SYLLABUS STRUCTURE
(Effective from academic session 2023-24)

FOR THE DEGREE

OF

Master of Computer Applications
(MCA)

Four-Semester Full Time
Programme

SCHOOL OF BASIC SCIENCES

MANIPAL UNIVERSITY
JAIPUR
INSPIRED BY LIFE

ELIGIBILITY OF THE CANDIDATES:

Passed B.C.A/ B.Sc. (Computer Science)/ B.Sc. (IT) / B.E. (CSE)/ B.Tech. (CSE) / B.E. (IT) / B.Tech. (IT) or equivalent Degree.
OR
Passed any graduation degree (e.g.: B.E. / B.Tech. / B. Sc / B. Com. / B. A./ B. Voc./ etc.) preferably with Mathematics at 10+2 level or at Graduation level Obtained at least 50% marks (45% marks in case of candidates belonging to reserved category) in the qualifying examination.
Note: For students having no Mathematics background compulsory bridge course will be framed by the respective University/ Institution and additional bridge courses related to computer subjects as per the norms of the concerned University).

Program specific outcomes for Master of Computer applications (MCA) program:

[PSO.1.] To work productively as IT professional both at supportive and leadership roles.

[PSO.2.] To advance successfully in their chosen career path utilizing technical abilities, leadership qualities, communication and interpersonal skills with high regard to legal and ethical responsibilities.

[PSO.3.] To build their profession adaptable to the changes in the technology with lifelong learning.

YEAR	FIRST SEMESTER						SECOND SEMESTER					
	Course Code	Subject Name	L	T	P	C	Course Code	Subject Name	L	T	P	C
I	MA6105	Discrete Mathematical Structures with Graph Theory	3	1	0	4	CA6203	Computer Networks & Protocols	3	1	0	4
	CA6101	Web Technologies	3	1	0	4	CA6204	Object Oriented Programming using Java	3	1	0	4
	CA6105	Programming & Problem-Solving using C	3	1	0	4	CA6205	Operating System	3	1	0	4
	CA6106	Relational Database Management Systems	3	1	0	4	CA6206	Data Structures and Algorithms	3	1	0	4
	CA6108	Data Visualization	3	1	0	4	CA62XX	Program Elective-I	3	0	0	3
	CA6130	Web Technology Lab	0	0	2	1	CA6233	Object Oriented Programming using Java Lab	0	0	2	1
	CA6133	Programming & Problem-Solving using C Lab	0	0	2	1	CA6234	Data Structures and Algorithms Lab	0	0	2	1
	CA6134	Relational Database Management Systems Lab	0	0	2	1	CA6235	Pragmatic Learning	0	0	2	1
							CA6210	Aptitude & Technical Development	1	1	0	2
			15	5	6	23			16	5	6	24
	Total Contact Hours (L+T+P)		26				Total Contact Hours (L+T+P) + OE		27			

YEAR	THIRD SEMESTER						FOURTH SEMESTER					
	Course Code	Subject Name	L	T	P	C	Course Code	Subject Name	L	T	P	C
II	CA7104	Unix & Shell Programming	3	1	0	4	CA7270	Major Project	0	0	0	16
	CA7105	Android Application Development	3	1	0	4						
	CA7106	Software Engineering & Project Management	3	1	0	4						
	CA7152-54	Program Elective-II	3	0	0	3						
	CA7155-57	Program Elective-III	3	0	0	3						
	CA7131	Minor Project	0	0	4	2						
	CA7132	Unix & Shell Programming Lab	0	0	2	1						
	CA7133	Android Application Development Lab	0	0	2	1						

		15	3	8	22		0	0	0	16
	Total Contact Hours (L+T+P)		26			Total Contact Hours (L+T+P)		00		
Total Credit= 85										

Program Electives – I

CA6245 CRYPTOGRAPHY AND NETWORK SECURITY

CA6246 ARTIFICIAL INTELLIGENCE

CA6247 INTERNET OF THINGS

Program Electives – II

CA7152 DATA MINING TECHNIQUES

CA7153 CLOUD COMPUTING & INFRASTRUCTURE SERVICES

CA7154 ADVANCED COMPUTER NETWORKS

Program Electives – III

CA7155 ADVANCES IN MACHINE LEARNING

CA7156 BLOCKCHAIN TECHNOLOGIES

CA7157 DATA SCIENCE

FIRST SEMESTER

XXXXXX: BASIC MATHEMATICS FOR COMPUTER APPLICATIONS (BRIDGE COURSE) [0 0 0 0]

Set Theory: sets, subsets, set operation, Cartesian product, relation (properties, equivalence relation, and partition) and Function: Definition, domain and range of function, types of functions (into, onto, one to one), composite function. Special Functions: Trigonometric functions and their properties, exponential functions, logarithmic functions, hyperbolic functions, inverse circular functions and related properties, simple problems. Rational functions, partial fraction and simple problems. Principal of inclusion and exclusions (statement only and simple problems), Generating Functions recurrence relation.

References:

1. Shanti Narayan, "Differential calculus", S. Chand & Co, Delhi, 2012.
2. M.D. Raisinghania, et.al, "Differential calculus", Delhi, 2010.
3. N. Piskunov, "Differential and integral calculus", Vol I & Vol II, CBS, 2000.

XXXXXX: DISCRETE MATHEMATICAL STRUCTURES WITH GRAPH THEORY [3 1 0 4]

POSET, Lattices, distributive, and complemented lattices, Boolean Lattice, Uniqueness of Boolean Lattices Boolean expression & function. Mathematical Logic: Statement and notations, connectives, normal forms, well-formed formulas, implication, Tautology, Predicate calculus. Graphs: Introduction, Isomorphism, Sub graphs, Walks, Paths, Circuits, Connectedness, Components, Euler graphs, Hamiltonian paths and circuits, Trees, Properties of trees, Distance and canters in tree, Rooted and binary trees. Trees, Connectivity & Planarity Spanning trees, Fundamental circuits, spanning trees in a weighted graph, cut sets, Properties of cut set, all cut sets, Fundamental circuits and cut sets, Connectivity and separability, Network flows: Isomorphism, Combinational and geometric graphs, Planer graphs, Different representation of a planer graph. Matrices, Coloring and Directed Graph. Chromatic number, Chromatic partitioning, Chromatic polynomial, Matching, Covering, Four color problem, Directed graphs, Types of directed graphs, Digraphs and binary relations, Directed paths and connectedness, Euler graphs.

References:

1. C.L. Lui, Elements of Discrete Mathematics, (4e) Houghton Mifflin, 2017
2. N. Deo, Graph Theory: With Application to Engineering and Computer Science, (New Edition) Prentice Hall of India, 2003.
3. R.P. Grimald Discrete and Combinatorial Mathematics: An Applied Introduction, (5e) Addison Wesley, 2003.

