Bansilal Ramnath Agarwal Charitable Trust's

Vishwakarma Institute of Technology

(An Autonomous Institute affiliated to Savitribai Phule Pune University)

Structure & Syllabus

B.Tech. (Computer Engineering)

With Effect from Academic Year 2024-25

Prepared by: - Board of Studies in Computer Engineering

Approved by: - Academic Board, Vishwakarma Institute of Technology, Pune

Chairman – BOS **Chairman – Academic Board**

Vision of the Institution

"To be globally acclaimed Institute in Technical Education and Research for holistic Socio-economic development".

Mission of the Institution

- To ensure that 100% students are employable and employed in Industry, Higher Studies, become Entrepreneurs, Civil / Defense Services / Govt. Jobs and other areas like Sports and Theatre.
- To strengthen Academic Practices in terms of Curriculum, Pedagogy, Assessment and Faculty Competence.
- Promote Research Culture among Students and Faculty through Projects and Consultancy.
- To make students Socially Responsible Citizen.

Vision of the Department

"To be a leader in the world of computing education practising creativity and innovation".

Mission of the Department

- To ensure students' employability by developing aptitude, computing, soft, and entrepreneurial skills
- To enhance academic excellence through effective curriculum blended learning and comprehensive assessment with active participation of industry
- To cultivate research culture resulting in knowledge-base, quality publications, innovative products and patents
- To develop ethical consciousness among students for social and professional maturity to become responsible citizens

Program Outcomes [PO]

PO	Program Outcome Statements
PO1	**Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
PO2	**Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
PO3	**Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
PO4	**Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
PO5	**Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO6 **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7 **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8 **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9 **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10 **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11 **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12 **Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Program Specific Outcomes (PSOs)

Program Specific Outcome Statements

PSO1 Select and incorporate appropriate computing theory principles, data structures and algorithms, programming paradigms to innovatively craft scientific solution addressing complex computing problems.

PSO2 Adapt to new frontiers of science, engineering and technology by getting acquainted with heterogeneous computing environments and platforms, computing hardware architectures and organizations through continuous experimentation.

PSO3 Conceive well-formed design specifications and constructs assimilating new design ideas and facts for identified real world problems using relevant development methodologies and practices, architecture styles and design patterns, modeling and simulation, and CASE tools.

PSO4 Exercise research and development aptitude focusing knowledge creation and dissemination through engineering artifacts construction, preparation and presentation of engineering evidences using procedures, techniques, guidelines, and standards considering technology migration and evolution.

Program Educational Objectives (PEOs)

- Demonstrate application of sound engineering foundations to be a committed technology workforce
- Apply mathematical and computing theory knowledge base to provide realistic computer engineering solutions
- Exhibit problem-solving skills and engineering practices to address problems faced by the industry with innovative methods, tools, and techniques
- Develop professional and ethical practices adopting effective guidelines to acquire desired soft skills in the societal and global context
- Aim for continuing education and entrepreneurship in emerging areas of computing

Course Name Nomenclature as per NEP (For FY and SY)

BSC: Basic Science Course	MDOE: Multi Disciplinary Open Elective
ESC: Engineering Science Course	CC: Co-curricular Course
PCC: Program Core Course	HSSM: Humanities Social Science and Management
PEC: Program Elective Course	IKS: Indian Knowledge System
ELC: Experiential Learning Course	FP: Field Project
MD: Multi Disciplinary	INT: Internship

Nomenclature for Teaching and Examination Assessment Scheme AY 2024-25

Sr No.	Category	Head of Teaching/ Assessment	Abbreviation used
1	Teaching	Theory	Th
2	Teaching	Laboratory	Lab
3	Teaching	Tutorial	Tut
4	Teaching	Open Elective	OE
5	Teaching	Multi Disciplinary	MD
6	Teaching	Computer Science	CS
7	Assessment	Laboratory Continuous Assessment	CA
8	Assessment	Mid Semester Assessment	MSA
9	Assessment	End Semester Assessment	ESE
10	Assessment	Home Assignment	HA
11	Assessment	Course Project	CP
12	Assessment	Group Discussion	GD
13	Assessment	PowerPoint Presentation	PPT
14	Assessment	Class Test –1	CT1
15	Assessment	Class Test –2	CT2
16	Assessment	Mid Semester Examination	MSE
17	Assessment	End Semester Examination	ESE
18	Assessment	Written Examination	WRT
19	Assessment	Multiple Choice Questions	MCQ
20	Assessment	Laboratory	LAB

Semester I

Title: Course Structure **FF No. 653**

Branch: Computer **Year:** T.Y. **A.Y.:** 2025-26 **Module: V**

Subject No.	Subject Code	Subject Name	Teaching Scheme (Hrs/Week)			Examination Scheme									Credits
			Th	Lab	Tut	CA	MSA	ESA						Total	
						LAB (%)	MSE (%)	HA (%)	LAB (%)	CP (%)	PPT /GD (%)	CVV (%)	ESE (%)		
S1	CS3052	Computer Networks (Modules V & VI))	2	2	1	10	-	20	-	20	-	20	30 (WRT)	100	4
S2	Coursera *	Coursera* (Modules V & VI))											100	100	4
S3	CS3205	Design and Analysis of Algorithms	2	2	1	10	-	-	-	20	20 (PPT)	20	30 (WRT)	100	4
S4	CS3061	Software Modelling and Design	2	2	1	10	-	-	-	20	20 (GD)	20	30 (MCQ)	100	4
S5	CS3059	Design Thinking-5	-	-	-	-	-	-	-	-	-	-	100	100	1
S6	CS3057	Engineering Design and Innovation – V	-	12	-	-	30	-	-	-	-	-	70	100	6
S7	SH3001	Reasoning and Aptitude Development	-	-	-	-	-	-	-	-	-	-	-	-	1
S8	MD3154	Advanced Problem Solving and Programming (Audit course)	-	-	-	-	-	-	-	-	-	-	-	-	0
S8	MD3155	Data Engineering (Audit course)	-	-	-	-	-	-	-	-	-	-	-	-	0
		Total	**6**	**18**	**3**	**30**	**30**	**20**	**0**	**60**	**40**	**60**	**360**	**600**	**24**

Semester-I

Title: Course Structure **FF No. 653**

Branch: Computer **Year:** T.Y. **A.Y.:** 2025-26 **Module:VI**

Subject No.	Subject Code	Subject Name	Teaching Scheme (Hrs/Week)			Examination Scheme									Credits
			Th	Lab	Tut	CA	MSA	ESA						Total	
						LAB (%)	MSE (%)	HA (%)	LAB (%)	CP (%)	PPT /GD (%)	CVV (%)	ESE (%)		
S1	CS3052	Computer Networks (Modules V & VI))	2	2	1	10	-	20	-	20	-	20	30 (WRT)	100	4
S2	Coursera*	Coursera* (Modules V & VI))	-	-	-	-	-	-	-	-	-	-	100	100	4
S3	CS3202	Artificial Intelligence	2	2	1	10	-	-	50	20	-	20	-	100	4
S4	CS3334	Cryptography and Information Security	2	2	1	10	-	-	-	20	20 (PPT)	20	30 (WRT)	100	4
S5	CS3059	Design Thinking-5	-	-	-	-	-	-	-	-	-	-	100	100	1
S6	CS3057	Engineering Design and Innovation – V	-	12	-	-	30	-	-	-	-	-	70	100	6
S7	SH3001	Reasoning and Aptitude Development	-	-	-	-	-	-	-	-	-	-	-	-	1
S8	AC	Advanced Problem Solving and Programming (Audit course)	-	-	-	-	-	-	-	-	-	-	-	-	0
S8	AC	Data Engineering (Audit course)	-	-	-	-	-	-	-	-	-	-	-	-	0
		Total	**6**	**18**	**3**	**30**	**30**	**20**	**50**	**60**	**20**	**60**	**330**	**600**	**24**

Coursera*

Subject Code	Subject Name	Subject Code	Subject Name
MD3101	IBM Full Stack Software Developer	MD3126	Meta iOS/Android Developer
MD3102	Meta Back-End Developer	MD3135	Salesforce Sales Development Representative
MD3113	Google Data Analytics	MD3140	SAP Technology Consultant
MD3120	IBM Data Warehouse Engineer	MD3141	AWS Cloud Technology Consultant
MD3121	IBM DevOps and Software Engineering		

Audit Courses

Subject Code	Subject Name	Subject Code	Subject Name
MD3144	Basics of Game Development--offered by Zensar Technologies	Course 2	The Synergy of Sensors and PLCs in Automation Systems
MD3146	Main Frame Technologies –offered by BMC	Course 3	Unlocking the Potential of Industrial Automation Through Robotics
MD3145	Data Engineering ----offered by Barclays (AbInitio)	Course 4	Industrial Automation & Skill Development Solutions --- offered by SSIGMA
Course 1	SMART City		

Semester-II

Title: Course Structure **FF No. 653**

Branch: Computer **Year:** T.Y. **A.Y.:** 2025-26 **Module: V**

Subject No.	Subject Code	Subject Name	Teaching Scheme (Hrs/Week)			Examination Scheme									Credits
			Th	Lab	Tut	CA	MSA	ESA						Total	
						LAB (%)	MSE (%)	HA (%)	LAB (%)	CP (%)	PPT /GD (%)	CVV (%)	ESE (%)		
S1	CS3215	Web Technology	2	2	1	10	-	-	50	20	-	20	-	100	4
S2	CS3053	Compiler Design	2	2	1	10	-	20	-	20	-	20	30 (MCQ)	100	4
S3	CS3205	Design and Analysis of Algorithms	2	2	1	10	-	-	-	20	20 (PPT)	20	30 (WRT)	100	4
S4	CS3061	Software Modelling and Design	2	2	1	10	-	-	-	20	20 (GD)	20	30 (MCQ)	100	4
S5	CS3062	Design Thinking-6	-	-	-	-	-	-	-	-	-	-	100	100	1
S6	CS3058	Engineering Design and Innovation – VI	-	12	-	-	30	-	-	-	-	-	70	100	6
S8	SH3002	Reasoning and Aptitude Development	-	-	-	-	-	-	-	-	-	-	-	-	1
S9	MD3149	Advanced Game Development and Experience Design	-	-	-	-	-	-	-	-	-	-	-	-	0
		Total	**8**	**20**	**4**	**40**	**30**	**20**	**50**	**80**	**40**	**80**	**260**	**600**	**24**

Semester-II

Title: Course Structure **FF No. 653**

Branch: Computer **Year:** T.Y. **A.Y.:** 2025-26 **Module:VI**

Subject No.	Subject Code	Subject Name	Teaching Scheme (Hrs/Week)			Examination Scheme									Credits
			Th	Lab	Tut	CA	MSA	ESA						Total	
						LAB (%)	MSE (%)	HA (%)	LAB (%)	CP (%)	PPT /GD (%)	CVV (%)	ESE (%)		
S1	CS3215	Web Technology	2	2	1	10	-	-	50	20	-	20	-	100	4
S2	CS3053	Compiler Design	2	2	1	10	-	20	-	20	-	20	30 (MCQ)	100	4
S3	CS3202	Artificial Intelligence	2	2	1	10	-	-	50	20	-	20	-	100	4
S4	CS3334	Cryptography and Information Security	2	2	1	10	-	-	-	20	20 (PPT)	20	30 (WRT)	100	4
S5	CS3062	Design Thinking-6	-	-	-	-	-	-	-	-	-	-	100	100	1
S6	CS3058	Engineering Design and Innovation – VI	-	12	-	-	30	-	-	-	-	-	70	100	6
S8	SH3002	Reasoning and Aptitude Development -2	-	-	-	-	-	-	-	-	-	-	-	-	1
S9	MD3149	Advanced Game Development and Experience Design	-	-	-	-	-	-	-	-	-	-	-	-	0
		Total	**8**	**20**	**4**	**40**	**30**	**20**	**100**	**80**	**20**	**80**	**230**	**600**	**24**

Semester-I

Title: Course Structure **FF No. 653**

Branch: Computer **Year:** BTech **A.Y.:** 2024-25 **Module: VIII (Course Work)**

Subject No.	Subject Code	Subject Name	Teaching Scheme (Hrs/Week)			Examination Scheme								Credits
			Th	Lab	Tut.	CA		MSA		ESA			Total	
						HA (%)	LAB (%)	MSE (%)	PPT (%)	ESE (%)	GD (%)	Viva (%)		
OE1	LinkedIn Learning *	LinkedIn Learning	0							100			100	2
OE2	CS4217	Human Computer Interaction	2	-	-	10	-	30	-	30	-	30	100	2
	CS4209	Parallel Computing	2	-	-	10	-	30	-	30	-	30	100	2
	CS4222	Image Processing	2	-	-	10	-	30	-	30	-	30	100	2
OE3	CS4275 Swayam	Introduction to Machine Learning	2	-	-	10	-	30	-	30	-	30	100	2
	CS4276 Swayam	Deep Learning	2	-	-	10	-	30	-	30	-	30	100	2
Major Project	CS4225	Major Project	0	20	-	-	-	30	-	70	-	-	100	9
	CS4292	Design Thinking -7	-	-	-	-	-	-	-	100	-	-	100	1
		Total	**4**	**20**	-	**20**	-	**90**	-	**330**	-	**60**	**500**	**16**

LinkedIn Learning Courses*

Subject Code	Subject Name	Subject Code	Subject Name
MD4274	Large Language Models Skill Development	MD4282	Natural Language Processing Skill Development
MD4275	Mastering Microsoft Power BI	MD4283	Prompt Engineering Skills
MD4276	Generative AI Skills for Developers	MD4284	Essentials in Generative AI
MD4277	Career in Data Analysis	MD4285	Python in Finance
MD4278	Concepts of Data Visualization and Storytelling	MD4286	Understanding Quantum Computing
MD4279	AWS Certified Solutions Architect	MD4287	Foundational Maths for Machine Learning
MD4280	IT Security Specialist		
MD4281	Technical Program Management		

Semester-I

Title: Course Structure **FF No. 653**

Branch: Computer **Year:** BTech **A.Y.:** 2024-25 **Module: VIII (Internship - Internship)**

Subject No.	Subject Code	Subject Name	Teaching Scheme (Hrs/Week)			Examination Scheme								Credits
			Th	Lab	Tut.	CA		MSA		ESA			Total	
						HA (%)	LAB (%)	MSE (%)	PPT (%)	ESE (%)	GD (%)	Viva (%)		
S1	CS4236	Industry Internship	-	32	-	-	-	30	-	70	-	-	100	15
S1	CS4238	Research Internship	-	32	-	-	-	30	-	70	-	-	100	15
S2	CS4292	Design Thinking-7	-	-	1	-	-	-	-	100	-	-	100	1
		Total		**32**	1	-	-	**30**	-	**170**	-		**200**	**16**

Semester-II

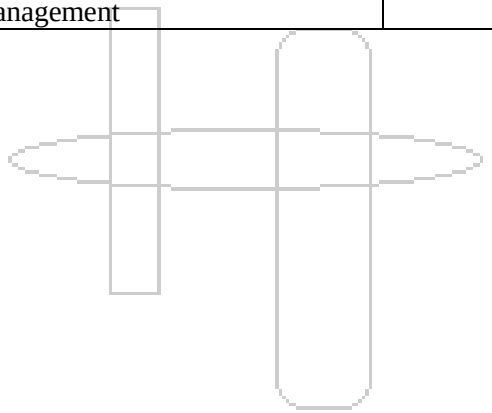
Title: Course Structure **FF No. 653**

Branch: Computer **Year:** BTech **A.Y.:** 2024-25 **Module: VIII (Course Work)**

Subject No.	Subject Code	Subject Name	Teaching Scheme (Hrs/Week)			Examination Scheme								Credits
			Th	Lab	Tut.	CA		MSA		ESA			Total	
						HA (%)	LAB (%)	MSE (%)	PPT (%)	ESE (%)	GD (%)	Viva (%)		
OE1	LinkedIn Learning *	LinkedIn Learning	0							100			100	2
OE2	CS4217	Human Computer Interaction	2	-	-	10	-	30	-	30	-	30	100	2
	CS4209	Parallel Computing	2	-	-	10	-	30	-	30	-	30	100	2
	CS4222	Image Processing	2	-	-	10	-	30	-	30	-	30	100	2
OE3	CS4275 Swayam	Introduction to Machine Learning	2	-	-	10	-	30	-	30	-	30	100	2
	CS4276 Swayam	Deep Learning	2	-	-	10	-	30	-	30	-	30	100	2
Major Project	CS4226	Major Project	0	20	-	-	-	30	-	70	-	-	100	10
		Total	**4**	**20**	-	**20**	-	**90**	-	**230**	-	**60**	**400**	**16**

