, gethostbyaddr Function, uname Function, gethostname Function, getservbyname and getservbyport Functions

Text Books:

1. W. Richard Stevens, ***"UNIX Network Programming Vol-I"***, Addison-Wesley Professional, 3rd Edition,2003.
2. William Stallings, ***"SNMP, SNMPv2, SNMPv3 and RMON 1 and 2"***, Pearson Edition, 3rd Edition, 2009.

Reference Book:

1. D.E. Comer, ***"Internetworking with TCP/IP Vol- III: Client-Server Programming and Application BSD Sockets Version"***, Pearson Edition, 2nd Edition, 2003.

Course Title:	**Advanced Database Technology**	**Semester VI**	
Course Code	**BTITPE606E**	**Course Type**	**Elective**
Pre-requisite	**Database Management Systems**	**L – T – P**	**3 – 0 – 0**
Stream	**Data Science**	**Credits**	**3**

Course Objectives:

1. To learn the various types of databases and their advanced applications.
2. To understand how and where databases are used in industry.
3. To examine the requirements on special databases.
4. To learn complex queries and interface them with applications.

Course Outcomes:

After learning the course, the students should be able:

1. To explain how databases are used in various fields of industry.
2. To apply query evaluation techniques and query optimization techniques.
3. To develop transaction processing systems with concurrency control.
4. To design and develop a database application system as part of a team.
5. To explore open issues in advanced databases.

Course Content:

UNIT I

Parallel And Distributed Databases: Database System Architectures: Centralized and Client-Server Architectures – Server System, Architectures – Parallel Systems- Distributed Systems – Parallel Databases: I/O Parallelism – Inter and Intra Query Parallelism – Inter and Intra operation Parallelism – Distributed Database Concepts - Distributed Data Storage – Distributed Transactions – Commit Protocols – Concurrency Control – Distributed Query Processing – Three Tier Client Server Architecture- Case Studies.

UNIT II

Object and Object Relational Databases: Concepts for Object Databases: Object Identity – Object structure – Type Constructors – Encapsulation of Operations – Methods – Persistence – Type and Class Hierarchies – Inheritance – Complex Objects – Object Database Standards, Languages and Design: ODMG Model – ODL – OQL – Object Relational and Extended – Relational Systems: Object Relational features in SQL /Oracle – Case Studies.

UNIT III

Xml Databases: XML Databases: XML Data Model – DTD - XML Schema - XML Querying – Web Databases – JDBC– Information Retrieval – Data Warehousing – Data Mining.

UNIT IV

Mobile Databases: Mobile Databases: Location and Handoff Management - Effect of Mobility on Data Management - Location Dependent Data Distribution - Mobile Transaction Models - Concurrency Control - Transaction Commit Protocols- Mobile Database Recovery Schemes.

UNIT V

Intelligent Databases: Active databases – Deductive Databases – Knowledge bases – Multimedia Databases- Multidimensional Data Structures – Image Databases – Text/Document Databases- Video Databases – Audio Databases – Multimedia Database Design.

UNIT VI

Complex Queries and Reasoning: Logic of Query Languages – Relational Calculi – Recursive rules – Syntax and semantics of Datalog – Fix-point semantics – Implementation Rules and Recursion – Rule rewriting methods – Compilation and Optimization – Recursive Queries in SQL – Open issues.

Text Books

1. Carlo Zaniolo, Stefano Ceri, ***"Advanced Database Systems"***, Morgan Kauffmann Publishers.
2. Subramaniam, ***"Multimedia Databases"***, Morgan Kauffman Publishers, 2008.
3. Rajesh Narang, ***"Object Oriented Interfaces and Databases"***, Prentice-Hall of India, Pvt. Ltd., 2004.
4. Thomas Cannolly and Carolyn Begg, ***"Database Systems, A Practical Approach to Design, Implementation and Management"***, Pearson Education, 3rd Edition, 2007.
5. Jeffrey A. Hoffer, Mary B. Prescottand Fred R. McFadden, ***"Modern Database Management"***, Prentice Hall, 2007.