CA6101: WEB TECHNOLOGIES [3 1 0 4]

Introduction : Concept of WWW, Internet and WWW, HTTP Protocol : Request and response, Web browser and Web servers, Features of Web 2.0, Web Design: Concepts of effective web design, Web design issues including Browser, Bandwidth and Cache, Display resolution, Look and Feel of the Website, Page Layout and linking, User centric design, Sitemap, Planning and publishing website, Designing effective navigation, JavaScript : Client side scripting with JavaScript, variables, functions, conditions, loops and repetition, Pop up boxes, Advance JavaScript: Javascript and objects, JavaScript own objects, the DOM and web browser environments, Manipulation using DOM, forms and validations, DHTML : Combining HTML, CSS and Javascript, Events and buttons, XML : Introduction to XML, uses of XML, simple XML, XML key components, DTD and Schemas, Using XML with application. Transforming XML using XSL and XSLT, PHP : Introduction and basic syntax of PHP, decision and looping with examples, PHP and HTML, Arrays, Functions, Browser control and detection, string, Form processing, Files, Advance Features: Cookies and Sessions, Object Oriented Programming with PHP, PHP and MySQL : Basic commands with PHP examples, Connection to server, creating database, selecting a database, listing database, listing table names, creating a table, inserting data, altering tables, queries, deleting database, deleting data and tables, PHP myadmin and database bugs. Introduction to Angular JS4.

References:

1. R. Moseley & M. T. Savaliya, Developing Web Applications, (2e) Wiley-India, 2013.
2. J. Sklar, Web Design Principles, (5e) Cengage Learning, 2012.
3. Harwani, Developing Web Applications in PHP and AJAX, (1e) McGrawHill, 2010.
4. P.J. Deitel & H.M. Deitel, Internet and World Wide Web How to program, (4e) Pearson, 2007.

CA6105: PROGRAMMING & PROBLEM-SOLVING USING C [3 1 0 4]

An overview: Algorithms & flowcharts; Characteristics of a good program. Rules/ conventions of coding, documentation, naming variables; Top-down design; Bottom-up design. Fundamentals of C Programming: History of C; Structure of a C Program; Data types; Constant & Variable, naming variables; Operators & expressions; Control Constructs – if-else, for, while, do-while; Case switch statement; Arrays; Formatted & unformatted I/O; Type modifiers & storage classes; Ternary operator; Type conversion & type casting; Priority & associativity of operators. Modular Programming: Functions; Arguments; Return value; Parameter passing – call by value, call by reference; Return statement; Scope, visibility and life-time rules for various types of variables, static variable; Calling a function; Recursion – basics, comparison with iteration, types of recursions, when to avoid recursion, examples. Advanced Programming Techniques: Special constructs – Break, continue, exit(), goto & labels; Pointers - & and * operators, pointer expression, pointer arithmetic, dynamic memory management functions like malloc(), calloc(), free(); String; Pointer v/s array; Pointer to pointer; Array of pointer & its limitation; Function returning pointers; Pointer to function, Function as parameter; Structure – basic, declaration, membership operator, pointer to structure. Introduction to Data Structures: Contiguous implementations of stack& queues, various operations on stack& queues.

References:
1. Kerninghan & Ritchie "The C programming language", PHI.
2. Schildt "C: The Complete reference" 4th ed TMH.
3. Cooper Mullish "The Spirit of C", Jaico Publishing House, Delhi.
4. Kanetkar Y. "Let us C", BPB.
5. TennenBaum A.M. & others: Data Structures using C & C++; PHI.

CA6106: RELATIONAL DATABASE MANAGEMENT DATABASE MANAGEMENT SYSTEMS [3 1 0 4]

Introduction: Database-System Applications, Relational Databases, Database Design, Data Storage and Querying, Transaction Management, Database Architecture; Relational Algebra: Fundamental Relational- Algebra Operations, Extended Relational-Algebra Operations, Null Values, Modification of the Database; SQL: Data Definition Language, Data manipulation language , SQL Data Types and Schemas, Integrity Constraints, Basic Structure of SQL Queries, Set Operations, Aggregate Functions, Null Values, Nested Sub-queries, Complex Queries, Views, Modification of the Database, Joined Relations, Authorization, Overview of the Design Process; The Entity-Relationship Model: Constraints, Entity-Relationship Diagrams, Entity-Relationship Design Issues, Weak Entity Sets, Extended E-R Features; Normalization: Anomalies, Referential integrity, 1NF, Functional Dependency, 2NF, 3NF, BCNF; Hashing Techniques: Dynamic Hashing; Transactions: Transaction State, Implementation of Atomicity and Durability, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, Testing for Serializability, Lock-Based Protocols, Log-Based Recovery, Recovery algorithms.

References:
1. S. Korth, Database System Concept, Mc-GrawHill, (6e), 2011.
2. R. Elmasri and S. Navathe, Fundamentals of Database Systems, (6e) Pearson Education, 2006.
3. T. Connolly, C. Begg, Database Systems–A Practical Approach to Design, Implementation and Management, (3e) Pearson Education, 2002.

CA6108: DATA VISUALIZATION [3 1 0 4]

Introduction to data visualization, importance of data visualization, data types, different tools for data visualization, Pie Chart, Bar Chart, Histogram, Gantt Chart, Heat Map, Box and Whisker Plot, Waterfall Chart, Area Chart, Scatter Plot, Pictogram Chart, Timeline, Highlight Table, Bullet Graph, Choropleth Map, Word Cloud, Network Diagram, Correlation Matrices, geographical plots, Density Maps, Bubble Chart, Tree maps. Text Table, Applying Filters, Time Series and trend analysis, working with dates, 3-D Scatter Plots, 3-D Mash, Dashboard development process, dashboard architecture, story development.

References:
1. Engebretsen, Martin, and Helen Kennedy, Data visualization in society, 2020.
2. Anouncia, S. Margret, Hardik A. Gohel, and Subbiah Vairamuthu, Data Visualization, Springer Verlag, Singapore, 2020.

CA6130: WEB TECHNOLOGY LAB [0 0 2 1]

Implement forms using HTML, Frames and CSS. Use of XML Tags, PHP : Use of basic syntax of PHP, decision and looping with examples, PHP and HTML, Arrays, Functions, Browser control and detection, string, Form processing, Files, Advance Features: Cookies and Sessions, Object Oriented Programming with PHP, PHP and MySQL : Basic commands with PHP examples, Connection to server, creating database, selecting a database, listing database, listing table names, creating a table, inserting data, altering tables, queries, deleting database, deleting data and tables, PHP myadmin and database bugs.

References:
1. R. Moseley & M. T. Savaliya, Developing Web Applications, (2e) Wiley-India, 2013.
2. Team at Kogent Learning Solutions Inc., Web Technologies, Black Book, (1e) Dreamtech Press, 2009.
3. Team at Kogent Learning Solutions Inc., HTML 5, Black Book, (2e) Dreamtech Press, 2011.
4. J. Sklar, Web Design Principles, (5e) Cengage Learning, 2012.
5. Harwani, Developing Web Applications in PHP and AJAX, (1e) McGrawHill, 2010.
6. P.J. Deitel& H.M. Deitel, Internet and World Wide Web How to program, (4e) Pearson, 2007.