LinkedIn Learning Courses*

Subject Code	Subject Name	Subject Code	Subject Name
MD4274	Large Language Models Skill Development	MD4282	Natural Language Processing Skill Development
MD4275	Mastering Microsoft Power BI	MD4283	Prompt Engineering Skills
MD4276	Generative AI Skills for Developers	MD4284	Essentials in Generative AI
MD4277	Career in Data Analysis	MD4285	Python in Finance
MD4278	Concepts of Data Visualization and Storytelling	MD4286	Understanding Quantum Computing
MD4279	AWS Certified Solutions Architect	MD4287	Foundational Maths for Machine Learning
MD4280	IT Security Specialist		
MD4281	Technical Program Management		



Semester-II

Title: Course Structure **FF No. 653**

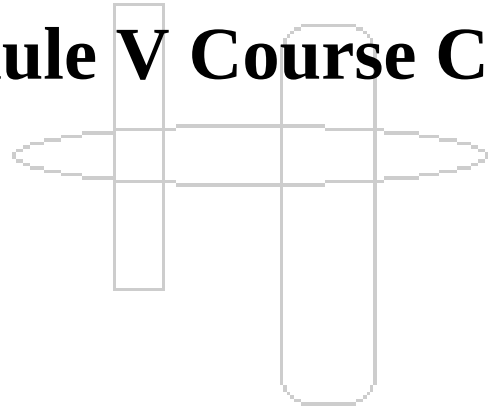
Branch: Computer **Year:** BTech **A.Y.:** 2024-25 **Module: VIII (Internship - Internship)**

Subject No.	Subject Code	Subject Name	Teaching Scheme (Hrs/Week)			Examination Scheme								Credits
			Th	Lab	Tut.	CA		MSA		ESA			Total	
						HA (%)	LAB (%)	MSE (%)	PPT (%)	ESE (%)	GD (%)	Viva (%)		
S1	CS4232	Industry Internship	-	32	-	-	-	30	-	70	-	-	100	16
S1	CS4202	Research Internship	-	32	-	-	-	30	-	70	-	-	100	16
		Total		**32**	1	-	-	**30**	-	**170**	-		**200**	**16**

INDEX

T. Y. B. Tech. Computer Engineering
AY 2025-26

Module V Course Content



Syllabus Template **FF No. : 654**

CS3052::Computer Networks

Credits: 4 **Teaching Scheme Theory: 2 Hours/Week**
Tutorial: 1 Hours/Week
Lab: 2 Hours/Week

Course Prerequisites: Operating System, Theory of Computer Science

Course Objectives:
1. To learn the data communication model, signal generation, data encoding, digital modulation and demodulation required for wired and wireless communication networks.
2. To learn the physical layer which includes transmission mediums, physical layer devices, transmission modes and topologies, performance issues for intranet and internetworks.
3. To learn multiple access schemes and wide area network connectivity for intranet and internetworks.
4. To learn IP protocol and routing algorithms for packet switching service framework used in intranet and internetworks.
5. To learn TCP and UDP protocol to provide quality of service over packet switching service framework used for intranet and internetworks.
6. To learn to select, design, develop, analyze and evaluate client server solutions for societal requirements at large.

Course Relevance:
The key technology of the information age is communications. Data communications and networking is a truly global area of study, both because the technology enables global communication over telephone lines and Internet. Data communication and networking is the backbone of all IT infrastructures in the world. These technologies and applications often emerge in communication within countries of countries and spread rapidly around the world.

Section 1: Topics/Contents

Unit-I Data Communication, Networking Fundamentals, Physical Layer **04 Hours**
Communication Model: Source, Transmitter, Transmission System, Receiver, Destination, Data Terminal Equipment (DTE), Data Communication Equipment (DCE). Transmission Configurations: Point to Point and Multipoint. Transmission Modes: Synchronous and Asynchronous. Transmission Methods: Serial and Parallel. Communication Modes: Simplex, Half Duplex, and Full Duplex. Line Coding: Unipolar NRZ, Polar NRZ, NRZ Inverted, Bipolar Encoding, Manchester Encoding, Differential Manchester Encoding. Modulation: Analog Modulation: Amplitude, Frequency, Phase.

Pulse Modulation Techniques: PCM, PAM, PWM, PPM. Digital Modulation: ASK, FSK, MSK, GMSK, PSK, BPSK, PSK, QAM, CPM, OFDM and multicarrier modulations.
Networking Fundamentals: Types of Computer Networks: LAN, MAN, WAN, PAN, Internet, internet and Intranet. Network Architectures: Client-Server; Peer To Peer. Network Architecture Modes: Infrastructure and Ad-hoc mode. Network Topologies: Mesh, Star and Hierarchical. Reference Models: OSI, TCP/IP. Design Issues for Layers. Is ATM still used? Is ISDN dying? Is Frame Relay outdated? Is SNA still present in the Market?
Physical Layer: Transmission Mediums: Air, Water, Vacuum, Coaxial, Cat5, Cat5e, Cat6, Cat6a, Cat7, Cat8, OFC - Single and Multicore. Networking Devices Wired and Wireless: NIC, Repeater, Bridge, Switch, Modem, Router, Gateways and Access Point.

Unit-II Logical Link Control **06 Hours**

Logical Link Control: Design Issues: Services to Network Layer, Framing, Error Control: Parity Bits, Hamming Codes and CRC. Flow Control Protocols: Unrestricted Simplex, Stop and Wait, Sliding Window Protocol, WAN Connectivity: PPP and HDLC. PPPoE, PPPoA. Is DOCSIS used in 2023? Do we use DSL line in 2023? Do we use coaxial cable in 2023? Is PPP still used?

Unit-III Medium Access Control **04 Hours**

Medium Access Control: Channel Allocation: Static and Dynamic, Multiple Access Protocols: Pure and Slotted ALOHA, CSMA, WDMA. Legacy Standard : 10 Mbps IEEE 802.3 Standard(Ethernet), Wiring Schemes and Frame Formats, CSMA/CD, Binary Exponential Back-off Algorithm. High Speed Ethernet Standards: Fast, Gigabit and 10Gigabit (Focus must be on Gigabit Networks). Wireless Standards: Radio Spectrum, Frequency Hopping (FHSS) and Direct Sequence (DSSS), IEEE 802.11a/b/g/n/ac, IEEE 802.15, IEEE 802.15.4 and IEEE 802.16 Standards, CSMA/CA, Introduction of Infrastructure and Data Processing Unit (IPU and DPU)

Section2: Topics/Contents

Unit-IV Network Layer **06 Hours**

Network Layer: Switching Techniques: Circuit, Message and Packet Switching. Logical Addressing: IPv4 and IPv6, Subnetting, NAT, CIDR. Network Layer Protocols: IP, ICMP, Routing Protocols: Distance Vector, Link State, and Path Vector. Routing in Internet: RIP, OSPF, BGP, Congestion control and QoS, MPLS, Mobile IP, Routing in MANET: AODV, DSR

Unit-V Transport Layer **04 Hours**

Transport Layer: Services: Berkeley Sockets, Addressing, Connection Establishment, Connection Release, Flow control and Buffering, Multiplexing. HTH Layer Protocols: TCP, TCP, TCP Timer management, UDP. Quality of Service: TCP Congestion Control. Traffic Shaping: AIMD. QUIC Protocol, Real Time Support Protocols: Real Time Transport protocol (RTP), Stream Control

Transmission Protocol (SCTP), Quality of Service (QoS), Differentiated services, TCP and UDP for Wireless

Unit-VI Application Layer **04 Hours**

Application Layer: Address Resolution: Domain Name System (DNS). WWW: Hyper Text Transfer Protocol (HTTP1.1/1.2/2.0) and HTTPS with SSL. Web Service. Email: SMTP, MIME, POP3 and Webmail. File Transfer: FTP, Dynamic Logical Addressing: Dynamic Host Control Protocol (DHCP), Custom packet generation, Design, development and evaluation of scalable enterprise application using communication and service frameworks.

List of Tutorials (13):

Unit-I Data Communication, N/w Fundamentals and Phy Layer

1) Examples and analysis of Encoding Methods: Unipolar NRZ, Polar NRZ, NRZ Inverted, Bipolar Encoding, Manchester Encoding, Differential Manchester Encoding

2) Line coding, Channel Encoding and modulations Techniques: used in IEEE 802.3 standard and its extensions, IEEE 802.11 standards and its extensions for 100 Mbps, 1 GbE, 1 Gbps, 2.5 Gbps, 5 Gbps, 10 Gbps, 25Gbps, 40 Gbps, 100 Gbps networks. Chanel Encodings in 3G, 4G and 5G Mobile Networks

Unit-II Logical Link Control

3) Examples on Network Performance parameters: RTT, Delay, Bandwidth, Throughput and efficiency

Unit-III Medium Access Control:

4) PHY and MAC Layer IEEE 802.3 Standards For Copper: Overview of 10 Mbps Ethernet, Fast Ethernet, GbE -Gigabit Ethernet, 2.5 Gigabit Ethernet, 5 Gigabit Ethernet, 10 Gigabit Ethernet, 25 Gigabit Ethernet,40 Gigabit Ethernet, 100 Gigabit Ethernet

5) PHY and MAC Layer IEEE 802.3 Standards For Optical Fiber: 100 Mbps Fast Ethernet, GbE - Gigabit Ethernet, 2.5 Gigabit Ethernet, 5 Gigabit Ethernet, 10 Gigabit Ethernet, 25 Gigabit Ethernet,40 Gigabit Ethernet, 100 Gigabit Ethernet

6) PHY and MAC Layer IEEE 802.11 Wireless LAN Standards: IEEE 802.11, Wi-Fi 1/IEEE 802.11a, Wi-Fi 2/IEEE 802.11b, Wi-Fi 3/IEEE 802.11g, Wi-Fi 4/IEEE 802.11n, Wi-Fi 5/IEEE 802.11ac, IEEE 802.11ad (WiGig), IEEE 802.11ah (HaLow), Wi-Fi 6/IEEE 802.11ax, Wi-Fi 6/IEEE 802.11ay, Wi-Fi 6/IEEE 802.11by, Wi-Fi 7/IEEE 802.11be

Unit-IV Network Layer:

7) Examples of Network Layer Logical Addressing

(a) Classful IP and CIDR: Subnetting, IP Prefixes

(b) NAT Mapping: Public to Private IP and Port Mapping

(c) Packet Delivery in Internetwork: Packets traversing through different sub-networks with different MTU and Speeds

(d) Packet Dropping Probabilities of Routers

8) Examples of Network Layer Routing

(a) Shortest Path and Spanning Tree
(b) Dijkstra's Algorithm
(c) Distance Vector Routing
(d) Link State Routing
(e) ECMP

Unit-V Transport Layer
9) Examples of Transport Layer
(a) TCP Connection Establishment: SYN and ACK, Normal Packets
(b) Flow Control: Calculating Optimal Size of Sliding Window
(c) Cumulative ACK scheme
(d) Smoothed RTT
(e) Slow Start and Additive Increase
Unit-VI Application Layer:
10) Examples of Application Layer
(a) DNS: URL Domain Processing
(b) Performance of HTTP1.0 and HTTP1.1
(c) CDN--------

List of Practical's (Minimum Six):

Unit-I Data Communication Networking Fundamentals and Physical Layer:
1) Write a program in C++/JAVA to implement - Unipolar NRZ, Polar NRZ, NRZ Inverted, Bipolar Encoding, Manchester Encoding and Differential Manchester Encoding.
2) Setting up small computer networks and Hands on networking commands:
Set up a small wired and wireless network of 2 to 4 computers using Hub/Switch/Access point. It includes installation of LAN Cards, Preparation of Cables/ Installation and Configuration of Access Point, Assigning unique IP addresses and use of ping utility. Hands on for network commands - ping, pathping, ipconfig/ifconfig, arp, netstat, nbtstat, nslookup, route, traceroute/tracert, nmap.
Unit-II and III MAC and Logical Link Layer
3) Write a program for error detection and correction for 7/8 bits ASCII codes using Hamming Codes or CRC. Demonstrate the packets captured traces using Wireshark Packet Analyzer Tool for peer to peer mode.(50% students will perform Hamming Code and others will perform CRC). Further extend it to real implementation of CRC over Ethernet standard.
4) Write a program to simulate Go back N and Selective Repeat Modes of Sliding Window Protocol in peer to peer mode and demonstrate the packets captured traces using Wireshark Packet Analyzer Tool for peer to peer mode. Further extend it to real implementation of Flow Control over TCP protocol.
Unit-IV Network Layer

5) Write a program to find the shortest path using Dijkstra Equation for Link State Routing Protocol which is used by Open Shortest Path First Protocol (OSPF) in the Internet for the network flow provided by instructor.

Unit-V Transport Layer

6) Write a program using TCP Berkeley socket primitives for wired /wireless network for following
a. Say Hello to Each other
b. File transfer
c. Calculator (Arithmetic)
d. Calculator (Trigonometry)
Demonstrate the packets captured traces using Wireshark/Fiddler for traffic analysis tool in peer to peer mode.
7) Write a program using UDP Berkeley Sockets for wired/wireless network to enable file transfer (Script, Text, Audio and Video one file each) between two machines. Demonstrate the packets captured traces using Wireshark/Fiddler for traffic analysis tool in peer to peer mode.

Unit-VI Application Layer
8) Understanding protocol stack of Intranet
a) Analyze packet formats of Ethernet, IP, TCP and UDP captured using Wireshark/Fiddler for traffic analysis tool in peer to peer mode for wired and wireless networks.
b) Use any tool for custom packet generation (Packet Sender Tool) or write your own code for packet generation and analyze the packets.
9) Develop a client-server using C++ or JAVA to demonstrate the behavior of HTTP1.0, HTTP1.1, HTTP1.2 and HTTP2.0 protocols along with all success and error messages. Use Firefox as client browser.

List of Course Project areas:
1. Simulation of modulation and demodulation for digital telephone lines
2. Simulation of modulation and demodulation for 100 Mbps Ethernet Network
3. Simulation of modulation and demodulation for Gigabit Ethernet Network
4. Simulation of modulation and demodulation for 10Gigabit Ethernet Networks
5. Simulation of modulation and demodulation for 3G for mobile networks
6. Simulation of modulation and demodulation for 4G mobile networks
7. Develop a tool fox for line encoding methods
8. Develop a tool fox for modulation and demodulation methods
9. Design and deploy TCP based Multithreaded HTTP client server for accessing student activity data in the institute.
10. Design and deploy TCP based Multithreaded FTP client server to share institute level notices.
11. Design and deploy UDP based Multithreaded TFTP client server for your class

12. Design and deploy TCP based Multithreaded SMTP and POP3 mail client server for your campus.
13. Design and deploy TCP based Multithreaded Chat client server for your class.
14. Design and deploy UDP based Multithreaded Chat client server for your class.
15. Design and deploy UDP based Multithreaded Audio Conferencing client server for computer engineering department.
16. Design and deploy UDP based Multithreaded Video Conferencing client server for computer department
17. Implementation of RIP/OSPF/BGP using Packet Tracer
18. Simulation of AODV routing protocol using Packet Tracer/ NS3/OMNet ---------

List of Group Discussion Areas:

1. Energy-Efficient Architectures For Communication System
2. Satellite Communication System
3. Data Communication in Software Defined Networks
4. Cognitive Radios for Future Communication Frameworks
5. Fast Ethernet (Encoding Framing, Modulation, Multiplexing, Diameter etc)
6. Gigabit Ethernet (Encoding Framing, Modulation, Multiplexing, Diameter etc)
7. 10G Ethernet (Encoding Framing, Modulation, Multiplexing, Diameter etc)
8. IEEE 802.11b protocol based on HR-DSSS for wireless physical layer standard
9. IEEE 802.11g protocol based on ERP-OFDM for wireless physical layer standard
10. IEEE 802.11n protocol based on HT-OFDM for wireless physical layer standard
11. IEEE 802.11ac protocol based on VHT-OFDM for wireless physical layer standard

List of Home Assignment Areas:

Design:

1. Design a communication framework for irrigation system
2. Design a communication framework for automated car
3. Design a communication framework for smart city applications
4. RIP Routing Protocol for Intranet in VIT campus
5. OSPF Routing Protocol for Internet on India
6. BGP Routing Protocol for Asia continent

Case Study:

1. WiTricity technology for industrial applications
2. Multiple access schemes implemented in 4G mobile networks
3. RFCs for wired TCP based reliable communication
4. RFCs for wireless TCP based reliable communication
5. RFCs for SSL Certificates

Blog:

1. Journey of line encoding methods
2. Journey of modulation techniques
3. Internet Logical Addressing
4 Internet Routing Protocols
5. Applications Layer Protocols

Survey

1. Analogy to digital transformations on communication systems
2. Routing protocols for MANET
3. IEEE 802.1 Physical layer standard for Internet
4. IEEE 802.15.4 standard for IoT applications
5. IEEE 802.11 Wireless Standards for Wi-Fi

Assessment Scheme: Suggest an Assessment scheme that is best suited for the course. Ensure 360 degree assessment and check if it covers all aspects of Bloom's Taxonomy.