Reference Books:

1. Henry F Korth, Abraham Silberschatz and S. Sudharshan, ***"Database System Concepts"***, McGraw Hill, 6th Edition, 2011.
2. C. J. Date, A. Kannan and S. Swamynathan, ***"An Introduction to Database Systems"***, Pearson Education, 8th Edition,2006.
3. R. Elmasri, S. B. Navathe, ***"Fundamentals of Database Systems"***, Pearson Education/Addison Wesley, 5th Edition, 2007.
4. Ramakrishnan, Gehrke, ***"Database Management System"***, Tata McGraw Hill Publications, 4th Edition.
5. Ramez Elmasri, Sham Navathe, ***"Fundamentals of Database Systems"***, Addison-Wesley, 2000.

Course Title:	**Operating Systems Lab**	**Semester VI**	
Course Code	**BTITL607**	**Course Type**	**Mandatory**
Pre-requisite	**Nil**	**L – T – P**	**0 – 0 – 2**
Stream	**Core**	**Credits**	**1**

Lab Experiments Objective:

1. To learn shell programming and the use of filters in the UNIX environment.
2. To learn to programming in C using system calls.
3. To learn to use the file system related system calls.
4. To process creation and inter process communication.
5. To familiarize with implementation of CPU Scheduling Algorithms, page replacement algorithms and Deadlock avoidance.

Lab Experiments List:

1. Basics of UNIX commands.
2. Shell Programming.
3. Implement the following CPU scheduling algorithms:
   - Round Robin
   - SJF
   - FCFS
   - Priority
4. Implement all file allocation strategies:
   - Sequential
   - Indexed
   - Linked
5. Implement Semaphores.
6. Implement all File Organization Techniques:
   - Single level directory
   - Two level
   - Hierarchical
   - DAG
7. Implement Bankers Algorithm for Dead Lock Avoidance.
8. Implement an Algorithm for Dead Lock Detection.
9. Implement e all page replacement algorithms:
   - FIFO
   - LRU
   - LFU
10. Implement Shared memory and IPC.
11. Implement Paging Technique of memory management.
12. Implement Threading & Synchronization Applications.

Course Title:	**Digital Image Processing Lab**	**Semester VI**	
Course Code	**BTITL608**	**Course Type**	**Mandatory**
Pre-requisite	**OOP with C++**	**L – T – P**	**0 – 0 – 2**
Stream	**Core**	**Credits**	**1**

Lab Experiments List:

Study of "Matlab/Scilab or any other open source Image Processing Toolbox" and experiments as per the syllabus, to be decided by the concerned faculty.

Course Title:	**Software Testing Lab**	**Semester VI**	
Course Code	**BTITPEL609A**	**Course Type**	**Elective**
Pre-requisite	**Nil**	**L – T – P**	**0 – 0 – 2**
Stream	**Software Application & Development**	**Credits**	**1**

Lab Experiments Objective:

1. To implement different testing techniques to practical teat and understand their merits and demerits.
2. To understand and study one of the software testing tool like QTP, WINRUNNER.

Lab Experiments List:

1. Design, develop, code and run the program in any suitable language to solve the commission problem. Analyze it from the perspective of data flow testing, derive at least 10 different test cases, execute these test cases and discuss the test results.
2. Design, develop, code and run the program in any suitable language to solve the NextDate problem. Analyze it from the perspective of decision table-based testing, derive at least 10 different test cases, execute these test cases and discuss the test results.
3. Design, develop, code and run the program in any suitable object-oriented language to solve the calendar problem. Analyze it from the perspective of OO testing, derive test cases to test the method that increment the date and the method that increments the month., execute these test cases and discuss the test results.
4. Design, develop, code and run the program in any suitable object-oriented language to solve the currency converter problem. Analyze it from the perspective of use case-based system testing, derive appropriate system test cases, execute these test cases and discuss the test results.
5. Design, develop, code and run the program in any suitable language to implement an absolute letter grading procedure, making suitable assumptions. Determine the basis paths and using them derive different test cases, execute these test cases and discuss the test results.
6. Design, develop, code and run the program in any suitable language to implement the binary search algorithm. Determine the basis paths and using them derive different test cases, execute these test cases and discuss the test results.