CA6133: PROGRAMMING & PROBLEM-SOLVING USING C LAB [0 0 2 1]

Simple C Programs (expression-oriented operations); Programs to illustrate various operators in C. Programs using branching constructs (if, if-else-if, switch-case); Programs using looping constructs (for, while, do-while, continue, break); Programs on 1D Arrays; Programs on 2D Arrays; Programs on strings; Programs using functions (with and without recursion), passing parameters by value and reference. Operations on Stacks: Push, Pop, Queues.

References:
1. Kerninghan & Ritchie "The C programming language", PHI.
2. Schildt "C:The Complete reference" 4th ed TMH.
3. Cooper Mullish "The Spirit of C", Jaico Publishing House, Delhi.
4. Kanetkar Y. "Let us C", BPB.
5. Tennen Baum A.M. & others: Data Structures using C & C++; PHI.

CA6134: RELATIONAL DATABASE MANAGEMENT SYSTEMS LAB [0 0 2 1]

Experiments on DDL and Basic SQL, Advanced SQL, ER diagrams using DIA tool, Data Integrity Constraints and Built-in Functions, Design and Implementing the data requirements of a simple DB application, Experiments on Basic PL/SQL, PL/SQL Exceptions and Transactions, PL/SQL Cursors, PL/SQL Procedures, Functions and Packages, Project (DB application development with Java as front end).

References:
1. S. Korth, Database System Concepts, (6e) Mc-GrawHill, 2011.
2. R. Elmasri and S. Navathe, Fundamentals of Database Systems, (6e) Pearson Education, 2009.

SECOND SEMESTER

CA6203: COMPUTER NETWORKS AND PROTOCOLS [3 1 0 4]

Network introduction: Classful addressing, other issues, Subnetting Classless addressing, variable length blocks, Subnetting, address allocation, Network Address Translation. Encapsulation, operation Data Link Layer: ARP package & RARP- Introduction, packet format Encapsulation, RARP server datagram , fragmentation , options, checksum, Network Layer: IP Package Types of messages, message format, error reporting, Query, Checksum, Debugging tools; Transport Layer: Process to process communication, User datagram, checksum, UDP operation UDP package Introduction, TCP services, TCP features, segment, TCP connection, State transition diagram, Flow control, Error control, Congestion control, TCP timers, options, TCP package; TCP Variants: SCTP services, SCTP features, packet format, association, state transition diagram, flow control, error control, congestion control, TCP RENO, Dynamic routing protocols : RIP,OSCF & BGP; Domain name Space (Application Layer): Name space, distribution of name space, DNS in the internet, resolution, DNS messages, controlling the server, out of band signaling, escape character. Transition from IPv4 to IPv6. Introduction to VLAN concept, Wireless Network protocols: WAP Architecture introduction. Introduction to MANET & VANET

References:
1. W. R Stevens, TCP/IP Illustrated, Volume 1: The Protocols, (2e) Addison-Wesley, 1994.
2. P. Loshin, IPV6 Clearly Explained, (4e) Morgan Kauffman, 2003.
3. B. A. Forouzan, TCP/IP Protocol Suite, (2e) TMH, 2005.

CA6204: OBJECT ORIENTED PROGRAMMING USING JAVA [3 1 0 4]

Introduction to Object Oriented Programming. Overview of Core Java, AWT: Components and class hierarchy, Event Handling, Layouts Manager, Container. Swing: Concepts of Swing, Java Foundation Class (JFC), Swing Packages and Classes, Working with Swing, Swing Components. Overview of JavaFX. JSP and Servlets: Java Server Pages (JSP) – Introduction, what is needed to write JSP based web application? How does JSP look? How to test a JSP? Servlets – Introduction, History of Web Application, Web Architecture, Servlet Life Cycle. JDBC: Database Management; Mechanism for connecting to a back-end database; Loading the ODBC driver. Networking: Networking in Java; URL Objects.

Reference:
1. H. Schildt, The Complete Reference Java Eight Edition, (8e) Tata McGraw-Hill, reprint 2011.
2. S. Holzner, Java 2 programming black book, (5e) Dream Tech, New Delhi, reprint: 2005.

CA6205: OPERATING SYSTEM [3 1 0 4]

Introduction to System Programs & Operating Systems, Evolution of Operating System (mainframe, desktop, multiprocessor, Distributed, Network Operating System, Clustered & Handheld System), Operating system services, Operating system structure, System Call & System Boots, Operating system design & Implementations, System protection, Buffering & Spooling. Types of Operating System: Bare machine, Batch Processing, Real Time, Multitasking & Multiprogramming, time-sharing system. File: concepts, access methods, free space managements, allocation methods, directory systems, protection, organization, sharing & implementation issues, Disk & Drum Scheduling, I/0 devices organization, I/0 devices organization, I/0 buffering, I/O Hardware, Kernel I/O subsystem, Transforming I/O request to hardware operations. Device Driver: Path managements, Sub module, Procedure, Scheduler, Handler, Interrupt Service Routine. File system in Linux & Windows Process: Concept, Process Control Blocks (PCB), Scheduling criteria Preemptive & non-Preemptive process scheduling, Scheduling algorithms, algorithm evaluation, multiple processor scheduling, real time scheduling, operations on processes, threads, inter process communication, precedence graphs, critical section problem, semaphores, classical problems of synchronization. Deadlock: Characterization, Methods for deadlock handling, deadlock prevention, deadlock avoidance, deadlock detection, recovery from deadlock, Process Management in Linux. Memory Hierarchy, Concepts of memory management, MFT & MVT, logical and physical address space, swapping, contiguous and non-contiguous allocation, paging, segmentation, and paging combined with segmentation. Structure & implementation of page table. Concepts of virtual memory, Cache Memory Organization, demand paging, page replacement algorithms, allocation of frames, thrashing, demand segmentation. Unit V Distributed operating system: Types, Design issues, File system, Remote file access, RPC, RMI, Distributed Shared Memory (DSM), Basic Concept of Parallel Processing &Concurrent Programming. Case study of Unix, Linux & Windows.

References:
1. Silberschatz," Operating system", Willey Pub.
2. Stuart," Operating System Principles, Design & Applications", Cengage Learning.
3. Tannanbaum, "Modern operating system", PHI Learning.
4. Dhamdhere," Operating System", TMH.
6. William stalling, "operating system" Pearson Edu.
7. Deitel & Deitel, "Operating Systems", Pearson Edu.
8. Flynn & Mchoes, "Operating Systems", Cengage Learning.
9. Haldar, "Operating System", Pearson Edu.