Assessment scheme covers following aspects of Modified Blooms Taxonomy:

L2 Understanding, L3 Apply, L3 Design, L3 Apply, L4 Analyze and L5 Evaluate

Laboratory Continuous Assessment: 100 Marks converted to 10 Marks
Home Assignments: 100 Marks converted to 20 Marks
Course Project: End Semester Examination: 100 Marks converted to 20 Marks
Theory: End Semester Examination (Written): 60 Marks converted to 30 Marks
Comprehensive Viva Voce: End Semester Examination: 100 Marks converted to 20 Marks

Text Books: ***(As per IEEE format)***

1. Andrew S. Tanenbaum, "Computer Networks",5th Edition, PHI, ISBN 81-203-2175-8.
2. Kurose, Ross "Computer Networking a Top Down Approach Featuring the Internet", Pearson; 6th edition (March 5, 2012), ISBN-10: 0132856204
3. Frouzan B., "Data Communications and Networking", 5th edition, Tata McGraw- Hill, Publications, 2006

Reference Books: ***(As per IEEE format)***

1. Matthew S. Gast "802.11 Wireless Networks", O'Reilly publications; 2nd Edition.
2. C. Siva Ram Murthy and B. S. Manoj, "Ad Hoc Wireless Networks: Architectures and Protocols" Prentice Hall, 2004
3. Holger Karl and Andreas Willing, "Protocols and Architectures for Wireless Sensor Networks", Wiley, ISBN: 0-470-09510-5

MOOCs Links and additional reading material:

www.nptelvideos.in, www.coursera.com, www.udemy.com

Course Outcomes:

The student will be able to –

1. Select line encoding, modulation, topology, essential components of physical layer, data transmission rates to design computer networks.
2. Estimate reliability issues based on error control, flow control and pipelining by using bandwidth, latency, throughput and efficiency.
3. Propose mechanisms for server channel allocation in wired and wireless computer networks
4. Develop Client-Server architectures and prototypes by the means of correct standards, protocols and technologies
5. Analyze data flow between peer to peer in an IP network using Application, Transport and Network Layer Protocols
6. Compare sustainable engineering practice indicating the scientific purpose and utility of communication frameworks and standards.

CO-PO Map:

CO	Program Outcomes (PO)												PSO			
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3	PSO 4
1	3	**2**			**2**									3		
2	3	2	**2**		2									3		
3	2	3	3	**2**		2	3							3		2
4	3	3	3	2	1	**2**	3	3	3				3	3	2	
5	3	3		2	3							**3**		3		
6	3	3		2						3		3		**3**		
Avg	**2.84**	**2.67**	**2.67**	**2.0**	**2**	**2**	**3**	**3**	**3**	**3**	**0**	**3**	**3**	**3**	**2**	**2**

Attainment Levels: 1, 5, 3, 4, 2, 4

CO Attainment levels:

Weights for attainment levels: L1 - Easy-0.75 L2 - Comfortatble-0.7 L3 – Medium – 0.65
L4 – Somewhat difficult – 0.6 L5 – Difficult – 0.55

CO1 – L1, CO2 – L5, CO3 – L3, CO4 – L4, CO5 – L2 and CO6 – L4

Future Course Mapping:

High Speed Networks, Wireless Networks, Mobile Networks, Network Security, Cyber Security

Job Mapping:

Network Engineer, Network Stack Developers, Application Developer

Syllabus Template **FF No. :654**

CS3205::Design and Analysis of Algorithms

Credits: 4..... **Teaching Scheme Theory: 2 Hours/Week**

Tutorial: 1 Hours/Week

Lab: 2 Hours/Week

Course Prerequisites: Basic courses on programming, data structures, discrete structures, theory of computing.

Course Objectives:

1. Students will gain understanding of asymptotic notations and will be able to apply suitable mathematical techniques to find asymptotic time and space complexities of algorithms.
2. Students will develop the ability to formulate computational problems in the abstract and mathematically precise manner.
3. Student will gain understanding of different algorithm design paradigms such as divide and conquer, dynamic programming, greedy, backtracking and will apply suitable paradigm for designing algorithms for computational problems.
4. Students will develop understanding of notions of NP-hardness and NP-completeness and their relationship with the intractability of decision problems.
5. Students will design randomized, approximation algorithms for some computational problems.
6. Students will be able to incorporate algorithm design principles, data structures and provide efficient solutions for complex computational problems.

Course Relevance:

This is a foundational course for Computer science and Engineering. This course develops algorithmic thinking capability of students. Designing algorithms using suitable paradigm and analysing the algorithms for computational problems has a high relevance in all domains where computer science plays a crucial role (equally in Industry as well as research). This course is also an essential pre-requisite for advanced domain specific algorithmic courses such as Algorithmic Graph Theory, Algorithmic Number Theory, Computational Geometry, Motion planning and Robotics, etc, to give a few examples.Once the student gains expertise in Algorithm design and in general gains ability of Algorithmic thinking, it facilitates in systematic study of any other domain (in computer science or otherwise) which demands logical thinking.This course is also relevant for students who want to pursue research career in theory of computing, computational complexity theory, advanced algorithmic research.

Section 1: Topics/Contents

Unit-I Basic introduction and time and space complexity analysis [4 Hours]

Asymptotic notations (Big Oh, small oh, Big Omega, Theta notations). Best case, average case, and worst-case time and space complexity of algorithms. Overview of searching, sorting algorithms.Cache

optimization,Adversary lower bounds (for the comparison-based sorting algorithms). Using Recurrence relations and Mathematical Induction to get asymptotic bounds on time complexity. Master's theorem and applications.

Unit-II Divide and Conquer [4 Hours]

General strategy, Application of divide and conquer for solution of some computational problems like:Quick sort, Merge sort, Finding a majority element, Order statistics (randomized and deterministic algorithms), Efficient algorithms for Integer arithmetic (Euclid's algorithm, Karatsuba's algorithm for integer multiplication, fast exponentiation).

Unit-III Dynamic Programming [6 Hours]

General strategy, Application of dynamic programming for solution of some computational problems like:computing Fibonacci numbers, binomial coefficients, Matrix Chain multiplication,Coin change problem, 0-1 Knapsack, Traveling Salesperson Problem, Optimal Binary Search Tree construction, Shortest paths in directed acyclic graphs, All pair shortest path algorithm, Longest increasing subsequence problem, Largest independent set for trees.

Section2:Topics/Contents

Unit-IV Greedy and Backtracking strategy [4 Hours]

Greedy: General strategy, Analysis and correctness proof of minimum spanning tree and shortest path algorithms, fractional knapsack problem, conflict free scheduling, Huffman coding.

Backtracking: General strategy, backtracking strategy for some problems like:n-queen problem, graph coloring, subset sum problem, vertex cover, independent set, solution of SUDOKU puzzle.

(Note: for all the strategies discussed in Units II, III, and IV the student is expected to understand the essence of the strategies and not just the specific applications and should be able to apply suitable strategies for fresh problems)

Unit-V Introduction to complexity classes and NP-completeness [6 Hours]

Complexity classes P, NP, coNP, and their interrelation, Notion of polynomial time many one reductions reduction, Notion of NP-hardness and NP-completeness, Cook-Levin theorem and implication to P versus NP question, NP-hardness of halting problem. NP-Complete problems (some selected examples from - Satisfiability problem, Circuit-SAT, 3-CNF SAT, vertex cover problem, independent set problem, clique problem, Hamiltonian-circuit problem, subset sum problem, Integer Linear Programming (ILP))**.** Brief introduction to Linear Programming and modeling NP-complete problems using ILP.

Unit-VI Introduction to Randomized and Approximation algorithms [4 Hours]

Introduction to randomness in computation, Las-Vegas and Monte-Carlo algorithms, Abundance of witnesses/solutions and application of randomization, solving SAT for formulas with "many" satisfying assignments, randomized quick sort, Karger's Min-cut algorithm, coupon collector problem,

Introduction to Approximation algorithms for NP-optimization problems, Approximation algorithm for Vertex Cover, metric Traveling-Sales-Person Problem (metric-TSP), Hardness of approximation for TSP.Introduction to quantum computation.

List of Tutorials (12):

1. Problem solving based on asymptotic notations, solution of recurrences.
2. Proving correctness of algorithms: some techniques
3. Problem solving based on Divide and Conquer strategy (Binary search interesting applications, counting inversions)
4. Advanced problem solving based on Divide and Conquer strategy (Discrete Ham-Sandwich theorem, efficient algorithm for Josephus problem)
5. Problem solving based on Dynamic Programming strategy (Largest sum contiguous block and generalizations, Optimal binary search tree (OBST) construction)
6. Advanced problem solving based on Dynamic Programming strategy (Winning strategy for two player games, Variants of shortest path algorithms)
7. Problem solving based on Greedy strategy with emphasis on proof of correctness.
8. Problem solving based on Backtracking strategy.
9. reducing NP problems to Integer Linear Programming**.**
10. Problem solving based on complexity classes, NP-completeness.
11. Problem solving based on Randomized Algorithms
12. Problem solving based on Approximation Algorithms

List of Practical's (Minimum Six):

1. Assignment based on some simple coding problems on numbers, graphs, matrices.
2. Assignment based on Divide and Conquer strategy (e.g., majority element search, finding kth rank element in an array)
3. Assignment based on Divide and Conquer strategy (e.g., efficient algorithm for Josephus problem using recurrence relations, fast modular exponentiation)
4. Assignment based on Dynamic Programming strategy (e.g., Matrix chain multiplication, longest increasing subsequence)
5. Assignment based on Dynamic Programming strategy (e.g., All pair shortest path, Traveling Salesperson problem)
6. Assignment based on Greedy strategy (e.g., Huffman encoding)
7. Assignment based on Backtracking (e.g., graph coloring, n-queen problem)
8. Assignment based on analysis of quick sort (deterministic and randomized variant)
9. Assignment based on Las-Vegas and Monte-Carlo algorithm for majority element search.
10. Assignment based on factor-2 approximation algorithm for metric-TSP.

List of Course Project topics:

1. Applications of A* algorithm in gaming
2. Pac-Man game

3. File compression techniques
4. Solution of Maze (comparing the backtracking based solution and Dijkstra's algorithm)
5. Different exact and approximation algorithms for Travelling-Sales-Person Problem
6. Creation of Maze using backtracking
7. Knight tour algorithms
8. Network flow optimization and maximum matching
9. AI for different games such as minesweeper, shooting games, Hex, connect-4, sokoban, etc
10. SUDOKU solver
11. Graph theoretic algorithms
12. Computational Geometry Algorithms
13. AKS primality testing
14. Algorithms for factoring large integers.
15. Randomized algorithms for primality testing (Miller-Rabin, Solovay-Strassen)
16. Slider puzzle game

List of Course Seminar Topics:

1. Divide and Conquer Vs Dynamic Programming
2. Greedy strategy
3. NP-hardness
4. Backtracking strategy
5. Dynamic Programming Vs Greedy
6. Computational Complexity
7. Philosophical relevance of P Vs NP question
8. Complexity classes
9. Space complexity
10. Compression Techniques
11. Real world applications of Graph theoretic algorithms
12. Approximation algorithms
13. Hardness of approximation
14. Pseudorandom number generators

List of Home AssignmentsTopics:

List of Design Based Home Assignments:

1. Problem solving based on Divide and Conquer strategy
2. Problem solving based on Dynamic Programming strategy
3. Problem solving based on Greedy strategy
4. Problem solving based on Backtracking strategy
5. Problems on Randomized Algorithms
6. Problems on Approximation Algorithms
7. Problems on NP completeness

List of Case Study Based Home Assignments:

1. AKS primality test
2. Quadratic sieve factoring algorithm

3. Huffman Encoding, LZW encoding
4. Network flow optimization algorithms
5. Approximation algorithms for TSP
6. Cook-Levin theorem and its relationship with intractability of computational problems
7. Sorting techniques

List ofBlog Based Home Assignment:

1. Approximation Algorithms
2. Randomized Algorithms
3. Computational Geometry Algorithms
4. Number Theoretic Algorithms
5. Graph Theoretic Algorithms
6. P Vs NP Problem
7. Complexity classes
8. Greedy Algorithms
9. Divide and Conquer Vs Dynamic Programming

List of Survey Based Home Assignments:

1. Primality Testing Algorithms
2. Integer Factoring Algorithms
3. NP-complete problems
4. Compression Techniques
5. Shortest Path Algorithms
6. Algorithms for finding Minimum Weight Spanning Tree

Assessment Scheme: Suggest an Assessment scheme that is best suited for the course. Ensure 360 degree assessment and check if it covers all aspects of Bloom's Taxonomy.

Laboratory Continuous Assessment: 100 Marks converted to 10 Marks
Course Project: End Semester Examination: 100 Marks converted to 20 Marks
Presentation: End Semester Examination: 100 Marks converted to 20 Marks
Theory: End Semester Examination (Written): 60 Marks converted to 30 Marks
Comprehensive Viva Voce: End Semester Examination: 100 Marks converted to 20 Marks

Text Books: ***(As per IEEE format)***

1. *Cormen, Leiserson, Rivest and Stein "Introduction to Algorithms" ,PHI 3nd edition, 2009. ISBN 81-203-2141-*
2. *Jon Kleinberg, Eva Tardos "Algorithm Design", Pearson, 1st edition, 2005. ISBN 978-81-317-0310-6*
Dasgupta, Papadimitriu, Vazirani "Algorithms" McGraw-Hill Education; 1 edition (September 13, 2006), ISBN-10: 9780073523408, ISBN-13: 978-0073523408

Reference Books: ***(As per IEEE format)***

1. *Motwani, Raghavan "Randomized Algorithms", Cambridge University Press; 1 edition (August 25, 1995), ISBN-10: 0521474655, ISBN-13: 978-0521474658*
2. *Vazirani, "Approximation Algorithms", Springer (December 8, 2010), ISBN-10: 3642084699, ISBN-13: 978-3642084690Gerd Keiser, MC Graw Hill International edition, optical fiber communication , third edition*

MOOCs Links and additional reading material:

1. www.nptelvideos.in

Course Outcomes:

The student will be able to –

1. To formulate computational problems in abstract and mathematically precise manner
2. To design efficient algorithms for computational problems using appropriate algorithmic paradigm
3. To analyze asymptotic complexity of the algorithm for a complex computational problem using suitable mathematical techniques.
4. To establish NP-completeness of some decision problems, grasp the significance of the notion of NP-completeness and its relationship with intractability of the decision problems.
5. To understand significance of randomness, approximability in computation and design randomized algorithms for simple computational problems and design efficient approximation algorithms for standard NP-optimization problems.
6. To incorporate appropriate data structures, algorithmic paradigms to craft innovative scientific solutions for complex computing problems.

CO-PO Map:

CO	Program Outcomes (PO)												PSO			
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3	PSO 4
1	2	3	2	1			2					2	3		2	
2	2	3	3	1			2					2	3		2	
3	2	3	2	1			2					2	3		2	
4	2	3	3	1			2					2	3		2	
5	2	3	3	1			2					2	3		2	
6	2	3	3	1			2					2	3		2	
Avg	**2**	**3**	**2.66**	**1.0**			**2.0**					**2.0**	**3**		**2**	

CO Attainment levels:

Weights for attainment levels: L1 - Easy-0.75 L2 - Comfortatble-0.7 L3 – Medium – 0.65
L4 – Somewhat difficult – 0.6 L5 – Difficult – 0.55

CO1 – L1, CO2 – L3, CO3 – L2, CO4 – L3, CO5 – L4 and CO6 – L5

Future Course Mapping:

Advanced Algorithms, Computational Complexity, Computational Geometry, Algorithmic Number Theory, Algorithmic Graph Theory

Job Mapping:

Algorithm design lie at heart of any Computer Science/Engineering application. Once the student gains expertise in Algorithm design and in general gains ability of Algorithmic thinking, it facilitates in systematic studying any other domain (in computer science or otherwise) which demands logical thinking. Algorithm design is an essential component of any job based on programming. All Industries in computer Engineering always look for a strong knowledge in Algorithm design and Data structures. If student wants to pursue higher education/ research in Computer Science, this course is must.