Course Title:	**Data Storage Technologies & Networks Lab**	**Semester VI**	
Course Code	**BTITPEL609B**	**Course Type**	**Elective**
Pre-requisite	**Computer Networks, Operating Systems**	**L – T – P**	**0 – 0 – 2**
Stream	**Infrastructure & Security Management**	**Credits**	**1**

Lab Experiments Objective:

1. Understand the functionalities of storage network administration.
2. Set up a NAS server to support file level data access via the NSF and the CIFS protocols.
3. Set up a SAN server to support the iSCSI protocol for block level data access.
4. Demonstrate ability to design and build a small-scale data center and a small-scale cloud computing environment.
5. Be hand-on with data and network management software.

Lab Experiments List:

1. Install a hard disk on a Linux machine covering all the below activities:
   a) Connecting the disk to an HBA (Host Bus Adapter) and BIOS setup for the disk;
   b) Partitioning the disk;
   c) Creating file systems within disk partitions;
   d) Mounting the files systems;
   e) Setting up automatic mounting;
   f) Labeling disk partitions;
   g) Setting up swapping on swap partitions.
2. Use “smartmontools” to monitor the disk performance monitoring and testing:
   a) Use “smartctl” to enable S.M.A.R.T. support and offline data collection on the disk;
   b) Check the overall health of the disk;
   c) Run a self-test on the disk;
   d) Set up “smartd” to do tests automatically.
3. Use “hdparm”, “iostat”, and “iometer” tools to measure the performance of different storage devices, such as SATA drive, SCSI drive, and USB drives.
   a) Plot graphs to compare read/write and sequential/random access rates among different storage devices.
4. Use Navisphere Manager Simulator to perform management on SAN disk array systems:
   a) Configure storage pools and LUNs (Logical Unit Number) for storage groups;
   b) Configure snapshots and clones;
   c) Create SANCopy full and incremental sessions;
   d) Create MirrorView synchronous and asynchronous images;
   e) Expand a LUN to create metaLUNs;
   f) Migrate a LUN to another LUN.
5. Use Openfiler for network storage configuration management:
   a) Configure the Openfiler to support locally attached USB drives;
   b) Set up a NAS server to support NSF and CIFS protocols;
   c) Set up a SAN server to support an iSCSI protocol.
6. Configure Openfiler as a NAS Server:
   a) Configure access control rules and NFS/CIFS shares for the NAS server;
   b) Configure the Linux client machine to access the NFS shares on the NAS server;
   c) Configure a Windows VM on the Linux client machine to access the CIFS shares on the NAS server;
   d) Use Openfiler to set up a SAN server, to supports iSCSI protocol for the block level data access;

e) Configure access control rules for the SAN server and configure iSCSI targets on the server.

7.

a) Use VMware to create virtual disks, Virtual Machine File Systems and provisioning.
b) Use thin and thick provisioning concepts.

Course Title:	**Service Oriented Architecture Lab**	**Semester VI**	
Course Code	**BTITPEL609C**	**Course Type**	**Elective**
Pre-requisite	**Programming in Java**	**L – T – P**	**0 – 0 – 2**
Stream	**Information Management & Quality Control**	**Credits**	**1**

Lab Experiments Objective:

1. To learn to create web services and web service clients.
2. To learn SOAP, UDDI and WSDL platforms.

Lab Experiments List:

1. Write a simple web application program in Java to create web services incorporating:
   - **a)** Development of web service.
   - **b)** Testing the web service.
   - **c)** Developing the client.
   - **d)** Deploying the application.
2. Write a factorial application program in Java to create web services.
3. Implement a Calculator program and calculate Simple and Compound Interest using .Net.
4. Develop an invoice order processing system.
5. Invoke EJB components as Web Service.