CA6206: DATA STRUCTURES AND ALGORITHMS [3 1 0 4]

Introduction: Definitions, Concept of Data Structures, Overview of Data Structures. Linked Lists: Definition, Representation in memory, Operations, Stacks: Definition, Array and linked-list representation of stack, operations on Stack, Queues: Definition, Array and linked-list representation of Queue. Sorting and Searching: Bubble Sort, Quick Sort, Merge Sort, Linear Search, Binary Search. Tree: Definitions and Concepts, Representation of binary tree, Binary tree traversal, Binary search trees, Applications of Trees, Graph: Matrix Representation of Graphs, Elementary Graph operations. Algorithm Analysis: Time Space Tradeoff, Asymptotic Notations, Properties of asymptotic notations, Recurrence equations, Solving recurrence equations using Substitution method and Master's method, Divide and Conquer: Binary Search, Merge Sort, Quick Sort, Matrix Multiplication, Greedy Algorithms: Knapsack Problem, Job Sequencing with deadline, Optimal Merge Pattern, Single Source Shortest Path, Minimum Cost Spanning tree; Dynamic Programming: Multistage Graphs, Matrix Chain Multiplication, All-Pair shortest paths, Optimal binary search trees, 0/1 Knapsack, Travelling salesperson problem, Graph Traversals.

References:
>   1. A. M. Tenenbaum, Data Structures Using C, (1e), Pearson Education, 2008.
>   2. R. Thareja, Data Structures Using C, (2e), Oxford University Press, 2014.
>   3. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, Introduction to Algorithms, (3e), MIT press, 2009.

CA6233: OBJECT ORIENTED PROGRAMMING USING JAVA LAB [0 0 2 1]

Core java concept implementation, AWT components and their classes, Events handling and Layout manager, Graphical user interface, Swing components and their classes, JDBC and ODBC implementation, JSP and Servlets, Client server applications.

References:
>   1. P. Radhak, Object Oriented Programming Through Java, (1e), Universities Press, 2006
>   2. H. Schildt, The Complete Reference Java Eight Edition, (8e) Tata McGraw-Hill, reprint 2011.
>   3. S. Holzner, Java 2 programming black book, (5e) Dream Tech, New Delhi, reprint: 2005.

CA6234: DATA STRUCTURES AND ALGORITHMS LAB [0 0 2 1]

Programs based on Stacks, Queues, Linked Lists, Programs based on Sorting and Searching algorithms: Bubble, Quick sort, Merge sort, Linear and Binary search, Programs based on Trees and its applications, Graphs and its applications.

References:
>   1. A. M. Tenenbaum, Data Structures Using C, (1e), Pearson Education, 2008.
>   2. R. Thareja, Data Structures Using C, (2e), Oxford University Press, 2014.

CA6235: PRAGMATIC LEARNING [0 0 2 1]

Introduction: Experiential learning (Minor Project) is learning through doing. The aim of this course is to encourage students designing small projects in a multidisciplinary environment. In this course students are challenged to move from problem to solution through a series of task-oriented steps. This collaborative process creates lifelong learners by igniting a curiosity about the world around them. In the experiential learning, students are inspired to build a small project which will enables them to acquire skills to position them for success in both academics and industry. The students will acquire practical knowledge within the chosen area of technology for project development and identify, analyse, formulate, and handle programming projects with a comprehensive and systematic approach.

Course Outcome: At the end of the course, the students will be able to:
- To design, implement and evaluate a project.
- Gain project management skills.
- To learn to work effectively and ethically in a team towards project development.
- To demonstrate the ability to produce a technical document.

Syllabus: There will be general meetings, group discussion and mid-term presentation to track the progress of the project and term-end presentation to evaluate project. In the examination student must demonstrate the project. A team of maximum two students can develop the project. However, during the examination, each student must demonstrate the project individually. Finally, student/team must submit a short project report/summary that must include the following:
- Problem Statement
- Objectives
- Requirement Analysis
- Software Requirement Specification
- Methodology – How you build the project?
- Conclusion
- References

CA6210: APTITUDE AND TECHNICAL DEVELOPMENT [1 1 0 2]

Section I: Quantitative: Number System, Percentage, Time & Distance, Profit & Loss, Time & Work, Average, Permutation & Combinations, Game Based. Verbal: Sentence Improvement, Sentence Rearrangement, Fill in the Blanks. Logical: Coding & Decoding, Direction, Blood Relation, Puzzle, Series, Statement & Arguments. Mock Interview Preparation and Group Discussion.

Section II: C Programming: C Fundamentals, Function, Array, Pointers, Structure and File Handling. Object Oriented Concepts. Data Structure: Types of Data Structure and their implementation. Program Logic Development and MCQ Solving. DBMS; SQL Queries. Software Engineering: Use case preparation and Implementation. Overview of Operating Systems and Computer Networks.

Program Electives – I

CA6245: CRYPTOGRAPHY AND NETWORK SECURITY [3 0 0 3]

Elements of Number Theory : Euclid Algorithm, Prime Number Theorem, Euler's, Fermat's Little theorems, Entropy ; Classical Cipher Techniques: Caesar, Affine, Mono-alphabetic, Transposition, Polyalphabetic Ciphers; Security Attacks: Active V/S Passive, Security Services; Symmetric Encryption: Fiestel Cipher, Confusion and Diffusion, DES Algorithm; Asymmetric Encryption: Principles of Public Key Cryptosystems, RSA Algorithm; Message Authentication & Hashing; Digital Signatures: RSA Based, El-Gamal Signatures; Key distribution; User Authentication Protocols; E-Mail Security: PGP, S/MIME; IPsec: AH & ESP; SSL; TLS; Intrusion Detection: Statistical Anomaly Detection, Rule based detection, honeypots; Password Protection.

References:
1. S. Williams, Cryptography and Network Security: Principles and Practices, (6e) Pearson Education, 2013.
2. A. Kahate, Cryptography and Network Security, (4e) Tata Mc-Graw Hill, 2019
3. K. Charlie, Network Security: Private Communication in a Public World, (2e), Pearson Education, 2016.
4. V. Bagad, I. Dhotre, Cryptography and Network Security, (2e), Technical Publications, 2008.
5. B.A. Forouzan, Network Security, (3e), Tata Mc-Graw Hill, 20011.

CA6246: ARTIFICIAL INTELLIGENCE [3 0 0 3]

General Issues and Overview of AI: The AI problems, what is an AI technique, Characteristics of AI applications. Introduction to LISP programming: Syntax and numeric functions, Basic list manipulation functions, predicates and conditionals, input output and local variables, iteration and recursion, property lists and arrays. Problem Solving, Search and Control Strategies: General problem solving, production systems, control strategies forward and backward chaining, exhausive searches depth first breadth first search. Heuristic Search Techniques Hill climbing, branch and bound technique, best first search & A* algorithm, AND / OR graphs, problem reduction & AO* algorithm, constraint satisfaction problems. Knowledge Representations: First order predicate calculus, skolemization, resolution principle & unification, interface mechanisms, horn's clauses, semantic networks, frame systems and value inheritance, scripts, conceptual dependency.