Syllabus Template **FF No. : 654**

CS3215::Web Technology

Credits: 4 **Teaching Scheme Theory: 2 Hours/Week**
Tutorial: 1 Hours/Week
Lab: 2 Hours/Week

Course Prerequisites: Computer Networks

Course Objectives:

1. To describe most commonly used HTML5 and CSS3 tags and attributes for website development.
2. To associate event handling with HTML5 forms and CSS3 using JavaScript as a front-end technology for website development.
3. To extend HTML5 and CSS3 and JavaScript front end technologies with PHP and MySQL as a server side and backend technologies for website development.
4. To simplify website development using REST API and Spring Boot as server-side technologies.
5. To build single page applications using REACT as a reusable UI component technology as client-side technology
6. To assemble REACT as a front-end technology and Node JS as a server-side technology to develop enterprise applications

Course Relevance:

The key technology of the information age is global communication. Web technology is a truly global area of study as it enables global communication with the help of web sites. Web technologies are the backbone of all IT infrastructures and their applications in the world. These technologies and applications often emerge in communication within countries of countries and spread rapidly around the world. The main objective of the course is present the basic web technology concepts that are required for developing web applications. The key technology components are descriptive languages, server-side program elements and client-side program elements. In addition, the course gives specific contents that are beneficial for developing web-based solutions, like relational data-base communication basics and information security principles and approaches. Most of the jobs available in the IT industries are web technology related.

Section 1: Topics/Contents

Unit-I Front End Tools **04 Hours**

Introduction: Internet and WWW, Web site planning and design issues. HTML5: structure of html document, HTML elements: headings, paragraphs, line break, styles, colors, fonts, links, frames, lists, tables, images and forms, CSS, Bootstrap, XML, JSON.

Unit-II Client-Side Technologies **06 Hours**

JavaScript: Overview of JavaScript, Data types, Control Structures, Arrays, Functions and Scopes, HTML5 forms Validation, Objects in JS, flex, DOM: DOM levels, DOM Objects and their properties and methods, Manipulating DOM. JQuery: Introduction to JQuery, Loading JQuery, selecting elements, changing styles, creating elements, appending elements, removing elements, handling events, Introduction to POSTMAN usages.

Unit-III Server-Side Technologies **04 Hours**

PHP: Introduction to PHP, Features, sample code, PHP script working, PHP syntax, conditions & Loops, Functions, String manipulation, Arrays & Functions, Form handling, Cookies & Sessions, File Handling, Exception Handling, E-mail validations, MySQL with PHP, Laravel - The PHP Framework For Web Artisans. Introduction and usage of web3.0

Section2: Topics/Contents

Unit-IV Spring Boot **06 Hours**

Spring Framework, Spring Boot Framework, Installing Spring Boot, Build Tool Maven/Gradle/Ant, Core Features, Spring Security, Web Applications, JPA for database connectivity, working with SQL and NoSQL, Messaging, Testing, Deploying Spring Boot Applications, Monitoring and Testing. POSTMAN Tool for API testing.

Unit-V React **04 Hours**

Introduction to React, React component, JSX, Render function, Component API, Component lifecycle, State, Props, Mixins, Component composition, Pass data from parent to child, Pass data from child to parent, Component styling, Forms, Events, Refs, Keys, Router, Flux, Redux[4 Hrs]

Unit-VI Node JS **04 Hours**

Introduction to Node JS, Installation of Node JS, Node JS Modules, Node Package Manager (NPM), Creating Web server, File System, Express JS, Serving Static Resources, Database connectivity.[4 Hrs]

List of Tutorials (13):

----- Django

1. Django - Environment, Creating a Project,
2. Apps Life Cycle, Admin Interface,
3. Creating Views, URL Mapping, Template System, Models,
4. Page Redirection, Sending E-mails,
5. Generic Views, Form Processing,
6. File Uploading, Apache Setup, Cookies Handling,
7. Sessions, Caching,
8. RSS

9. AJAX
10. -------- Introduction to JFrog Artifactory

    JFrog Artifactory is a DevOps solution for hosting, managing, and distributing binaries and artifacts. Any type of software in binary form – such as application installers, container images, libraries, configuration files, etc. – can be curated, secured, stored, and delivered using Artifactory. The name "Artifactory" reflects the fact that it can host any type of "artifact" needed in your software development "factory."

List of Practical's (Minimum Six):

1. Design and develop a responsive web page for your CV using multiple column layouts having video background. You can make the use of bootstrap as well as jQuery.

2. Design and develop a website using toggleable or dynamic tabs or pills with bootstrap and JQuery to show the relevance of SDP, EDI, DT and Course projects in VIT.

3. Design and develop a website to demonstrate (a) searching and sorting array for integer elements using JavaScript (b) array for named entities using JavaScript. You can make the use of bootstrap as well as jQuery.

4. Design and develop a responsive website to calculate Electricity bill using Django/Springboot/Node JS/ PHP. Condition for first 50 units – Rs. 3.50/unit, for next 100 units – Rs. 4.00/unit, for next 100 units – Rs. 5.20/unit and for units above 250 – Rs. 6.50/unit. You can make the use of bootstrap as well as jQuery.

5. Design and develop a responsive website to calculate Electricity bill using Django/Springboot/Node JS/ PHP. Condition for first 50 units – Rs. 3.50/unit, for next 100 units – Rs. 4.00/unit, for next 100 units – Rs. 5.20/unit and for units above 250 – Rs. 6.50/unit. You can make the use of bootstrap as well as jQuery.

6. Design and develop a responsive website to prepare one semester result of VIT students using REACT, Django/Springboot/Node JS/ PHP and MySQL/ MongoDB/Oracle. Take any four subjects with MSE Marks (30%) ESE Marks (70%).

7. Design and develop a responsive website to prepare one semester result of VIT students using JavaScript, Django/Springboot/Node JS/ PHP and MySQL/MongoDB/Oracle. Take any four subjects with MSE Marks (30%) ESE Marks (70%).

8. Design and develop a responsive website for an online book store using Django/Springboot/Node JS/ PHP and MySQL/ MongoDB/Oracle having 1) Home Page2) Login Page 3) Catalogue Page: 4) Registration Page: (database)

9.Design and develop a responsive website for an online book store using REACT, Django/Springboot/Node JS/ PHP and MySQL/ MongoDB/Oracle having 1) Home Page2) Login Page 3) Catalogue Page: 4) Registration Page: (database)

List of Course Project areas:

1. Develop a responsive web application for Student Grievance System
2. Develop a responsive web application for Workflow Management System for MNC
3. Develop a responsive Gaming Website
4. Develop a responsive web application to help farmers to solve their farming problems
5. Develop a responsive web application for GST Billing Software for Small Business
6. Develop a responsive web application for online Crime Reporting System using PHP
7. Develop a responsive web application for online College Voting System
8. Develop a responsive web application for online Loan Processing System for Farmers.
9. Develop a responsive web application for restaurant food order management
10. Develop a responsive web application for e-book shop
11. Develop a responsive web application for on-line music store
12. Develop a responsive web application for guest visiting management to your society
13. Develop a responsive web application for web search engine

---

Assessment Scheme: Suggest an Assessment scheme that is best suited for the course. Ensure 360 degree assessment and check if it covers all aspects of Bloom's Taxonomy.

Assessment scheme covers following aspects of Modified Blooms Taxonomy:

L2 Understanding, L3 Apply, L3 Design, L3 Apply, L4 Analyze and L5 Evaluate

Laboratory Continuous Assessment: 100 Marks converted to 10 Marks
Course Project: End Semester Examination: 100 Marks converted to 20 Marks
Laboratory Practical: End Semester Examination: 100 Marks converted to 50 Marks
Comprehensive Viva Voce: End Semester Examination: 100 Marks converted to 20 Marks

Text Books: ***(As per IEEE format)***

1. *Kumar, A., Web technologies, CRC press, 2019*
2. *Gupta, R., Internet & Web Technologies, Engineering Handbook, 2019*

3. *Martin, M.G., Programming for Beginners: 6 Books in 1 – Swift+PHP+Java+Javascript+Html+CSS: Basic Fundamental Guide for Beginners, independently published, 2018*
4. *Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5, O'Reilly Media; 5th edition, 2018*
5. *Kohli, S., Web Technologies, PPB Publications, 2015*
6. *Adam Bretz& Colin J Ihrig, "Full Stack Javascript Development with MEAN", SPD, First Edition 2015, Indian Reprint September 2015*
7. *GiulioZambon, "Beginning JSP, JSF and Tomcat", Apress Publication, Second Edition, 2013*
8. *Jeremy McPeak& Paul Wilton," Beginning JavaScript", Wrox Publication, Fifth Edition, 2015*
9. *Jeffrey C.Jackson, "Web Technologies: A Computer Science Perspective", Second Edition, Pearson Education, 2007, ISBN 978-0131856035.*
10. *Robert W. Sebesta: Programming the World Wide Web, 4th Edition, Pearson education, 2008*

Reference Books: ***(As per IEEE format)***

1. *Marty Hall, Larry Brown,"Core Web Programming", Second Edition, Pearson Education, 2001, ISBN 978-0130897930.*
2. *H.M. Deitel, P.J. Deitel and A.B. Goldberg, "Internet & World Wide Web How To Program", Third Edition, Pearson Education, 2006, ISBN 978-0131752429.*
3. *Chris Bates: Web Programming Building Internet Applications, 3rd Edition, Wiley India, 2006.*
4. *Xue Bai et al: The web Warrior Guide to Web Programming, Thomson, 2003*

MOOCs Links and additional reading material:

1. www.w3.org
2. HTML, The Complete Reference
3. www.htmlref.com
4. w3schools.org
5. php.net/ https://jquery.com/
6. developer.mozilla.org/en-US/docs/AJAX
7. www.tutorialspoint.com/css/
8. PHP: Data Structures - Manual -----
9. docs.spring.io/spring-boot/docs/current/reference/html/
10. nodejs.org/en
11. react.dev

Course Outcomes:

The student will be able to –

1. Create front end web pages using HTML5 and CSS3 tags and attributes
2. Provide validation mechanism and event handling in a website using JavaScript as a front end technology
3. Integrate front end with server side and backend technologies for commercial websites using PHP and MySQL
4. Write Web API/RESTful API application programming interface to communicate with Springboot as a server side technology.
5. Build single page applications using REACT as a reusable UI component technology as client side technology and Springboot and Node JS as server side technologies

6. Design and develop three tier enterprise application using client side, server side and back end technologies

CO-PO Map:

	Program Outcomes (PO)												PSO			
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4
CS3215.1	2	2	2									3	1		2	
CS3215.2	2	2	2									3	1		2	
CS3215.3	3	3	3	1	2	2	2	2				3	1		3	2
CS3215.4	3	3	3	1	2	2	2	2				3	1		3	2
CS3215.5	3	3	3	1	2	2	2	2			2	3	1		3	2
CS3215.6	3	3	3	1	2	2	2	2	2	3	2	3	1		3	2
Average	**2.66**	**2.66**	**2.66**	**1.0**	**2**	**2**	**2**	**2**	**2.0**	**3.0**	**2.0**	**3.0**	**1.0**		**2.66**	**2.0**

CO Attainment levels:

Weights for attainment levels: L1 - Easy-0.75 L2 - Comfortatble-0.7 L3 – Medium – 0.65
L4 – Somewhat difficult – 0.6 L5 – Difficult – 0.55

CO1 – L1, CO2 – L2, CO3 – L3, CO4 – L4, CO5 – L4 and CO6 – L5

Future Course Mapping:

Cloud Computing, Distributed System, Mobile Application Development

Job Mapping:

Software Engineer, Web Developer, IT Engineer, UI Developer

Syllabus Draft **FF No. : 654**

CS3061: Software Modeling and Design (SMD)

Credits: 04 **Teaching Scheme Theory: 2 Hours/Week**
Tutorial: 1 Hours/Week
Lab: 2 Hours/Week

Course Prerequisites:

Course Objectives:

1. To summarize capabilities and impact of software development process models and justify process maturity through application of Software Engineering principles and practices,
2. To differentiate feasible and competing system requirements, indicating correct real world problem scope and preparing stepwise system conceptual model,
3. To formulate system specifications by analyzing user-level tasks and compose software artifacts using agile principles, practices and scrum framework,
4. To compose system analysis and design specifications using UML diagrams,
5. To design a system architecture and map it with a suitable architectural style,
6. To comprehend the nature of design patterns and apply these patterns in system design.

Course Relevance:

Given that Software Engineering is built upon the foundations of Computer Science as well as Computer Engineering, a Software Engineering curriculum can be focused on two perspectives - a Computer Science-first or Software Engineering-first perspective. Software engineering spans the entire software lifecycle. It involves creating high-quality and reliable programs in a systematic, controlled, and efficient manner using formal methods for specification, analysis, design and evaluation of proposed systems. It requires suitable software development techniques and processes that successfully scale to large applications, which should satisfy timing, size, and security requirements all within acceptable application/project budgets and deadlines. For these reasons, Software Engineering requires both the analytical and descriptive tools and techniques developed in Computer Science and the rigor that the Computer Engineering discipline brings to the reliability and trustworthiness of the systems that software developers design and implement, while working as a cohesive team.

Section 1: Topics / Contents

Unit-I Software Engineering Paradigms: **05 Hours**

Process Models: Code-and-Fix Model, Waterfall Model, Rapid Application Development Model, Incremental Model, Evolutionary Model and Others.

Unit-II Requirements Engineering: **05 Hours**

Requirements Engineering Tasks, Requirement Elicitation Techniques, Functional, Non- Functional and Domain Requirements, Requirements Characteristics, Eliminating Requirement Ambiguities, Conflict Identification and Resolution, Requirement Qualities, Requirement Specification, System Scope Determination and Feasibility Study.

Unit-III Agile Methodology: **04 Hours**

Landscape of Agile and Planned Methods, Definition - Scrum, Scrum Origins, Scrum Framework, Agile Principles, Sprints, Requirements, User Stories, Product Backlog, Roles: Product Owner, Scrum Master, Development Team, Managers, Scrum Team Structures, Scrum Planning.

Section 2: Topics/Contents

Unit-IV Static and Dynamic Interaction Modeling: **05 Hours**

Static Behavior: Use Case, Use Case Diagram, Class Diagram, Component Diagram, Deployment Diagram, Dynamic Behavior: Sequence Diagram, Collaboration Diagram, Activity Diagram, Communication Diagram, Interaction Diagrams.

Unit-V Software Architecture Design: **05 Hours**

Design Model, Design Qualities, Characteristics of Design Activities, Design Principles, Cohesion and Coupling, Software Architecture Vs Software Design, Software Reuse, Design Heuristics, Layered Architecture, Client-Server Architecture, Pipe-Filter Architecture, Model-View Controller Architecture.

Unit-VI Design Patterns: **04 Hours**

Definition, Describing Design Pattern,
Creational Patterns: Abstract Factory, Builder, Factory Method, Prototype
Structural Patterns: Adapter, Bridge, Composite, Decorator, Façade,
Behavioral Patterns: Chain of Responsibility, Command, Interpreter.

List of Tutorials:

1. Requirement Engineering,
2. System Requirement Specification,
3. Scrum Artifacts,
4. User Stories and Use Cases,
5. Product Backlog Development,
6. Burn-up and Burn-down Chart Development and Management,
7. Software System Analysis and Design: UML Static Diagram,
8. Software System Analysis and Design: UML Dynamic Diagram,
9. Software Architecture Design,
10. Use of Design Patterns,
11. Software Testing,

12. Automated Testing,
13. Project Management Techniques.

List of Practicals (Minimum SIX):

1. To prepare a Statement Of Work (SOW) document, which addresses the vision, goals and objectives of the real-world problem.
2. To prepare a Software Requirement Specification (SRS) document, based on several types of system requirements, such as functional and non-functional requirements.
3. To document a product backlog for the project aimed at maintaining a prioritized queue of project requirements.
4. To develop a Sprint-plan and Sprint-design indicating detailed activity planner accommodating user story points.
5. To prepare Class Collaboration-Responsibility (CRC) cards for the Conceptual classes traced from the system analysis phase.
6. To develop a static structure of the target system with a Class Diagram using all components of it.
7. To decompose and organize the problem domain area into broad subject areas and identify the use cases to show them in a Use Case Diagram.
8. To depict the dynamic behavior of the target system using Sequence Diagram. The Sequence diagram should be based on the scenarios generated by the inter-object Communication.
9. To depict the dynamic behavior using a detailed Activity Diagram.
10. To prepare an Architecture Diagram with appropriate design patterns. Suitable Architectural Styles shall be selected and the structural elements shall be well-documented.