Course Title:	**Network Programming Lab**	**Semester VI**	
Course Code	**BTITPEL609D**	**Course Type**	**Elective**
Pre-requisite	**Programming in Java**	**L – T – P**	**0 – 0 – 2**
Stream	**Network**	**Credits**	**1**

Lab Experiments Objective:

1. To develop TCP Socket Programming, UDP applications and to implement File Transfer Protocols.
2. To utilize RMI and Routing Algorithms.

Lab Experiments List:

1. Write a socket Program for Echo/Ping/Talk commands.
2. Create a socket (TCP) between two computers and enable file transfer between them.
3. Create a socket (UDP) between two computers and enable file transfer between them.
4. Write a program to implement Remote Command Execution. (Two M/Cs may be used)
5. Write a code simulating ARP /RARP protocols.

1. Create a socket for HTTP for web page upload and download.
2. Write a program for TCP module implementation. (TCP services)
3. Write a program for File Transfer in client-server architecture using following methods.
(a) RS232C (b) TCP/IP
4. Write a program to implement RMI (Remote Method Invocation)
5. Perform a case study about the different routing algorithms to select the network path with its optimum and economical during data transfer.
   a. Shortest path routing
   b. Flooding
   c. Distance vector
6. Implement client in C and server in Java and initiate communication between them.
7. Using OPNET
   a. Create a scenario with the following specifications.
      i. No of subnets – 2
      ii. No. of nodes – 40
      iii. Traffic
         1. FTP - 11 to 21
         2. FTP - 30 to 40
         3. UDP - 5 to 7
      iv. Routing Protocol – AODV
      v. 802.16, Show the throughput using different bandwidths i.e., 10 Mbps and 100 Mbps respectively.
   b. Create a scenario as described below.
      No of students – 2
      SN -1 Nodes – 15
      SN -2 Nodes - 10
      Generate FTP Traffic & HTTP traffic between Nodes 1 to 11 (FTP)
      14 to 7 (HTTP / Gen FTP)
8. Trace the packet within the Simulation time and display the Trace file.

Course Title:	**Advanced Database Technology Lab**	**Semester VI**	
Course Code	**BTITPEL609E**	**Course Type**	**Elective**
Pre-requisite	**Database Management Systems**	**L – T – P**	**0 – 0 – 2**
Stream	**Data Science**	**Credits**	**1**

Lab Experiments Objective:

1. To learn the various types of databases and their advanced applications.
2. To understand how and where databases are used in industry.
3. To examine the requirements on special databases.
4. To learn complex queries and interface them with applications.

Lab Experiments List:

1. A University wants to track persons associated with them. A person can be an Employee or Student. Employees are Faculty, Technicians and Project associates. Students are Full time students, Part time students and Teaching Assistants.
   a. Design an Enhanced Entity Relationship (EER) Model for university database. Write OQL for the following
      1. Insert details in each object.
      2. Display the Employee details.
      3. Display Student Details.
      4. Modify person details.
      5. Delete person details.

   b. Extend the design by incorporating the following information.
      Students are registering for courses which are handled by instructor researchers (graduate students). Faculties are advisors to graduate students. Instructor researchers' class is a category with super class of faculty and graduate students. Faculty is having sponsored research projects with a grant supporting instruction researchers. Grants are sanctioned by different agencies. Faculty belongs to different departments. Department is chaired by a faculty. Implement for the Insertion and Display of details in each class.
2. Consider the application for University Counseling for Engineering Colleges. The college, department and vacancy details are maintained in 3 sites. Students are allocated colleges in these 3 sites simultaneously. Implement this application using parallel database [State any assumptions you have made].
3. There are 5 processors working in a parallel environment and producing output. The output record contains college details and students mark information. Implement parallel join and parallel sort algorithms to get the marks from different colleges of the university and publish 10 ranks for each discipline.
4. Create triggers and assertions for Bank database handling deposits and loan and admission database handling seat allocation and vacancy position. Design the above relational database schema and implement the following triggers and assertions.
   a. When a deposit is made by a customer, create a trigger for updating customers account and bank account
   b. When a loan is issued to the customer, create a trigger for updating customer's loan account and bank account.
   c. Create assertion for bank database so that the total loan amount does not exceed the total balance in the bank.
   d. When an admission is made, create a trigger for updating the seat allocation details and vacancy position.