References:
1. Munakata, T.: Fundamentals of the New Artificial Intelligence, Springer-Verlag New York, Inc., (1e), 1998. ISBN 0-387-98302-3.
2. Cordón, O., Herrera, F., Hoffman, F., Magdalena, L.: Genetic Fuzzy systems, World Scientific Publishing Co. Pte. Ltd., (1e), 2001, ISBN 981-02-4016-3.
3. Kecman, V.: Learning and Soft Computing, The MIT Press, (1e), 2001, ISBN 0-262-11255-8.
4. Mehrotra, K., Mohan, C., K., Ranka, S.: Elements of Artificial Neural Networks, The MIT Press, (1e), 1997, ISBN 0-262-13328-8.

CA6247: INTERNET OF THINGS [3 0 0 3]

Internet of Things: An overview, System Architecture, Design Principles for Connected Devices, Design Principles for Web connectivity for Connected Devices, Internet Connectivity Principles, Data Acquiring, Organizing and Analytics in IoT, data Collection, Storage & Computing Using Cloud Platform, Sensors and Actuators, Radio Frequency Identification, Wireless Sensor Networks and Participatory Sensing Technology, Prototyping of Embedded Devices for IoT, Gateways, Internet and Web/Cloud Services Software Component, IoT Privacy, Security and governance. IoT based Case studies.

References:
1. Theoleyre, Fabrice, and Ai-Chun Pang, eds," Internet of Things and M2M Communications", River Publishers, (1e), 2013.
2. Delsing, Jerker, ed, "IoT automation: Arrowhead framework. CRC Press", (1e), 2017.
3. Raj Kamal, "Internet of Things", (1e), McGraw-Hill, 2017.

CA7104: UNIX & SHELL PROGRAMMING [3 1 0 4]

Introduction: UNIX System Overview, Program and Processes, Error Handling, User Identification, Signals, System Calls and Library Functions.: File I/O: File Descriptors, Function for File Modification, I/O Efficiency, File Sharing, Atomic Operations.; Directories: Stat, Fstat, and Lstat Functions, File Types, Set-User-ID and Set-Group-ID, File Access Permissions, Function for modifying file permission and ownership, Symbolic Links, System Data Files and Information: Password File, Shadow Passwords and Other Data Files.; Process Environment: Process Termination, Memory Layout of a C Program, Memory Allocation, setjmp and longjmp Functions.; Process Control: fork Function, vfork Function, exit Functions, wait and waitpid Functions, Race Conditions, Changing User IDs and Group IDs.; Process Relationship: Logins, Process Groups, Sessions, Controlling Terminal, Job Control.; Signals: Signal Concepts, Functions to raise and handle Signals, Program Termination, abort and system functions.; Threads: Thread Concepts, Creation, Termination and Synchronization, Threads Control, Threads and Signals, Threads and fork, Threads and I/O. Shell programming: Basics of Shell Programming, UNIX shell commands, shell scripts variables, loops (for, while), and conditional statements (if else, case), Shell variables, arguments to shell procedure, test command, arithmetic with EXPR command, interactive shell procedures with read.

References:

1. W. R. Steven, S. A. Rago "Advanced Programming in the Unix environment", Addison Wesley, (1e), 2011
2. Y. P. Kanetkar "Unix Shell Programming". BPB Publication, (1e), 2009.

CA7105: ANDROID APPLICATION DEVELOPMENT [3 1 0 4]

Basic Concept of Mobile Operating Systems Introduction, Design Principal, Structure, platform and features of mobile operating system comparison of mobile OS to traditional OS Application life cycle Micro Kernel Design concept of MOS. Introducing Development Framework, Dalvik Virtual Machine, Android Studio IDE, Android Virtual Device and SDK Manager, Android Architecture and OOPS, Android Architecture and OOPS, Activity Lifecycle, Activity Classes Introduction to Application Manifest, Android Layouts, Understanding Android Menus , Inter process Communication , Multithreading, JSON Parsing, SQLite Data Types , Cursors and Content Values, SQLite Open Helper, Adding, Updating and Deleting Content. Working with Android Sensor, Camera, Wi-Fi.

References:

1. John Horton, "Android Programming for Beginner",Packt Publishing, (2e), 2018.
2. Dawn Griffiths, David Griffiths, "Head First Android Development: A Brain-Friendly", O'Reilly Media, (2e), 2017.
3. Pradeep kothari, "Android Application Development (With Kitkat Support), Black Book", WILEY, (1e) (May 20, 2014).

CA7106: SOFTWARE ENGINEERING & PROJECT MANAGEMENT [3 1 0 4]

Introduction to Software Engineering: Software Components, Software Characteristics, Software Crisis, Software Engineering Processes; Software Development Life Cycle (SDLC) Models: Water Fall Model, Prototype Model, Spiral Model, Agile; Requirement Engineering Process: Analysis, Documentation, Review and Management of User Needs, Data Flow Diagrams, Entity Relationship Diagrams, Decision Tables, SRS Document, IEEE Standards for SRS; Basic Concept of Software Design, Architectural Design, Low Level Design Modularization, Design Structure Charts, Pseudo Codes, Flow Charts, Coupling and Cohesion Measures; Design Strategies: Function Oriented Design, Object Oriented Design, Top-Down and Bottom-Up Design; Software Testing: Definition, Test characteristics, Types of testing - Black-Box Testing, White-Box Testing, Stress Testing, Performance Testing; Categories of Maintenance: Preventive, Corrective and Perfective Maintenance, Cost of Maintenance, Software Re-Engineering, Reverse Engineering; The Management spectrum- (The people, the product, the process, the project), Estimation of Various Parameters such as Cost, Efforts, Schedule/Duration, Constructive Cost Models (COCOMO) and its types, SEI capability maturity model, Verification and Validation, SQA Plans, Software Quality Frameworks, ISO 9000 Models. Basics of Project Management: Introduction, Need for Project Management, Project Management Knowledge Areas and Processes, The Project Life Cycle, The Project Manager (PM), Phases of Project Management Life Cycle, Project Management Processes, Impact of Delays in Project Completions, Essentials of Project Management Philosophy, Project Management Principles.

References:
1. R. S. Pressman, Software Engineering: A Practitioners Approach, (3e) McGraw Hill, 2009.
2. R. Mall, Fundamentals of Software Engineering, (4e) PHI Publication, 2014.
3. K. K. Aggarwal and Y. Singh, Software Engineering, (3e) New Age International Publishers, 2008.
4. P. Jalote, Software Engineering, Wiley, (1e) 2010.
5. Sommerville, Software Engineering, Addison Wesley, (10e) 2013.