List of Course Projects:

1. Automated Parking Lot Identifier,
2. Healthcare Software,
3. Financial Application,
4. Appraisal System,
5. Smart Project Administrative System,
6. Translator for Agriculture System,
7. Development of Applications using Agile Methodology,
8. Development of SMART Mobile Applications,
9. Graphics-based Password Identification System
10. System Security Application

List of Course Seminar Topics:

1. Mobile Apps and App Store Analysis,
2. Automated Reasoning Techniques,
3. Autonomic and Self-Adaptive System,
4. Component-based Software Engineering,

5. Computer-Supported Cooperative Work (CSCW),
6. Configuration Management and Deployment,
7. Crowd-Sourced Software Engineering,
8. Cyber-Physical System,
9. Data-driven Software Engineering,
10. Dependability, Safety and Reliability.

List of Home Assignments:

Design:

1. Software Visualization
2. Specification and Modeling Languages
3. Tools and Environments
4. Traceability
5. Ubiquitous and Pervasive Software Systems

Case Study:

1. Software Economics and Metrics
2. Machine Learning in Software Engineering
3. Software Evolution and Maintenance
4. Software Modeling and Design
5. Software Product Lines

Blog

1. Mining Software Engineering Repositories
2. Model-driven Engineering
3. Parallel, Distributed and Concurrent systems
4. Recommendation Systems
5. Refactoring

Surveys

1. Reverse Engineering
2. Safety-Critical Systems
3. Security, Privacy and Trust
4. Software Architecture
5. Software Reuse
6. Software Testing

Assessment Scheme: Suggest an Assessment scheme that is best suited for the course. Ensure 360 degree assessment and check if it covers all aspects of Bloom's Taxonomy.

Laboratory Continuous Assessment: 100 Marks converted to 10 Marks
Course Project: End Semester Examination: 100 Marks converted to 20 Marks
Group Discussion: End Semester Examination: 100 Marks converted to 20 Marks
Theory: End Semester Examination (MCQ): 60 Marks converted to 30 Marks
Comprehensive Viva Voce: End Semester Examination: 100 Marks converted to 20 Marks

Text Books: ***(As per IEEE format)***

1. *Ian Sommerville, 'Software Engineering',Pearson, 10th Edition, 2017, ISBN-13: 978-9332582699.*
2. Kenneth Rubin, 'Essential SCRUM: A Practical Guide To The Most Popular Agile Process', Addison-Wesley, 2012, ISBN-13: 978-0-13-704329-3.
3. Tom Pender, 'UML Bible', John Wiley & Sons, 2003, ISBN - 0764526049

Reference Books: ***(As per IEEE format)***

1. *SorenLauesen, 'Software Requirements: Styles and Techniques*, *Addison Wesley, 2002, ISBN 0201745704.*
2. Dean Leffingwell, 'Agile Software Requirements', Addison-Wesley, 2011, ISBN-13: 978-0-321-63584-6.
3. Grady Booch, James Rambaugh, Ivar Jacobson, 'Unified Modeling Language User's Guide', 2nd Edition, Addison-Wesley 2005, ISBN – 0321267974.
4. Erich Gamma, Richard Helm, Ralph Johnson, 'Design Patterns: Elements of Reusable Object-Oriented Software', Addison-Wesley Professional, 1994, ISBN-13: 978-0201633610.
5. Paul Clements, Felix Bachmann, Len Bass, David Garlan, 'Documenting Software Architectures: Views and Beyond', Addison-Wesley Professional, 2003, ISBN-13: 9780201703726.

MOOCs Links and additional reading material:

www.nptelvideos.in, www.coursera.com, www.udemy.com

Course Outcomes:

The student will be able to –

1. Compare Software Development Process Models and justify process maturity through application of Software Engineering principles and practices,
2. Differentiate competing and feasible system requirements identifying problem scope in the real-world,
3. Apply agile principles and practices through scrum framework,
4. Design UML diagrams through efficient system analysis, using identified design specifications
5. Formulate system architecture as per a suitable architectural style,
6. Apply relevant design patterns for effective system design.

CO-PO Map:

	Program Outcomes (PO)												PSO			
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4
1	2	3	2									3	1		2	
2	2	3	2									3	1		2	
3	3	2	3		3	2	2	2			2	3	1	2	3	2
4	3	2	3		3	2	2	2	2		2	3	1	2	3	2
5	3	2	3		3	2	2	2			2	3	1	2	3	2
6	3	2	3		3	2	2	2		3	2	3	1	2	3	2
Average	**2.66**	**2.33**	**2.66**		**3**	**2**	**2**	**2**	**2.0**	**3.0**	**2.0**	**3.0**	**1.0**	**2.0**	**2.66**	**2.0**

CO Attainment levels:

Weights for attainment levels: L1 - Easy- 0.75 L2 - Comfortable - 0.7 L3 – Medium – 0.65
L4 – Somewhat difficult – 0.6 L5 – Difficult – 0.55

CO1 – L1, CO2 – L2, CO3 – L3, CO4 – L4, CO5 – L4 and CO6 – L5

Future Course Mapping:

Software testing and Quality Assurance, Service-Oriented Software

Job Mapping:

Application Architect, Project Designer, SCRUM Role Player, Project Manager

Syllabus Draft **FF No. : 654**

Audit Courses

Semester I

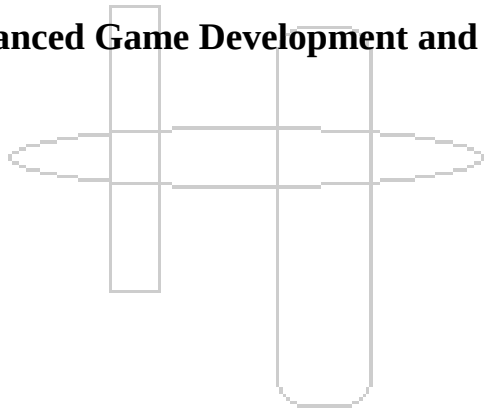
MD3154: Advanced Problem Solving and Programming (Audit course)

and

MD3155: Data Engineering (Audit course)

Semester II

MD3149:: Advanced Game Development and Experience Design



T. Y. B. Tech. Computer Engineering
AY 2025-26

Module VI Course Content

Syllabus Template **FF No. : 654**

CS3226: Cloud Computing

Credits: 4..... **Teaching Scheme Theory: 2 Hours/Week**

Tutorial: 1 Hours/Week

Lab: 2 Hours/Week

Course Prerequisites: Operating Systems, Computer Networks, Database Management System

Course Objectives:

1. To become familiar with cloud computing and its ecosystem
2. To acquire basics of virtualization and its importance
3. To evaluate in-depth analysis of Cloud Computing capabilities and its services.
4. To configure and implement storage services.
5. To analyze different cloud-based services to meet a set of given requirements.
6. To design security aspects for cloud computing

Course Relevance: Cloud computing to enable transformation, business development and agility in an organization.

SECTION-I Topics and Contents:

Unit-I Introduction to Cloud Computing **[4 Hrs]**

Recent trends in computing, Cluster computing, Distributed computing ,Evolution of cloud computing, Cloud versus traditional architecture, Cloud Computing Architecture, Google Cloud architecture, Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS), Public cloud, Private cloud, Hybrid cloud, Community cloud

Unit-II Virtualization **[6 Hrs]**

Introduction to virtualization, Different approaches to virtualization, Hypervisors, Machine Image, Virtual Machine (VM), Compute options in the cloud, Exploring IaaS with Compute Engine, Configuring elastic apps with auto scaling, Basics of virtualization and implementation challenges. System virtualization technologies-architectures and internals. KVM, Xen, VMware. Amazon Elastic Compute Cloud EC2 as computing service.

Memory virtualization-virtualization techniques, ballooning, deduplication and sharing. Network and storage virtualization, Virtual machine migration and replication techniques pre-copy and post-copy techniques, applicability to system availability.

Unit-III Cloud Services [4 Hrs]

Service Oriented Architecture (SOA), Web services, Web 2.0, Web OS. Introduction to IaaS, PaaS, SaaS. Cloud Platform and Management, Exploring PaaS with App Engine, Event driven programs with Cloud Functions, Containerizing and orchestrating apps with Google Kubernetes Engine Software as a Service (SaaS) Docker flow, orchestration with Docker, dynamic linking and legacy linking of containers. The GCP Console, understanding projects, Billing in GCP, Install and configure Cloud SDK, Use Cloud Shell, GCP APIs.

SECTION-II Topics and Contents:

Unit-IV Cloud Storage [4 Hrs]

Storage options in the cloud, Structured and unstructured storage in the cloud, unstructured storage using Cloud Storage, SQL managed services, Exploring Cloud SQL, Cloud Spanner as a managed service, NoSQL managed service options, Cloud Datastore, a NoSQL document store, Cloud Bigtable as a NoSQL option. OpenStack: NOVA, Neutron, Keystone Cinder, Swift and Glances, VMware Suit, Apache Cloud Stack, Data Lakes, Snowflake.

Unit-V Service Management [4 Hrs]

Service Level Agreements (SLAs), Billing and accounting, Billing in GCP Cloud Security: Introduction to security in the cloud, the shared security model, Encryption options, Authentication and authorization with Cloud IAM, Identify Best Practices for Authorization using Cloud IAM., Introduction to configuration and management tools Ansible, Architecture of DevOps.

Unit-VI Cloud Network and Security [6 Hrs]

Introduction to networking in the cloud, defining a Virtual Private Cloud, Public and private IP address basics, Google's network architecture, Routes and firewall rules in the cloud, Multiple VPC networks, building hybrid clouds using VPNs, interconnecting, and direct peering, Different options for load balancing. Introduction to security in the cloud, the shared security model, Encryption options,.

List of Tutorials (Any Thirteen)

List of Tutorials:

Unit-I Introduction to Cloud Computing

1) Install VirtualBox/VMware Workstation with different Linux or Windows Operating Systems.
2) Study Google Cloud Architecture.

Unit-II Virtualization

3) Find a procedure to launch virtual machine
4) Find a procedure to transfer the files from one virtual machine to another virtual machine.

Unit-III Cloud Services

5) Simulate a cloud scenario using CloudSim and run a scheduling algorithm that is not present in CloudSim.

6) Install Google App Engine. Create hello world app and other simple web applications using python/java.

Unit-IV Cloud Storage

7) Launch the Web Applications using GAE launcher.

8) Install Hadoop single node cluster and run simple applications like wordcount.

Unit-V Service Management

9) Use AWS Pricing Calculator: Create estimate for EC2 Compute cost for VM instance. Use region closest to you. Find On demand cost and compare the pricing for other regions.

Unit-VI Cloud Network and Security

9) Launch EC2instane and explore Public/Private/Elastic IP

Practical's:

List of Practical's (Any Six)

Unit-I Introduction to Cloud Computing

1) To setup AWS accounts and launch instances.

Unit-II Virtualization

2) To install an OS using VirtualBox/ VMWare Workstation. Add Storage to create new virtual disk.

3) To Deploy Virtual Machine on hypervisor such as KVM, ESXi. Take Backup and Migrate them.

Unit-III Cloud Services

4) To use Infrastructure as a Service to facilitates for creating and deleting compute resources. Create network and attach volumes to run instances.

5) To install docker on window/linux and build docker image from docker hub.

6) Deploy a stateless/stateful application on Kubernetes cluster.

Unit-IV Cloud Storage

7) To work on different Cloud Storage Services.

Unit-V Service Management

8)To create login into AWS and use S3 Bucket Service for storage.

Unit-VI Cloud Network and Security

9) Develop elastic services for dynamic load scenario using AWS APIs. Build load balancer and explore on scalability, fault detection and performance.

Course Projects:

List of Course Project Topics

1. Creating Google Account to store files and programs.
2. Creating Account to Store Images.
3. Creating a Warehouse Application in SalesForce.com
4. Creating an Application in SalesForce.com using Apex programming Language.

5. To study and implement Web services in SOAP for JAVA Applications.

6. Implementation of Para-Virtualization using VMWare ‘s Workstation/ Oracle’s Virtual Box and Guest Operator System.
7. Installation and Configuration of Hadoop.
8. AWS Case Study: Amazon.com.
9. Case Study of Google App Engine.
10. Case Study of Face book.

Seminars:

List of Course Seminar Topics

1. Storage Cost Optimization on Cloud.
2. Cloud Security and Cryptography
3. Infrastructure As A Code (IAC)
4. Cloud Computing in Healthcare
5. Serverless
6. Deployment of Microservices in Kubernetes Engine
7. RPA Using AWS Cloud
8. Cloud Trends In Supporting Ubiquitous Computing
9. Mobile Cloud Computing
10. Modern Data Center Architecture

Group Discussion:

List of Group Discussion Topics

1. Data Storage Security in Cloud
2. Cloud Services for SMB’s.
3. Monitoring Services Provided by GCP and AWS.
4. Docker and Kubernetes.
5. SaaS vs FaaS (Function as a service).
6. Hybrid Cloud.
7. GCP Vs AWS Web Service Architecture.
8. Cloud based security issues and threats.
9. Authentication and identity.
10. Future of Cloud-Based Smart Devices.

List of Home Assignments:

List of Design Based Home Assignments

1. Serverless Web App to order taxi rides using AWS lambda.
2. Deploying App on Kubernetes.
3. Serverless web Application (GCP Cloud Functions).

4. Demonstration of EBS, Snapshot, Volumes.
5. Single Node Cluster Implementation (Hadoop).

List of Case Study Based Home Assignments

1. PayU Migration to AWS.
2. Cloud object storage.
3. Deployment and Configuration options in AWS.
4. Deployment and Configuration options in Microsoft Azure.
5. Deployment and Configuration options in GCP.

List of Blog Based Home Assignment

1. Comparing design of various cloud computing platforms.
2. AWS EKS and Google Cloud Functions.
3. App Engine.
4. Cloud Endpoints.
5. Cloud Pub/Sub.

List of Survey Based Home Assignments

1. Disaster Recovery in Cloud Computing.
2. Cloud Economics.
3. Data archiving solutions.
4. Salesforce.
5. Dropbox.

Assessment Scheme: Suggest an Assessment scheme that is best suited for the course. Ensure 360 degree assessment and check if it covers all aspects of Bloom's Taxonomy.

Laboratory Continuous Assessment: 100 Marks converted to 10 Marks
Course Project: End Semester Examination: 100 Marks converted to 20 Marks
Presentation: End Semester Examination: 100 Marks converted to 20 Marks
Theory: End Semester Examination (MCQ): 60 Marks converted to 30 Marks
Comprehensive Viva Voce: End Semester Examination: 100 Marks converted to 20 Marks

Text Books: ***(As per IEEE format)***

1. Judith Hurwitz, R.Bloor, M.Kanfman, F.Halper, "Cloud Computing for Dummies", Wiley,India.
2. Ronald Krutz and Russell Dean Vines, "Cloud Security", Wiley-India
3. Gautam Shroff. "Enterprise Cloud Computing", Cambridge

Reference Books: ***(As per IEEE format)***

1. *Barrie Sosinsky, "Cloud Computing Bible", Wiley India*
2. Antohy T Velte, et.al, "Cloud Computing : A Practical Approach", McGraw Hill.

3. Michael Miller, "Cloud Computing", Que Publishing.
4. Tim Malhar, S.Kumaraswammy, S.Latif, "Cloud Security & Privacy", SPD,O'REILLY
5. Scott Granneman, "Google Apps", Pearson

MOOCs Links and additional reading material:

https://nptel.ac.in/courses/106/105/106105167/
https://swayam.gov.in/nd1_noc20_cs55/preview
https://www.coursera.org/specializations/cloud-computing
https://azure.microsoft.com/en-in/overview/what-is-cloud-computing/
https://aws.amazon.com/what-is-cloud-computing/
https://www.ibm.com/in-en/cloud/learn/cloud-computing

Course Outcomes:

On the completion of course, student will able to

1. Describe the main concepts, key technologies, strengths, and limitations of cloud computing and the possible applications for state-of-the-art cloud computing
2. Explain the architecture and infrastructure of cloud computing, including SaaS, PaaS, IaaS, public cloud, private cloud, hybrid cloud, etc.
3. Identify problems, and explain, analyze, and evaluate various cloud computing solutions.
4. Choose the appropriate technologies, algorithms, and approaches for the related issues.
5. Display new ideas and innovations in cloud computing.
6. Collaboratively research and write a paper on the state of the art (and open problems) in cloud computing.