5. Construct a knowledge database for kinship domain (family relations) with facts. Extract the following relations using rules.

   Parent, Sibling, Brother, Sister, Child, Daughter, Son, Spouse, Wife, husband, Grandparent, Grandchild, Cousin, Aunt and Uncle.

6. Work with Weka tool classification and clustering algorithms using the given training data and test with the unknown sample. Also experiment with different scenarios and large data set

7. Design XML Schema for the given company database, Department (deptName, deptNo, deptManagerSSN, deptManagerStartDate, deptLocation), Employee (empName, empSSN, empSex, empSalary, empBirthDate, empDeptNo, empSupervisorSSN, empAddress, empWorksOn), Project (projName, projNo, projLocation, projDeptNo, projWorker)

   a. Implement the following queries using XQuery and XPath

   i. Retrieve the department name, manager name, and manager salary for every department'

   ii. Retrieve the employee name, supervisor name and employee salary for each employee who works in the Research Department.

   iii. Retrieve the project name, controlling department name, number of employees and total hours worked per week on the project for each project.

   iv. Retrieve the project name, controlling department name, number of employees and total hours worked per week on the project for each project with more than one employee working on it.

   b. Implement a storage structure for storing XML database and test with the above schema.

Course Title:	**Mini Project**	**Semester VI**	
Course Code	**BTITP610**	**Course Type**	**Elective**
Pre-requisite		**L – T – P**	**0 – 0 – 2**
Stream		**Credits**	**1**

Evaluation Criteria

The total term work shall be of 50 marks. The 30 marks shall be distributed over internal assessments / reviews (at least 02 reviews) during the semester by a review committee. The remaining 5 marks shall be distributed for attendance. The Head of the Department shall constitute the review committee. The student shall make a presentation on the progress made before the committee. The 20marks of the practical will be awarded based on the performance in the practical exam conducted by the University at the end of the semester.

General Suggestions and Expectations / Guidelines

• The project shall be developed in C++/JAVA/PYTHON

• The students may choose the theory concepts they studied in different subjects as project topic.

• Interdisciplinary project proposals and innovative projects are encouraged and more appreciable.

• The project topic can be suggested by the staff member or it can be proposed by the students.

• The project topic shall be approved by the project in-charge.

• The Guides are advised to give projects and suggest project titles focusing more on the current field of research and ensure the level of innovation.

• A project team shall contain a maximum of 2 members.

• The project work should be properly distributed among the team members.

• Students should submit the project documentation at the beginning of the semester consisting of:

Title, Abstract

Modules Split-up

Deliverables for each review

Data Model (If Any)

Details of Team Members

• Reviews for the project work will be conducted at regular intervals by the panel of examiners formed by the Head of the Department.

• The student failing to attend the project review will be subject to strict action as decided by the Head of the Department.

• Throughout the semester at any point of time if students are found to be involved in any of the following:

Using project codes available on the Internet

Using project codes developed by someone else

Using project work which is already submitted in other institute or university Such students shall be declared failed or penalized as decided by the Examiners.

• The students must arrange regular meetings with the guide and present progress of project work.

• A Spiral bound Project report to be prepared as per the guidelines and format given by the Department

• The guides are advised to check for the formatting of the presentation and project report.

• Students must submit a report well before the end of the semester.

Course Title:	**Field Training/ Internship/ Industrial Training**	**Semester VI**	
Course Code	**BTITF611**	**Course Type**	**Elective**
Pre-requisite		**L – T – P**	**0 – 0 – 0**
Stream		**Credits**	**0**

Minimum 4 Weeks which can be completed partially in 5th and 6th semester or at one time.