CA7131: MINOR PROJECT [0 0 4 2]

Introduction: The goal of the mini project is to provide students the practical skills and knowledge they need to address issues that arise in the workplace, in educational settings, and in computer science research. The course's mini-project entails doing hands-on work to comprehend and address issues in the field of computers. An information system or subsystem, like a piece of software, is typically analysed, designed, coded or otherwise implemented, and tested as part of any computer science project. A design document might be the proper outcome of a design study instead of a computer programme being the subsystem. However, in this course, we expect a software system or subsystem. The design and implementation of a hardware system/subsystem would also be a suitable project.

Course Outcome
- CA7131.1 To demonstrate a depth of knowledge of modern technology.
- CA7131.2 Design Understand about project organization and feasibility analysis in Project Management.
- CA7131.3 To complete an independent project using Software Development Life Cycle.
- CA7131.4 To acquire the skills to communicate effectively and to present ideas clearly and coherently to specific audience in both the written and oral forms.
- CA7131.5 To reflect learning and take appropriate actions to improve entrepreneur skills.

Syllabus: The Mini Project is not just a component of the coursework; it also serves as a way for you to highlight your skills and areas of expertise. It gives you the chance to show off your creativity, cooperation, inspiration, planning, and organizational skills in a software project.

Textbook(s):
1. Prasanna Chandra; Projects- Planning, Analysis, Selection, Financing, Implementation and Review',VI Edition, Tata Mc Graw Hill.

Reference Book(s):
1. Chaudhary S.; Project Management, Tata Mc Graw Hill
2. Kerzner H.; Project Management, II Edition, CBS Publishers

Project Guidelines
- Each student should submit a unique project title unless/otherwise in a team project.
- Project work should include software development.
- Only two students can work on one project as a team. However, there contribution should be clearly specified and reported.
- The project should focus on solving some real-life problems, though it is not mandatory. However, the project idea should be creative, and it can be a fresh take on an old idea which is often worth as much as a brand-new idea.
- The project work may be done internally in the university campus or in any external organizations/institutes approved by the head of the department/university authority.
- Prior to starting project work, a student must get his/her project idea/problem statement approved by the supervisor.
- The student must submit a project synopsis, presenting his idea. The student may start working on project only if the synopsis is approved.
- The student should present the progress of the project works as per the timeline specified by the department /project coordinator/ supervisor.

Project Synopsis Format
The project synopsis must be prepared and approved with the supervisor's input. The synopsis should include a detailed description of the proposed project and objectives. The synopsis should be prepared as per the following format.

- Title of the project
- Name of the supervisor/project guide
- Project Introduction
- Objectives of the project
- DFD, ER Diagrams
- Project Timeline
- Tools / platform, hardware and software requirement specifications
- References

Project Report Format
The final project report should describe the detailed work completed by the student. The report must be prepared as per the following format.

General Guidelines
- Project Report to be minimum 35 pages. Reports less than 35 pages will be rejected.
- Project report to be maximum 50 - 60 pages (preferred but not mandatory).
- Paper Size: A4; Left = Right = Top = Bottom Margins = 0.7".
- Page Numbering Position: Bottom with right justified and continuous numbering from the Introduction Chapter.
- Use Times New Roman Font with Normal Style, paragraph justified and 1.15 line spacing.
- Paragraph Heading: Times New Roman Font, Bold, Font Size 14; Paragraph Matter: Times New Roman Font, Normal, Font Size 12.
- Sub-paragraphs be appropriately numbered as in 1.1, 1.2, 1.3 etc; Sub-paragraph Heading: Times New Roman Font, Italics, Font Size 12; Sub-paragraph Matter: Times New Roman Font, Normal, Font Size 12.
- Figure captions below Figure with chapter wise numbering.
- Tables captions above Table with chapter wise numbering.
- All references must be listed in the order in which they appear in the report (follow IEEE format for referencing).
- Only hard bound reports will be accepted, colour of the front cover to be in mustard yellow.
- Note: The Cover page color as mentioned above has CMYK Values are C: 00 M:20 Y:75 K:00 & Hex is: FFCC00

Project Report Structure
The following structure should be followed while preparing the final project report.
1. Title Page
2. Certificate of Completion (internal/External)
3. Acknowledgement
4. Table of contents / index with page numbering
5. List of tables
6. List of figures
7. Introduction / objectives of the project
8. System analysis
9. Feasibility study
10. Software and hardware requirement specifications
11. System design (DFD, ER Diagram, Class diagram etc.)
12. Database Schema
13. Project code
14. Screenshot of the project
15. Implementation/deployment details
16. Testing (testing techniques and testing strategies used along with the test data and the errors listed for each test case).
17. Conclusion
18. Future scope and further enhancement of the project
19. Bibliography/ references
20. Appendices (if required)

Note: Reports, tables figures should be properly numbered/labelled. Two hard copies of the project report should be submitted. The soft copy of the project report in PDF should also be submitted along with the hard copy.

CA7132: UNIX & SHELL PROGRAMMING LAB [3 1 0 4]

Testing the use of UNIX commands, UNIX shell commands, Basics of Shell Programming, UNIX System Calls, CPU Scheduling Algorithms, Deadlock Detection Algorithms, Deadlock Avoidance Algorithms, Page Replacement Algorithms, Memory Allocation Algorithms, Disk Scheduling Algorithms, and UNIX Inter Process Communication.

References:

1. W. R. Steven, S. A. Rago "Advanced Programming in the Unix environment", Addison Wesley, 2011.
2. Y. P. Kanetkar "Unix Shell Programming". BPB Publication, 2009.

CA7133: ANDROID APPLICATION DEVELOPMENT LAB [0 0 2 1]

Basic Concept of Mobile Operating Systems, setting up development environment, Dalvik Virtual Machine & .apk file extension, AndroidManifest.xml, Resources & R.java, Activities and Activity lifecycle, Launching emulator, Hello World App, switching between activities, Form widgets, Text Fields, Layouts, Option menu, Context menu, Explicit Intents, Implicit intents, Time and Date, Images, Media, ListView, ListActivity, Custom listview, Threads handling, SQLite Programming, SQLiteOpenHelper, SQLiteDatabse, XML Parsing, JSON Parsing, Using Orientation and Accelerometer sensors, Monitoring and managing Internet connectivity, Taking pictures, Media Recorder.

References:

1. John Horton, "Android Programming for Beginner", Packt Publishing, (2e), 2018.
2. Dawn Griffiths, David Griffiths, "Head First Android Development: A Brain-Friendly", O'Reilly Media, (2e), 2017.
3. Pradeep kothari, "Android Application Development (With Kitkat Support), Black Book", WILEY, (1e) (May 20, 2014).