CO-PO Map:

CO	Program Outcomes (PO)												PSO			
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4
CS3226.1	2	1			1									2	2	
CS3226.2	2	2	1	1	1									2	2	
CS3226.3	3	2	2	2	2		3	3						2	3	
CS3226.4	3	2	2	2	3	3			3					2	3	
CS3226.5	3	3	1	3	3				1		2			2	3	
CS3226.6	2	2	1	3	1					3		3			2	2
Average	2.50	2.00	1.40	2.20	1.83	3.00	3.00	3.00	2.00	3.00	2.00	3.00		2.0	2.50	2.00

CO attainment levels:

Attainment Levels:1,2,3,5,4,3

Future Course Mapping:

After completing this course different certifications courses in cloud be taken such as AWS, Azure, Google cloud certifications. One can go for higher studies in specialization of cloud computing and allied subjects

Job Mapping:

Cloud Architect, Cloud Engineer, Cloud Administrator, Solutions Architect - Cloud Computing - AWS / Kubernetes, Cloud Computing Technical Consultant, Associate Cloud Computing Engineer, Cloud Computing Trainer

Syllabus Template **FF No. : 654**

CS3202: Artificial Intelligence

Credits: 4..... **Teaching Scheme Theory: 2 Hours/Week**

Tutorial: 1 Hours/Week

Lab: 2 Hours/Week

Course Prerequisites:

- A course on "Computer Programming and Data Structures"
- A course on "Mathematical Foundations of Computer Science"
- Some background in linear algebra, data structures and algorithms, and probability will be helpful

Course Objectives:

1. To learn the distinction between optimal reasoning Vs. human like reasoning
2. To understand the concepts of state space representation, exhaustive search, heuristic search together with the time and space complexities.
3. To learn different knowledge representation techniques.
4. To understand the applications of AI, namely game playing, theorem proving, and machine learning.

Course Relevance: Technologies driven by artificial intelligence (AI) have transformed industries and everyday life. The possibilities for AI applications are virtually unlimited and sought after in practically every industry segment. That's why global organizations are actively recruiting professionals with specialized skills and proficiencies needed to develop future AI technological innovations.

SECTION-I -Topics and Contents:

Unit-I Title: Fundamentals of Artificial Intelligence

Introduction: A.I. Representation, Non-AI & AI Techniques, Representation of Knowledge, Knowledge Base Systems, State Space Search, Production Systems, Problem Characteristics, Types of production systems, Turing Test. **Intelligent Agents**: Agents and Environments, concept of rationality, the nature of environments, structure of agents, problem solving agents, problem formulation. **Formulation of problems**: Vacuum world, 8 queens, Route finding, robot navigation.

Unit-II Title: UninformedSearch Strategies

Uninformed Search Methods: Depth First Search, Breadth First Search, Depth Limited Search,Iterative Deepening Depth First Search, Bidirectional Search, Comparison of Uninformed search Strategies.

Unit-III Title: Informed Search Methods:

Generate & test, Hill Climbing, Best First Search, A* and AO* Algorithm, Constraint satisfaction, Means Ends Analysis, **Game playing:** Minimax Search, Alpha-Beta Cut offs, Waiting for Quiescence

SECTION-II Topics and Contents:

Unit-IV Title: Logical Agents:

Knowledge based agents, Wumpus world. **Propositional Logic**: Representation, Inference, Reasoning Patterns, Resolution, Forward and Backward Chaining. **First order Logic**: Representation, Inference, Reasoning Patterns, Resolution, Forward and Backward Chaining.

Unit-V Title: Basics of PROLOG:

Representation, Structure, Backtracking. **Expert System**: Design, Implementation, Case study of Expert System in PROLOG

Unit-VI Title: Planning:

Blocks world, STRIPS, Implementation using goal stack, **Planning with state space search:** Forward state space search, Backward state space search, Heuristics for state space search. Partial Order Planning, Planning Graphs, Hierarchical planning, Least commitment strategy. Conditional Planning, Continuous Planning

List of Tutorials

1. AI problem formulation
2. Task Environment
3. AI Problem Characteristics
4. Missionaries and Cannibals Problem
5. Water Jug Problem
6. Monkey Banana problem
7. 8 Puzzle Problem
8. Magic Square problem
9. Tic-Tac Toe Problem
10. Robot Navigation

11. Propositional Logic Examples
12. Predicate Logic Examples
13. Mini Expert system examples

List of Practical's

1. Implementation of AI and Non-AI technique by implementing any two player game
2. Implementation of Uninformed strategies
3. Implementation of Informed strategies
4. Implementation of CSP Problem
5. Implementation predicate logic using PROLOG
6. Implementation of Expert system using PROLOG

List of Course Project Topics (Sample topics)

1. Inventory management E Commerce
2. stock market price prediction
3. Object Identification / detection
4. Product Delivery Drones
5. Pick and drop robotic arm
6. Arrangement of blocks
7. Smart city water / light management system
8. Human Tracking system
9. Automatic Interview Conduction system
10. Student Information Chatbot Project
11. Product Review Analysis For Genuine Rating
12. Customer Targeted E-Commerce
13. College Enquiry Chat Bot
14. Artificial Intelligence HealthCare Chatbot System
15. Intelligent Tourist System Project

List of Course Seminar Topics

1. Fundamentals of Artificial Intelligence
2. Intelligent Agents
3. Uninformed searching Techniques
4. Informed searching Techniques
5. Gaming Techniques
6. Planning Techniques
7. Applications of AI
8. Predicate Logic
9. Propositional Logic

10. Adversarial Search Techniques

List of Home Assignments:

List of Design Based Home Assignments

1. Design of intelligent algorithm for AI Accessibility
2. Design of AI algorithm for Robot Navigation.
3. Design of AI algorithm for Customer Experience
4. Design of AI algorithm for Data-Informed Design
5. Design of AI algorithm for AI Decision Making
6. Design of AI algorithm for any application for Children
7. Design of AI algorithm for problems of Senior Citizens
8. Design of AI algorithm for ecommerce Applications
9. Design of AI algorithm for Enterprise UX Design
10. Design of AI algorithm as Teaching Aid for teachers

List of Case Study Based Home Assignments

1. How Automobile Sector Is Preparing For The 4th Industrial Revolution using AI
2. How Indian Retail Giant Is Using AI And Robots To Prepare For The 4th Industrial Revolution
3. Rolls-Royce And Google Partner To Create Smarter, Autonomous Ships Based On AI
4. The Amazing Ways Tesla Is Using Artificial Intelligence And Big Data
5. The Incredible Ways John Deere Is Using Artificial Intelligence To Transform Farming
6. Challenges/Issues in AI applications
7. Research problems in AI
8. AI in Search Engine
9. Future of AI
10. AI in Agriculture

List of Blog Based Home Assignment

1. AI Trends
2. AI Research
3. AI Chatbot
4. Chatbot Magazine
5. AI Medical / Agriculture
6. AI Challenges
7. Knowledge based Inference Engine
8. Rule based inference Engine

9. Truth maintenance system
10. AI in CSP problems

List of Survey Based Home Assignments

1. Adaption of AI in 2020
2. AI in Industry
3. AI in Digital Marketing
4. AI in Gaming
5. AI after Covid-19
6. AI in rule based systems
7. Analysis of Search Engines : AI perspective
8. Page rank algorithms in AI
9. AI in Ecommerce
10. Analysis of Expert systems in medical diagnosis

Assessment Scheme: Suggest an Assessment scheme that is best suited for the course. Ensure 360 degree assessment and check if it covers all aspects of Bloom's Taxonomy.

Laboratory Continuous Assessment: 100 Marks converted to 10 Marks
Course Project: End Semester Examination: 100 Marks converted to 20 Marks
Laborarotory Practical: End Semester Examination: 100 Marks converted to 50Marks
Comprehensive Viva Voce: End Semester Examination: 100 Marks converted to 20 Marks

Text Books: ***(As per IEEE format)***

1. *Elaine Rich and Kevin Knight: "Artificial Intelligence." Tata McGraw Hill*
2. Stuart Russell & Peter Norvig : "Artificial Intelligence : A Modern Approach", Pearson Education, 2nd Edition.
3. Deepak Khemani: "A First Course in Artificial Intelligence", Mc Graw Hill
4. Saroj Kaushik: "Artificial Intelligence" Cengage Publication

Reference Books: ***(As per IEEE format)***

1. Ivan Bratko : "Prolog Programming For Artificial Intelligence" , 2nd Edition Addison Wesley,1990.
2. Eugene, Charniak, Drew Mcdermott: "Introduction to Artificial Intelligence.", Addison Wesley
3. Patterson: "Introduction to AI and Expert Systems", PHI
4. Nilsson: "Principles of Artificial Intelligence", Morgan Kaufmann.
5. Carl Townsend, "Introduction to turbo Prolog", Paperback, 1987

MOOCs Links and additional reading material:

www.nptelvideos.in

Course Outcomes:

On the completion of course, student will able to

1. Understand the basics of the theory and practice of Artificial Intelligence as a discipline and about intelligent agents capable of problem formulation.
2. Identify problems that are amenable to solution by AI methods, and which AI methods may be suited to solving a given problem.
3. Evaluation of different uninformed and informed search algorithms on well formulated problems along with stating valid conclusions that the evaluation supports.
4. Formulate and solve a given problem using Propositional and First order logic.
5. Analyze the AI problem using different planning techniques.
6. Design and carry out an empirical evaluation of different algorithms on problem formalization, and state the conclusions that the evaluation supports.

CO-PO Map:

CO	Program Outcomes (PO)												PSO			
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4
CS3226.1	2											2			2	
CS3226.2	2	2		1								2			2	
CS3226.3	2	3		2								2	2		3	
CS3226.4	2	3	3	2								2	2	2	3	
CS3226.5	2	3		2								2	2	2	3	
CS3226.6	2	3	3	2			3	2	2	2		3	3		3	2
Average	2.0	2.8	3.0	1.8			3.00	2.00	2.00	2.00		2.16	2.33	2.0	2.50	2.00

CO attainment levels:

CO1 -2, CO2-2 CO3-3 CO4-1 CO5-3 CO6-1

Future Course Mapping:
Mention other courses that can be taken after completion of this course
Machine Learning

Job Mapping:
What are the Job opportunities that one can get after learning this course
AI Data Analyst, Data Scientist

Syllabus Template **FF No. : 654**

CS3334:: Cryptography and Information Security

Credits: 4..... **Teaching Scheme Theory: 2 Hours/Week**

Tutorial: 1 Hours/Week

Lab: 2 Hours/Week

Course Prerequisites: Computer Networks

Course Objectives:

1. Apply cryptographic techniques and security protocols to secure systems and networks.
2. Identify, resolve, and mitigate programming bugs and cyber threats.
3. Design secure systems using blockchain technology and ensure application security.
4. Understand and apply cloud security and physical security principles.
5. Integrate AI in cyber security and develop business continuity and disaster recovery plans.
6. Implement ethical hacking practices and perform effective penetration testing.

Course Relevance:
Cyber Security teaches how to protect operating systems, networks, and data from cyber attacks, monitor systems, and mitigate threats, aiming to develop skills to prevent attacks and protect data privacy.

Section 1: Topics/Contents

Information security

- **Key Security Properties:** Confidentiality, Integrity, Availability.
- **Risk Management:** Understanding governance policies, frameworks, laws, regulations, guidelines, and compliance.
- **Symmetric Key Cryptography:** Role of random numbers and nonce in security, importance of prime numbers, GCD, Euclid's Algorithm, Extended Euclid's algorithm.
- **Data Encryption Standard (DES):** Block cipher, stream cipher, Feistel structure, block cipher modes, S-DES, attacks on DES, S-AES, AES.

- **Public Key Cryptography:** RSA algorithm, key generation, attacks on RSA.
- **Elliptic Curve Cryptography (ECC):** Elliptic curves over real numbers and Zp, elliptic curve arithmetic.

Network Security

- **Certificates and Hashing:** Properties of hash functions, HASH + SALT, hashing algorithms (SHA1, SHA2).

- **Authentication and Authorization:** Network access control (SHA-512, Kerberos, and multifactor authentication).
- **Transport-Level Security:** Web security considerations, Secure Sockets Layer (SSL), Transport Layer Security (TLS), HTTPS standard, Secure Shell (SSH) application, IPSec.
- **Application Security:** Security by design, writing secure code, static and dynamic application security testing (SAST and DAST), interactive application security testing (IAST), Integrated Security in DevOps, OWASP, Application Security Services,

Section 2: Topics/Contents

Cyber Attacks and Penetration Testing (06 Hours)

- **Cyber Ethics:** Threats, threat modeling, injections, sniffing, and types of attacks.
- **Security Vulnerabilities:** risk, attack types, countermeasures.
- **Protocol Vulnerabilities:** DoS and DDoS, session hijacking, ARP spoofing.
- **Software Vulnerabilities:** Phishing, buffer overflow, cross-site scripting attack, ransomware, SYN-flooding, SQL-injection, DNS poisoning.
- **Penetration Testing:** Difference from automated vulnerability scans, objectives and limitations of a pen test, scoping and planning pen tests, executing pen tests and managing findings. Introduction to SDL (Secure Development Lifecycle) – Merging Security into SDLC,

Physical Security and Forensics (04 Hours)

- **Physical Security:** Physical access types, crime prevention through environmental design (CPTED).
- **IoT Security:** Definitions of OT, IoT, IIoT, and ICS, most widely used protocols in IoT environments (MQTT and CoAP).
- **Business Continuity (BC):** RTP/RPO, RTO, MTPD, ISO 22301 standard for business continuity management, importance, differences between BCMS and DRMS, risk management, testing, maintenance., Operation Resilience,
- **Digital Forensics:** Introduction to digital forensics, data recovery, OS forensics, email crimes and violations, cyber forensics.

Cloud Security

Principles / Key Concepts of Cloud Security: Overview of cloud security principles and key concepts.

- **Threats and Risks in Cloud Security:** Diverse types of threats and risks associated with cloud security.
- **Importance of Security Measures in Cloud Security:** Importance of implementing security measures in cloud environments.
- **Solutions for Cloud Security:** Effective solutions to address cloud security challenges.

Role of AI in Cyber Security: Examination of how AI is integrated into cyber security.

- **Challenges and Opportunities of AI in Cyber Security:** Analysis of the challenges and opportunities presented by AI in the field of cyber security.

Introduction to Blockchain

List of Tutorials (13)

1.Mathematical background for cryptography: modulo arithmetic, GCD (Euclid's algorithm), algebraic structures (Groups, Rings, Fields, Polynomial Field).

2.Chinese remainder theorem.

3. Diffie-Hellman key exchange: Algorithm, Key exchange protocol, Attack.

4. ECC over Diffie-Hellman key exchange.

5. Study of certificates and hashing algorithms.

6. Network access control and transport-level security.

7. Security by design and writing secure code.

8. Static and dynamic application security testing.

9. Study of Snort.

10. Nessus: a Security Vulnerability scanning tool.

11. Metasploit/Ollydbg.

12. Testing for Brute Force Password.

13. Testing for SQL Injection.

14. Computer forensics, Facebook forensic, mobile forensic, cyber forensic, digital forensic.

15. Source Code Analysis Tools.

16. OWASP Zed Attack Proxy (ZAP).

17. Study of various types of Blockchain, Connecting the Metamask wallet with the local Ganache network.

18. Simulation of Blockchain.

19. Creating Smart Contract using Solidity and Remix IDE.

20. Study of DOA and DAPP.

List of Practicals (Minimum Six)

Section-I:

- Simplified DES implementation.
- Simplified AES implementation.

- Encryption and Decryption by RSA algorithm.
- Implementation of ECC over Diffie Hellman Key Exchange Protocol.

- Implementation of authentication algorithms.
- Implementation of SHA.

Section-II:

- Acquisition of System Information/ RAM/Volume Shadow Copy/Detecting Encryption in information.
- Vulnerabilities finding in Mobile/ computer/ digital devices.

- Forensic of Disc Image/ Registry/ Meta data/ RAM.
- Digital forensic of images.
- Forensics of Video alteration.

- Implement and demonstrate the use of the following in Solidity: Variable, Operators, Loops, Decision Making, Strings, Arrays, Enums, Structs.
- Implement and demonstrate the use of the following in Solidity: Functions, Function Modifiers, View functions, Pure Functions, Mathematical functions, Cryptographic functions.
- Use Geth to configure a private Blockchain node in our machine.
- Cryptography in Blockchain, Merkle root tree hash.
- Creating Transactions using Solidity and Remix IDE.
- Case Study on Hyperledger Fabric.

List of Course Project areas:

Course Project 01 Statement: Design a System to develop a analyzer which will differentiate between different vulnerability and packets entered using it. This system will detect the intrusions coming through the vulnerabilities.

Course Project 02 Statement: Securing Video Conferencing App for online meetings

Course Project 03 Statement: Steganography for Image/Video/Files

Course Project 04 Statement: Secure Image display on online social media.

Course Project 05 Statement: Secure transfer of government subsidies to farmers/BPL people/ students etc

Course Project 06 Statement: Authentication of users for various applications for integrity, availability, confidentiality.

Course Project 07 Statement: Implementing a system for detecting the modification of videos/images on social media

Course Project 08 Statement: Secure App for online exams detecting Keystroke and camera movements.

Course Project 09 Statement: A system to detect the difference between the voice edited in the audio/video

Course Project 10 Statement: A System to check the vulnerabilities in the websites.

Course Project 11 Statement: Decentralized (Uber)Peer to Peer Carpooling

Course Project 12 Statement: Decentralized Skill Verification System

Course Project 13 Statement: Decentralized talent acquisition (like Nokari.com)

Course Project 14 Statement: Decentralized gaming DAPP(earn coin through game)

List of Course Seminar Topics

Seminar 01 Statement: Blockchain architecture and its implementation

Seminar 02 Statement: Cloud Security

Seminar 03 Statement: Mobile Security

Seminar 04 Statement: IoT and Security Issues/ Security Models for IoT

Seminar 05 Statement: Dark web

Seminar 06 Statement: Docker Security

Seminar 07 Statement: Access control methods for online social media and various organizations

Seminar 08 Statement: Security of Android Vs IOS

Seminar 09 Statement: Machine learning and SCADA Security

Seminar 10 Statement: Security Applications for Smart

List of Design Based Home Assignments

HA_D 01 Statement: Design a secure system using cryptography techniques for security of multimedia files.

HA_D 02 Statement: Design a secure system using steganography for hiding data files in image/video

HA_D 03 Statement: Design a system for educational institutes using authentication and authorization techniques, also give details about the access control policies that must be implemented for the design of system by various places.

HA_D 04 Statement: Design a secure system using SSL/TLS/IPSec for the various organizations

HA_D 05 Statement: Design a system for the analysis of cyber crime using various cyber forensic techniques and compare each technique with respect to integrity, confidentiality, availability

List of Case Study Based Home Assignments

HA_CS 01 Statement: How to improve the security of social media? Write a detail case study

HA_CS 02 Statement: Find out the vulnerability issues in educational institutes websites/online systems and give solutions to these problem. Perform a detailed case study of the various issues.

HA_CS 03 Statement: Write a detail case study about the banking security flows and solutions to these flows.

HA_CS 04 Statement: Give a detail case study of the antivirus system giving the flows and solutions to it.

HA_CS 05 Statement: Perform the detail case study of various operating systems used for mobile devices and give a secure solution to one for widely used OS.

List of Blog Based Home Assignment

HA_Blog 01 Statement: Dark Web

HA_Blog 02 Statement: Crypto currency and Economy

HA_Blog 03Statement:.Cybercrime and solutions

HA_Blog 04 Statement: Authentication and Access control for social media

HA_Blog 05 Statement: Cyber forensic and Cyber laws

List of Survey Based Home Assignments

HA_Survey 01 Statement: Survey on various blockchain related issues/ cryptocurrency/ application systems developed using blockchain

HA_Survey 02 Statement: Survey on various authentication and access control methods for different applications

HA_Survey 03 Statement: Steganography and Biometric Systems for authentication

HA_Survey 04 Statement: Survey of various attacks and its effect on Indian economy and its analysis

HA_Survey 05 Statement:Problems over Integer Lattices: A Study

Assessment Scheme: Suggest an Assessment scheme that is best suited for the course. Ensure 360 degree assessment and check if it covers all aspects of Bloom's Taxonomy.

Laboratory Continuous Assessment: 100 Marks converted to 10 Marks
Course Project: End Semester Examination: 100 Marks converted to 20 Marks
Presentation: End Semester Examination: 100 Marks converted to 20 Marks
Theory: End Semester Examination (Written): 60 Marks converted to 30 Marks
Comprehensive Viva Voce: End Semester Examination: 100 Marks converted to 20 Marks

Text Books: ***(As per IEEE format)***

1. *William Stallings, "Cryptography and Network Security-Principles and Practices" 6th Edition, Pearson Education, 2014, ISBN13:9780133354690.*
2. *Bernard Menezes, "Network Security and Cryptography", 1st Edition, Cengage Learning, 2010, ISBN 81-315-1349-1.*

3. *Raef Meeuwisse, "Cybersecurity for Beginners", 2nd Edition, Cyber Simplicity, 2017, ISBN-9781911452157*
4. *AmbadasTulajadasChoudhari, Arshad SarfarzAriff, Sham M R, "Blockchain for Enterprise Application Developers" Willey publications, ISBN: 9788126599967,2020*
Hyperledger Fabric - https://www.hyperledger.org/projects/fabric

Reference Books: ***(As per IEEE format)***

1. *M. Speciner, R. Perlman, C. Kaufman, "Network Security: Private Communications in a Public World", Prentice Hall, 2002*
2. *Michael Gregg, "The Network Security Test Lab: A Step-By-Step Guide", Dreamtech Press, 2015, ISBN-10:8126558148, ISBN-13: 978-8126558148.*
3. *Matt Bishop, "Computer Security: Art and Science", 1st Edition, Pearson Education, 2002, ISBN 0201440997.*
4. *Charlie Kaufman, Radia Perlman and Mike Spencer, "Network security, private communication in a public world", 2nd Edition, Prentice Hall, 2002, ISBN 9780130460196.*
5. *V.K. Pachghare, "Cryptography and Information Security", 2nd Edition, PHI, 2015, ISBN-978-81-203-5082-3.*
6. *Mastering Blockchain: Deeper insights into decentralization, cryptography, Bitcoin, and popular Blockchain frameworks by Bashir, Imran,2017.*

Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, and Steven Goldfeder. Bitcoin and cryptocurrency technologies: a comprehensive introduction. Princeton University Press, 2016

MOOCs Links and additional reading material:

1. Cryptography And Network Security By Prof. Sourav Mukhopadhyay, IIT Kharagpur Cryptography and Network Security - Course (nptel.ac.in)
2. Information Security and Cyber ForensicsBy Prof. Pratosh Bansal Devi Ahilya Vishwavidyalaya, Indore, Information Security and Cyber Forensics - Course (swayam2.ac.in)
3. Blockchain and its Applications By Prof. Sandip Chakraborty, Prof. Shamik Sural IIT Kharagpur Blockchain and its Applications - Course (nptel.ac.in)

Course Outcomes:

The student will be able to –

1. Demonstrate cryptographic techniques using a mathematical approach by examining nature of attack.
2. Design a secure system for protection from the various attacks for 7 layer model by determining the need of security from various departments of an organization
3. Justify various methods of authentication and access control for application of technologies to various sections of industry and society.
4. Identify and establish different attacks on the system.

5. Estimate future needs of security for a system by researching current environment on a continuous basis for the benefit of society.
6. Analyze the need of Decentralized system and implement using blockchain technology.

CO-PO Map:

CO	Program Outcomes (PO)												PSO			
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4
1	3	**3**		3	**2**	2		3	1				3		1	3
2	3	2	3	2			2	3	3	3	2	1				2
3	2	3	**3**		1	2		3	1						1	3
4	3	3	1	**3**	3	3	3	**3**	1			3		3	1	3
5	2	2	3	2	1		2		3	3	3	3				
6	3	2	1	1	3	3	3	3			**3**	**2**		3	3	3
Avg	**2.67**	**2.5**	**2.2**	**2.2**	**2**	**2.5**	**2.5**	**3**	**1.8**	**3**	**2.67**	**2.25**	**3**	**3**	**1.5**	**2.8**

CO Attainment levels:

Attainment Levels: 3,4, 2, 1, 5, 3

Future Course Mapping:

Cloud Computing and Security, IoT Security, Ethical Hacking &Cyber Forensics

Job Mapping:

Security Engineer/Network Security Engineer

Information Security Analyst

Cyber Security Analyst

Cyber Security Associate

Manager-Information Security Services

Security Consultant

Penetration Testing Engineer

Syllabus Template **FF No. : 654**

CS3053::Compiler Design

Credits: 4 **Teaching Scheme Theory: 2 Hours/Week**

Tutorial: 1 Hours/Week

Lab: 2 Hours/Week

Course Prerequisites: Operating System, Theory of Computer Science

Course Objectives:

1. Understand the process of program execution cycle.
2. Understand the translation process from High Level Languages to Machine Level Language.
3. Know the syntax and semantic analysis approaches for efficient code/program verification.
4. Learn the methods of code generation which helps for the optimization.
5. Learn code optimization and runtime code synthesis.
6. Know the process of compiler design for emerging programming languages.

Course Relevance:

All high level programming languages are easy for users to understand but not understandable to a computing machine. The computing machine knows only binary data. A translation is required to convert higher level language into machine level, so that the intended program can be executed. This translation is done by using a compiler. This course will give you detailed insights of how compilers function internally. This gives freedom to design your own programming language with its compiler. The key technology of Compiler Design principles is to provide an in-depth view of translation and optimization process. Compiler design covers basic translation mechanism and error detection and recovery.

Section 1: Topics and Contents

Unit-I: Introduction to Compilers, interpreters, Assembler, Linker and Loader [04 Hours]

Compilers: Introduction to compiler phases, features of machine-dependent and independent compilers, overview of types of compilers, introduction to cross compiler, Interpreters: compiler vs. interpreter, phases, and working, Preprocessor: header file and macro expansion, Assembler: Introduction to Assembler, overview of types of Assembler, Linker and Loader: Introduction to Linker and Loader, overview of types of Linker and Loader.

Unit-II: Lexical Analysis and Syntax Analysis: **[04 Hours]**

Introduction to Compiler, Phases and Passes, Bootstrapping, Role of a Lexical Analyzer, Specification and Recognition of Tokens, LEX/FLEX, Expressing Syntax, Top-Down Parsing, Predictive Parsers, Implementing Scanners, Operator Precedence Parsers.

Unit-III: Syntax Analysis and Semantic Analysis: [05 Hours]

Bottom-Up Parsing, LR Parsers: Overview of types of LR Parsers, Constructing LALR parsing tables, Introduction to YACC/BISON, Type Checking, and Type Conversion, Symbol Table Structure.

Section 2: Topics and Contents

Unit-IV: Syntax-Directed Translation and Intermediate Code Generation: [05 Hours]

Syntax-DirectedDefinitions, Bottom-Up Evaluation, Intermediate Representations, and Intermediate Code Generation: various formats of intermediate codes, Error Detection & Recovery: Lexical Phase errors, syntactic phaseerrors, semantic errors. More about translation: Array references in arithmetic expressions.

Unit-V: Code Generation: [04 Hours]

Issues in Code Generation, Basic Blocks and Flow Graphs, Next-use information, Simple Code Generator, DAG representation of Basic Blocks, Peephole Optimization. Generating code from DAGs

Unit-VI: Code Optimization, Run-Time Environments and Data Flow Analysis: [06 Hours]

Code Optimization and Run-Time Environments: Principle Sources of Optimization, Optimization of basic blocks, Global Data Flow Analysis, RuntimeEnvironments, and Source Language issues. Storage Organization, Storage Allocation strategies, Access to non-local names, Parameter Passing, Machine Dependent Optimization, Data Flow Analysis**:** Constant propagation, live range analysis.

Case studies: LLVM compiler Infrastructure, compiling OOP features, Compiling in multicore environment, Deep learning compilation, Parallel Compilers, Web Compilers.

List of Tutorials (13):

1. Single and two pass Assembler
2. Two pass Macro processor
3. Types of Linkers
4. Types of Loaders

5. Examples on First and Follow
6. Examples on Lex/Flex regular expressions
7. Construction of LL(1) parser
8. Construction of SLR parsing table
9. Construction of Canonical LR parsing table
10. Examples on YACC/Bison grammar rules
11. Translation Scheme
12. Examples of Intermediate code generation by Quadruples
13. Examples of DAG representation

List of Practical's (Minimum Six to be performed out of 10):

1. LEX/FLEX specification and programming regular expressions.
2. Implement LEX/FLEX code to count the number of characters, words and lines in an input file.
3. Implement LEX/FLEX code to select only lines that begin or end with the letter 'a' and delete everything else.
4. Convert all uppercase characters to lowercase except inside comments.
5. Change all numbers from decimal to hexadecimal notation, printing a summary statistic (number of replacements) to stderr.
6. Implement Lexical Analyzer for language C.
7. Implement LR/SLR/LALR Parser.
8. YAAC specifications and implement Parser for specified grammar.
9. Implement Parser for language C.
10. Implement Syntax directed Translator.
11. Implement an Intermediate code generator (three address code and Quadruples)
12. Implement a code optimizer for C/C++ subset.
13. Implement a code generator for C/C++ subset.

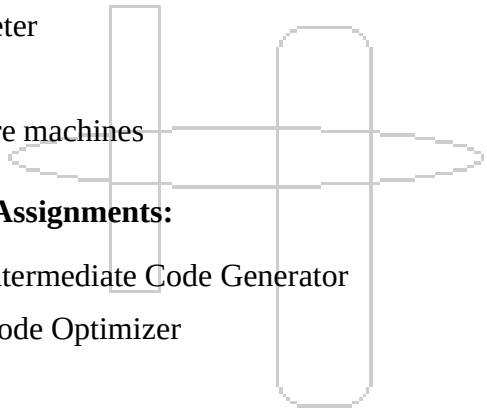
List of Course Project areas:

1. Compiler for subset of C using Lex and YAAC.
2. Compiler for Subset of Java programming Language.
3. Intermediate Code generator.
4. Code Optimizer.
5. Develop an Editor for Assembly programming. (Use available Assembler MASM/TASM to compile the code and execute in editor).

6. Design a system to check syntax and semantics of English Language.
7. Design a system to check syntax and semantics of a subset of Logical programming Language.
8. Design a System to check syntax and semantics of a subset of Python programming language.
9. Compiler for subset of C++ programming language.
10. Compiler for a subset of Algol programming language.

List of Seminar Topics:

1. Tools complementary to Lex
2. Tools complementary to YAAC
3. Semantic Analyser
4. Obsolete programming Language compiler advantage and issues
5. Android App program compiler
6. Approaches of Intermediate Code generation
7. Recent Trends in Compiler
8. Recent Trends in Interpreter
9. Decompilation
10. Compilation in multicore machines



List of Design based Home Assignments:

1. Recent methodologies in Intermediate Code Generator
2. Recent methodologies in Code Optimizer
3. Universal Compiler
4. Compiler for Deep learning
5. Recent trend in parsers

List of Case Study based Home Assignments:

1. Algol language Compiler
2. Compilation process (internals) of Functional Programming
3. Compilers for Mobile App development
4. LLVM compiler
5. Cross compiler

List of Blog based Home Assignments:

1. Decompilers: Ethical or Unethical?
2. Multi-paradigm programming compiler
3. State of the Art tools for rapid compiler development
4. Compiler for parallel machines
5. Compiler for distributed computing

List of Survey based Home Assignments:

1. Obsolete Programming Language Compilers
2. Obsolete Programming Language Interpreter
3. Compilers for various programming paradigms
4. Online compilers
5. Mobile app cross compiler

Assessment Scheme: Suggest an Assessment scheme that is best suited for the course. Ensure 360 degree assessment and check if it covers all aspects of Bloom's Taxonomy.

Laboratory Continuous Assessment: 100 Marks converted to 10 Marks
Course Project: End Semester Examination: 100 Marks converted to 20 Marks
Home Assignment: End Semester Examination: 100 Marks converted to 20 Marks
Theory: End Semester Examination (MCQ): 60 Marks converted to 30 Marks
Comprehensive Viva Voce: End Semester Examination: 100 Marks converted to 20 Marks

Text Books: ***(As per IEEE format)***

1. *Aho, A.V., Lam, M.S., Sethi, R., & Ullman, J.D. (2006). Compilers: Principles, Techniques,and Tools, Addison Wesley, ISBN 978-81317-2101-8 (2nd Edition).*

2. *Cooper, K., & Torczon, L. (2011). Engineering a compiler. Morgan Kaufmann, ISBN 155860-698-X.*

3. *Appel, A. W. (2004). Modern compiler implementation in C. Cambridge university press.*

4. *Appel, A. W., & Jens, P. (2002). Modern compiler implementation in Java. In ISBN 0–521–58388–8. Cambridge University Press.*

5. *Appel, A. W. (1998). Modern Compiler Implementation in ML, In ISBN 0-521-60764-.Cambridge University Press.*

6. Raghavan, V. (2010). Principles of Compiler Design. Tata McGraw-Hill Education

Reference Books: ***(As per IEEE format)***

1. *Muchnick, S. (1997). Advanced compiler design implementation. Morgan Kaufmann, ISBN8178672413.*

2. *Levine, J. R., Mason, J., Levine,* J. R., Mason, T., Brown, D., Levine, J. R., & Levine, P.(1992). Lex *& yacc.*

"O'Reilly Media, Inc".