Program Electives – II

CA7152: DATA MINING TECHNIQUES [3 0 0 3]

Introduction: Databases, Data Warehouses, Transactional databases, advanced database system and its applications, Data mining Functionalities: Class description, Association Analysis classification & Prediction, Cluster Analysis, Outlier Analysis, Evolution Analysis, Classification of Data Mining Systems, Major Issues in Data Mining. Data Warehouse and OLAP Technology for Data Mining: Differences between Operational Database Systems and Data Warehouses, a multidimensional Data Model, Data Warehouse Architecture, Data Warehouse Architecture, Data Warehouse Implementation, Data Cube Technology. Data Pre-processing: Data Cleaning, Data Integration and Transformation, Data Reduction, Discretization and Concept Hierarchy Generation. Data Mining Primitives, Languages, and System Architectures, Concept Description: Characterization and Comparison, Analytical Characterization. Mining Association Rules in Large Databases: Association Rule Mining, Market Basket Analysis, Basic Concepts, Mining Single-Dimensional Boolean Association Rules from Transactional Databases: the Apriori algorithm, Generating Association rules from frequent items, Multidimensional Association Rules, Constraint-Based Association Mining. Classification & Prediction and Cluster Analysis: Issues regarding classification & prediction, Different Classification Methods, Prediction, Cluster Analysis.

References:

1. Kamber and Han, "Data Mining Concepts and Techniques", Hartcourt India P. Ltd.,2001.
2. Paul Raj Poonia, "Fundamentals of Data Warehousing", John Wiley & Sons, 2003.
3. Efrem O, Mallach, "Decision Support and Data Warehousing Systems", Mcgraw-Hill International Edition, 2000.
4. W.H. Inmon "Building the Datawarehouse, (3e), Wiley India, 2016.
5. Anahory, "Data Warehousing in Real World", (2e), Pearson Education, 2017.

CA7153: CLOUD COMPUTING & INFRASTRUCTURE SERVICES [3 0 0 3]

Introduction to Clouds and Cloud Computing: Basic Concepts, Cloud Classifications, and Types of Services, deployment models; Classic Data Center (CDC): DBMS concepts, CDC drawbacks and need of Cloud Resources, CDC Management and case studies; Virtualized Data Center (VDC): Compute and Storage, Compute virtualization overview, Compute virtualization techniques, Virtual Machines, VM Resource management techniques.

Physical to virtual conversion, Hypervisor Management Software, Virtual Infrastructure Requirements; Storage: Storage virtualization overview, Virtual Machine Storage, Block level and File level virtualization, Virtual provisioning and automated storage tiering; Networking: VDC networking overview, VDC networking components , VLAN and VSAN technologies, Network traffic management Desktop and Application: Desktop virtualization , Application virtualization, Business Continuity in VDC, Fault tolerance mechanism in VDC, Backup in VDC, Replication and migration in VDC, Cloud Security: Security basics, Cloud security concerns and threats, Cloud security mechanisms, Access control and identity management in Cloud.

References:
1. Miller M, Cloud Computing, (8e), Que Publishers 2008.
2. Buyya R K, Cloud Computing: Principles and Paradigms, Wiley Press, (1e), 2011.
3. K Saurabh, Cloud Computing, (2e), Wiley India, 2017
4. V Joysula, M Orr, G Page, Cloud Computing: Automating the Virtualized Data Center: Cisco Press, (1e), 2012.
5. Mei- Ling Liu, "Distributed Computing: Principles and Application", Pearson Education, Inc. New Delhi, (1e), 2004.

CA7154: ADVANCED COMPUTER NETWORKS [3 0 0 3]

Introduction: Protocols and Standards: Introduction, Computer Networks, Types of topology, Categories of network, Network Models, OSI model, Internet model, Comparison of the OSI & the TCP/IP reference models, Addressing, Network Hardware Components. Multiplexing and Local Area Network: Introduction, Multiplexing, Frequency division multiplexing, Wavelength division multiplexing, Time division multiplexing, LAN Technologies, Ethernet, Token Ring, Fiber distributed data interface (FDDI), Switching, Circuit switching, Packet switching, Message switching. Optical Networking: Introduction, SONET/SDH Standards, Dense Wavelength Division Multiplexing (DWDM). Integrated Service Digital Network, Broadband ISDN, ATM Cells, ATM Adaption Layer (AAL), AAL protocols, Congestion Control in ATM Networks, Traffic characterization, Congestion control. Packet Switching Protocols: Introduction, X.25 Networks, Frame Relay, Internet protocols, Transmission Control Protocol, User Datagram Protocol, Internet Control Message Protocol, Hyper Text Transfer Protocol. Routing in Internet: Introduction, Unicast Routing Protocols, Intra-Domain Routing, Distance vector routing, Link state routing, Broader Gateway protocol (BGP), BGP Routing, BGP message types, Inter-Domain Routing Protocol (IDRP), Multicast Routing Protocols, Distance Vector Multicast Routing Protocol (DVMRP), Multicast Open Shortest Path First Protocol (MOSPF). Network Management: Introduction, Network Management System, Network management problems, Network management system architecture, Simple Network Management Protocol (SNMP), SNMP Message Format. Network Security: Introduction, Basics of Cryptography, Symmetric Key Encryption, Data encryption standard (DES), Public Key Encryption, RSA algorithm, Digital Signatures. Web Security and Traffic Management Basics: Introduction, Web Security Requirements, Secure Socket Layer (SSL), SSL architecture, SSL protocol, Traffic Management, Quality Characteristics for Network, Network characteristics and requirements. Quality of Service and Queue Analysis: Introduction, Applications and Quality of Service. Multimedia Over Internet: Introduction, Quality of Service Aspects of Multitasking, Resource Reservation Protocol (RSVP), RSVP traffic types, RSVP operation, RSVP messages, Real-Time Transport Protocol (RTP). Enterprise Network Security: Introduction, Demilitarized Zone (DMZ), Network Address translation, Destination network address translation (DNAT), Secure network address translation (SNAT), Port Forwarding, Filtering, Packet Filtering, Application Gateway.

References:
1. Kahate, Atul. Cryptography and network security. Tata McGraw-Hill Education, 2013.
2. Buckwalter, Jeff T. Frame relay: Technology and practice. Addison-Wesley Professional.
3. Perros, Harry G. Connection-oriented networks: SONET/SDH, ATM, MPLS and optical networks. John Wiley & Sons.
4. Peterson, Larry L., and Bruce S. Davie. Computer networks: a systems approach. Elsevier.
5. Olifer, Natalia, and Victor Olifer. Computer networks: Principles, technologies and protocols for network design. New York, NY, USA: John Wiley & Sons.

Program Electives – III

CA7155: ADVANCES IN MACHINE LEARNING [3 0 0 3]

Introduction of Machine Learning, types of data, tools for machine learning, types of machines learning techniques, understanding mathematical modelling, classifications and prediction of data, evaluation of machine learning algorithm (Confusion Matrix, RoC, AUC), logistic regression, decision tree, Naive Bayes. Hierarchical Clustering, Principal Component Analysis.

Neural Network, multilayer perceptron, distance or similarity based neural networks, associative memory and self-organizing feature map; radial basis function based multilayer perceptron. Introduction to CNN and RNN.

References:
1. Burkov, Andriy. The hundred-page machine learning book. Vol. 1. Quebec City, QC, Canada: Andriy Burkov, 2019.
2. Burkov, Andriy. Machine learning engineering. Vol. 1. True Positive Incorporated, 2020.
3. Bonaccorso, Giuseppe. Machine learning algorithms. Packt Publishing Ltd, 2017.

CA7156: BLOCKCHAIN TECHNOLOGIES [3 0 0 3]

Introduction – basic ideas behind blockchain, how it is changing the landscape of digitalization, introduction to cryptographic concepts required; Hashing, public key cryptosystems, private vs public blockchain and use cases, Hash Puzzles, Introduction to Bitcoin Blockchain; Bitcoin Blockchain and scripts, Use cases of Bitcoin Blockchain scripting language in micropayment, escrow etc, Downside of Bitcoin – mining; Alternative coins – Ethereum and Smart contracts; Alternative coins – Ethereum continued, IOTA; The real need for mining – consensus – Byzantine Generals Problem, and Consensus as a distributed coordination problem – Coming to private or permissioned blockchains – Introduction to Hyperledger; Permissioned Blockchain and use cases – Hyperledger, Corda; Uses of Blockchain in E-Governance, Land Registration, Medical Information Systems, and others.

References:
1. Kumar Saurabh, Ashutosh Saxena, Blockchain Technology: Concepts and Applications, Wiley, 2020.
2. Daniel Drescher, Blockchain Basics, A Non-Technical introduction in 25 steps, Apress, 2017.

CA7157: DATA SCIENCE [3 0 0 3]

Data Science Introduction: Introduction to data, types of data (quantitative and qualitative). Level of data: Nominal, Ordinal, Scale, Interval. Introduction data science, data science process, role of data scientist, different tools for data science (R, Python, Excel, Tableau, Power BI,). Handling Missing Data, Decoding of Data. Treatment of Outliers. Data visualization: scatter plot, line plot, Box plot, bar plot, stem and leaf plot. Data Distribution: Normal, Binomial, Poisson. Measures of central tendencies, measures of variations. Data correlation, data classifications and prediction, regression analysis, Decision Tree, Naïve Bayes.

References:
1. Andrew Wolf, Machine Learning Simplified: A Gentle Introduction to Supervised Learning, themlsbook.com, 2022.
2. Peter Bruce and Andrew Bruce, Practical Statistics for Data Scientists, Publisher(s): O'Reilly Media, Inc., 2017.

FOURTH SEMESTER

CA7270: MAJOR PROJECT [0 0 0 16]

Introduction: Each student shall carry out an industry level project in this semester. The project will be carried out under the supervision of a teacher of the department. When the project is carried out in an external organization (academic institution/ industry), a supervisor will also be appointed from the external organization.

Course Outcome
- CA7270.1 To demonstrate in-depth knowledge and application of development technology.
- CA7270.2 Understand about project organization and project management.
- CA7270.3 To complete an independent project using software development life cycle.
- CA7270.4 To acquire the skills to communicate effectively and to present ideas clearly and coherently to specific audience in both written and oral forms.
- CA7270.5 To reflect learning and take appropriate actions to improve entrepreneurship skills.

Project Guidelines
- Each student should submit a unique project title unless/otherwise in a team project.
- Project work should include software development.
- Only two students can work on one project as a team. However, their contribution should be clearly specified and reported.
- The project should focus on solving some real-life problems, though it is not mandatory. However, the project idea should be creative, and it can be a fresh take on an old idea which is often worth as much as a brand-new idea.

- The project work may be done internally in the university campus or in any external organizations/institutes approved by the head of the department/university authority.
- Prior to starting project work, a student must get his/her project idea/problem statement approved by the supervisor.
- The student must submit a project synopsis, presenting his idea. The student may start working on project only if the synopsis is approved.
- The student should present the progress of the project works as per the timeline specified by the department /project coordinator/ supervisor.

Project Synopsis Format

The project synopsis must be prepared and approved with the supervisor's input. The synopsis should include a detailed description of the proposed project and objectives. The synopsis should be prepared as per the following format.
- Title of the project
- Name of the supervisor/project guide
- Project Introduction
- Objectives of the project
- DFD, ER Diagrams
- Project Timeline
- Tools / platform, hardware and software requirement specifications
- References

Project Report Format

The final project report should describe the detailed work completed by the student. The report must be prepared as per the following format.

General Guidelines
- Project Report to be minimum 35 pages. Reports less than 35 pages will be rejected.
- Project report to be maximum 50 - 60 pages (preferred but not mandatory).
- Paper Size: A4; Left = Right = Top = Bottom Margins = 0.7".
- Number of hard bound copies – Two (2)
- Page Numbering Position: Bottom with right justified and continuous numbering from the Introduction Chapter.
- Use Times New Roman Font with Normal Style, paragraph justified and 1.15 line spacing.
- Paragraph Heading: Times New Roman Font, Bold, Font Size 14; Paragraph Matter: Times New Roman Font, Normal, Font Size 12.
- Sub-paragraphs be appropriately numbered as in 1.1, 1.2, 1.3 etc; Sub-paragraph Heading: Times New Roman Font, Italics, Font Size 12; Sub-paragraph Matter: Times New Roman Font, Normal, Font Size 12.
- Figure captions below Figure with chapter wise numbering.
- Tables captions above Table with chapter wise numbering.
- All references must be listed in the order in which they appear in the report (follow IEEE format for referencing).
- Only hard bound reports will be accepted, colour of the front cover to be in mustard yellow (refer format).
- Note: The Cover page color as mentioned above has CMYK Values are C:00 M:20 Y:75 K:00 & Hex is: FFCC00

Project Report Structure

The following structure should be followed, to whatever extent possible, while preparing the final project report.
- Title Page
- Certificate of Completion (Internal/External)
- Abstract
- Acknowledgement
- List of tables
- List of figures
- Table of contents (with page numbering)
- Introduction

- o Company Profile
- o Existing System and Need for System
- o Scope of Work
- o Operating Environment - Hardware and Software
- System Analysis and Design
  - o Feasibility study
  - o Software and hardware requirement specifications
  - o Data Flow Diagram (DFD)
  - o Functional Decomposition Diagram (FDD)
  - o Entity Relationship Diagram (ERD)
  - o Data Dictionary
  - o Table Design
  - o Code Design
  - o Menu Tree
  - o Menu Screens
  - o Input Screens
  - o Test Procedures and Implementation
- Project code
- Screenshots of the project
- Implementation/deployment details
- Conclusion
- Future scope of the project
- Drawbacks and Limitations
- Bibliography/ References
- Annexures (if required)

Note: Reports, tables figures should be properly numbered/labelled. Two hard copies of the project report should be submitted. The soft copy of the project report in PDF should also be submitted along with the hard copy.