MOOCs Links and additional reading material:

1. www.nptelvideos.in
2. https://swayam.gov.in/nd1_noc20_cs13/preview
3. https://www.udacity.com/course/compilers-theory-and-practice--ud168
4. https://online.stanford.edu/courses/soe-ycscs1-compilers

Course Outcomes:

The student will be able to –

1. Design basic components of a compiler including scanner, parser, and code generator.
2. Perform semantic analysis in a syntax-directed fashion using attributed definitions.
3. Apply local and global code optimization techniques.
4. Synthesize machine code for the runtime environment.
5. Develop software solutions for the problems related to compiler construction.
6. Adapt themselves to the emerging trends in language processing.

CO-PO Map:

	Program Outcomes (PO)												PSO			
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4
CO1	3	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0
CO2	3	2	1	0	1	0	0	0	0	0	0	0	3	3	0	0
CO3	2	3	3	2	0	2	0	0	0	0	0	0	0	3	0	2
CO4	3	3	3	2	2	0	0	0	0	0	0	0	3	0	0	0
CO5	3	3	1	2	3	2	3	2	1	0	0	3	3	3	3	2
CO6	2	3	0	1	2	0	3	0	0	0	0	1	0	3	3	0
Average	**2.67**	**2.83**	**2**	**1.75**	**2**	**2**	**3**	**2**	**1**	**0**	**0**	**2**	**3**	**3**	**3**	**2**

CO Attainment levels:

Weights for attainment levels: L1 - Easy-0.75 L2 - Comfortatble-0.7 L3 – Medium – 0.65

L4 – Somewhat difficult – 0.6 L5 – Difficult – 0.55

CO1 – L2, CO2 – L3, CO3 – L3, CO4 – L4, CO5 – L5 and CO6 – L5

Future Course Mapping: Parallel Compiler

Job Mapping: Compiler Engineer

B. Tech. Final Year Computer Engineering AY 2025-26

Module VII and VIII
Course Content

Syllabus Template **FF No. : 654**

OE1: LinkedIn Learning

LinkedIn Learning Courses **Credits: 2**

Sr. No.	Course code	Specialization Name	Link
1	MD4274	Large Language Models Skill Development	https://www.linkedin.com/learning/paths/develop-your-skills-with-large-language-models?u=126888530
2	MD4275	Mastering Microsoft Power BI	https://www.linkedin.com/learning/paths/master-microsoft-power-bi-15399694?u=126888530
3	MD4276	Generative AI Skills for Developers	https://www.linkedin.com/learning/paths/building-generative-ai-skills-for-developers?u=126888530
4	MD4277	Career in Data Analysis	https://www.linkedin.com/learning/paths/explore-a-career-in-data-analysis?u=126888530
5	MD4278	Concepts of Data Visualization and Storytelling	https://www.linkedin.com/learning/paths/master-the-concepts-of-data-visualization-and-storytelling?u=126888530
6	MD4279	AWS Certified Solutions Architect	https://www.linkedin.com/learning/paths/prepare-for-the-aws-certified-solutions-architect-associate-saa-c03-certification?u=126888530
7	MD4280	IT Security Specialist	https://www.linkedin.com/learning/paths/become-an-it-security-specialist?u=126888530
8	MD4281	Technical Program Management	https://www.linkedin.com/learning/paths/technical-program-management?u=126888530
9	MD4282	Natural Language Processing Skill Development	https://www.linkedin.com/learning/paths/advance-your-skills-in-natural-language-processing?u=126888530
10	MD4283	Prompt Engineering Skills	https://www.linkedin.com/learning/paths/develop-your-prompt-engineering-skills?u=126888530
11	MD4284	Essentials in Generative AI	https://www.linkedin.com/learning/paths/career-essentials-in-generative-ai-by-microsoft-and-linkedin?u=126888530
12	MD4285	Python in Finance	https://www.linkedin.com/learning/paths/python-for-data-professionals-in-finance?u=126888530
13	MD4286	Understanding Quantum Computing	https://www.linkedin.com/learning/paths/understanding-quantum-computing?u=126888530
14	MD4287	Foundational Maths for Machine Learning	https://www.linkedin.com/learning/paths/foundational-math-for-machine-learning?u=126888530

Syllabus Template **FF No. : 654**

CS4217:: Human-Computer Interaction (HCI)

Credits: 02 **Teaching Scheme Theory: 2 Hours/Week**

Course Prerequisites: NA

Course Objectives:

1. To categorize IT applications based on measurable human factors,
2. To study the user community through user survey and/or field visit,
3. To design user-friendly user interfaces with due consideration of interface theory and principles,
4. To apply usability evaluation methods to identify the usability issues with IT applications,
5. To understand the kind of documentation required for IT applications,
6. To integrate web and mobile app design approaches as per user requirements.

Course Relevance:

Human-Computer Interaction (HCI) is a multi-disciplinary socio-technical course, with a goal of bringing the power of computers and communication systems to users, customers or people. It aims to make all computing and communications systems more accessible, maintainable and useful in working, learning and recreational life of their users. It helps every computing, web or mobile application to become really user-centered, increasing its number of users as well as related sales.

Section 1: Topics / Contents

Unit-I Fundamentals of Human Computer Interaction: **05 Hours**

Definition of HCI, Interdisciplinary Nature, Related Disciplines, Usability, Types of Usability, User Interface (UI), Measurable Human Factors, Accessibility, Differently-abled Users.

Unit-II Interaction Concepts and Models: **05 Hours**

User Persona, User Categorization, Golden Rules of Interface Design, Miller's Principle, Task Analysis - GOMS, Contextual Inquiry, Work Models, Interaction Styles.

Unit-III Design Process: **04 Hours**

Design Concept, Three Pillars of Design, Process of Design, Ethnographic Observations, Participatory Design, Internationalization.

Section 2: Topics/Contents

Unit-IV Usability Evaluation: **05 Hours**

Expert-based Evaluation, User-based Evaluation, Formative Evaluation, Summative Evaluation, Heuristic Evaluation, Cognitive Walkthrough, Semiotic Analysis, Icon Categorization, User Surveys, Interviews, Usability Testing.

Unit-V Documentation and Groupware: 05 Hours

Classification of Documents, Reading from Displays, Online Help, Tutorials, Error / Warning Messages, Groupware, Computer Supported Cooperative Work (CSCW), Dimensions of Cooperation, Challenges with Online Communications.

Unit-VI Website and Mobile App Design: 04 Hours

Content Design, Interaction and Navigation Design, Presentation Design, Differences in Design Approaches, Design and Evaluation Tools.

List of Home Assignments:

Design:

6. Social Network for Spiritual Users
7. App for Alzheimer's disease
8. Health Tracking App
9. Ration Card Management App
10. Innovative e-Commerce Platform

Case Study:

6. Chatbot in Healthcare Domain
7. Best Food Ordering App in India
8. Online Teaching-Learning Process
9. Use of Twitter with Indian Users
10. User Experience with Car Booking Apps

Blog

6. Noise of Notifications
7. Challenges in Food Delivery Service
8. Need for Accessibility Guidelines
9. Usability of Autonomous Vehicles
10. Failure of Usability Testing

Surveys

7. User Experience with Video-Conferencing Apps
8. User Errors on Social Networking Sites (SNS)
9. Challenges for Hearing Impaired Users with IT Applications
10. Most Popular Indian Mobile Apps (Made In/By India)

11. Impact of Ban on Chinese Apps in India

Assessment Scheme: Ensures 360 degree assessment and covers all aspects of Bloom's Taxonomy.
MCQ Exam – Section I - Mid Semester 30 Marks converted to 30 equivalent Marks
Home Assignment - End of Semester 100 Marks converted to 10 equivalent Marks
MCQ Exam – Section II - End of Semester 30 Marks converted to 30 equivalent Marks
Comprehensive Viva Voce -End of Semester 100Marks converted to 30 equivalents Marks

Text Books: ***(As per IEEE format)***

1. Ben Shneiderman, "Designing the User Interface", Third Edition, Pearson Education, ISBN 81-7808-262-4.
2. Alan Dix, Janet Finlay, Gregory D. Abowd, Russell Beale, "Human-Computer Interaction", Third Edition, Pearson Education, ISBN 81- 297-0409-9.

Reference Books: ***(As per IEEE format)***

1. Donald Norman, "The Design of Everyday Things", 2002 Edition, Basic Books, ISBN 100-465-06710-7.
2. Wilbert Galitz, "The Essential Guide to User Interface Design", Second Edition, Wiley-Dreamtech India (P) Ltd., ISBN 81-265-0280-0.
3. John Carroll, "Human-Computer Interaction in the New Millennium", Pearson Education, ISBN 81-7808- 549-6.

MOOCs Links and additional reading material:

www.nptelvideos.in, www.coursera.com, www.udemy.com

Course Outcomes: Students will be able to

1. Appreciate the differences among IT applications and their categories based on measurable human factors.
2 Study the user community through user survey and/or field visit.
3. Design user-friendly user interfaces as per user requirements and UI design principles.
4. Apply a suitable usability evaluation method to identify the usability issues.
5. Understand the kind of documentation required for IT applications.
6. Enhance UI designs as per desired web or mobile app design approach.

	Program Outcomes (PO)												PSO			
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO 8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4
CS4217.1	2	**3**	-	-	-	-	-	-	-	-	-	2	-	-	-	2
CS4217.2	2	3	-	**2**	-	-	-	-	-	-	-	2	-	-	-	2
CS4217.3	3	2	3	-	2	3	2	**2**	2	-	-	2	-	-	3	2
CS4217.4	2	3	3	2	-	-	-		-	-	-	2	-	-	2	2
CS4217.5	3	-	-	-	-	-	-	-	-	2	**1**	-	-	-	**2**	2
CS4217.6	3	2	3	2	2	3	2	2	2	-	-	-	-	-	**3**	**2**
Average	**2.5**	**2.6**	**3.0**	**2.0**	**2.0**	**3.0**	**2.0**	**2.0**	**2.0**	**2.0**	**1.0**	**2.0**	**-**	**-**	**2.5**	**2.0**

CO Attainment levels:

Weights for attainment levels: L1 - Easy- 0.75 L2 - Comfortable - 0.7 L3 – Medium – 0.65
L4 – Somewhat difficult – 0.6 L5 – Difficult – 0.55

CO1 – L3, CO2 – L3, CO3 – L2, CO4 – L2, CO5 – L1 and CO6 – L3

Future Course Mapping:

User Interface Design, Usable Security & Intelligent User Interfaces

Job Mapping:

UI Designer, Product Designer, Software Engineer, Mobile App Developer and Other Jobs

Syllabus Template **FF No. : 654**

CS4209:: Parallel Computing

Credits: 02 **Teaching Scheme Theory: 2 Hours/Week**

Course Prerequisites: Computer Organization, Operating System, Design & Analysis of Algorithms, Data Structure

Course Objectives: Students will be able to

1. To introduce the basic concepts of parallel computing
2. To understand various GPU Architecture.
3. To write CUDA programs for parallel implementation
4. To organize the memory management in GPU
5. 5.To optimize parallel programs on GPU using CUDA
6. To solve the scientific problems using GPUs

Course Relevance: Parallel computing, on the other hand, uses multiple processing elements simultaneously to solve a problem. This is accomplished by breaking the problem into independent parts so that each processing element can execute its part of the algorithm simultaneously with the others. This course is required in the industry & used to set up data centers.

Section 1: Topics / Contents

Unit-I Introduction to Parallel Computing : 06 Hours

Motivating Parallelism, Scope of Parallel Computing, Parallelism vs Concurrency, Types and levels of parallelism, Flynn's classification, Amdhal's law; Parallel computer architectures : PRAM, Distributed memory systems,Shared memory systems and cache coherence. Concept of thread and process, programming parallel computers, Parallel computing architectures, interconnection networks, Modern GPU architecture (in brief), Performance comparison: Speedup, Gain time and scalability

Unit-II Parallel Programming Model : 04 Hours

Common Unified Device Architecture (CUDA), CUDA programming model, Concept of grid, block and thread, thread index generation, warp, kernel & kernel launch

Unit-III CUDA APIs : 04 Hours

Programming for GPU's in C/C++ using CUDA APIs: Memory transfers, Writing and executing kernel functions, Writing device functions, Thread synchronization, Data Dependences and Race Conditions, Organizing Parallel Threads.

.

Section 2:Topics/Contents

Unit-IV GPU Architecture: 04 Hours

GPU architecture, Overview of the graphics pipeline, Components of GPU: Parallel streaming processors, Multiprocessors, Shared instruction caches, Memory hierarchy – Global, Constant, Shared, and Texture memory; Case studies: NVIDIA Kepler K20/K40/K80/GP100/GV100/ Ampere / Hopper

Unit-VMemory Organization and Optimization: 06 Hours

Global, Shared, constant and texture memory. Memory coalescing, memory banks and bank conflicts, Page locked host memory. Reduction operation, CUDA code optimization. Need of profilers and analyzers, Introduction to CUDA Tools: MemCheck, Command line & Visual Profilers.

Unit-VI Problem solving using GPUs : 04 Hours

Single vs. double precision, light weight scientific computing exercises, Image processing applications, Matrices etc .

List of Home Assignments:

Design:

11. Parallelizing Search Trees for Chess
12. Parallel Algorithm for Searching
13. Parallel Algorithm for sorting
14. Parallel Algorithm for Data mining
15. Parallel Algorithm for Image Processing

Case Study:

11. Nvidia DGX2
12. Jetson nano Developer Kit
13. GPU Accelerated Apache Spark
14. The Jetson Xavier NX Developer Kit
15. NVIDIA Ampere architecture

Blog

11. Cuda library
12. Turing mesh shaders
13. Low level GPU Virtual memory management
14. Memory Hierarchy of GPU
15. Comparison of Various GPUs

Surveys

12. Smart Hospitals through AI with GPUs
13. Clara Models to help fight with COVID 19
14. GPU Accelerated Molecular Dynamics Applications
15. Medical Imaging applications of GPU
16. Ray Tracing Applications of GPU

Assessment Scheme: Ensures 360 degree assessment and covers all aspects of Bloom's Taxonomy.
MCQ Exam – Section I - Mid Semester 30 Marks converted to 30 equivalent Marks
Home Assignment - End of Semester 100 Marks converted to 10 equivalent Marks
MCQ Exam – Section II - End of Semester 30 Marks converted to 30 equivalent Marks
Comprehensive Viva Voce -End of Semester 100Marks converted to 30 equivalents Marks

Text Books: ***(As per IEEE format)***

1. *Ananth Grama, Anshul Gupta, George Karypis, and Vipin Kumar; Introduction to parallel computing; second edition., Addison- Wesley, 2003, ISBN: 0201648652*
2. *David Kirk, Wen-meiHwuCUDA: Programming Massively Parallel Processors: A Hands-On Approach. © ELSEVIER Inc. 3Jason Sanders and Edward KandrotCUDA by Example: An Introduction to General-Purpose GPU Programming"*

Reference Books: ***(As per IEEE format)***

1. *Hwang and Briggs,"Computer Architecture and Parallel Processing", Tata McGraw Hill Publication ISBN 13: 9780070315563.*
2. *John Cheng, Max Grossman, Ty McKercherProfessional CUDA C Programming,*
3. *CUDA C PROGRAMMING GUIDE by NVIDIA*

MOOCs Links and additional reading material:

www.nptelvideos.in, www.coursera.com, www.udemy.com, www.developer.nvidia.com

Course Outcomes: Students will be able to

1. Recognize various parallel computing architectures and their fundamentals
2. Investigate parallel solutions to complex real world problems
3. Code the parallel programs on GPU using CUDA
4. Realize GPU Architecture.

5. Optimize the parallel programs on GPU using CUDA
6. Design and develop new solutions to research problems

CO-PO Map:

	Program Outcomes (PO)												PSO			
CO/PO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**	**PSO3**	**PSO4**
CO1			3													
CO2					3											
CO3							2									
CO4											1					
CO5												1				
CO6															3	
Average	**0**	**0**	**3.0**	**0**	**3.0**	**0**	**2.0**	**0**	**0**	**0**	**1.0**	**1.0**	**0**	**0**	**3.0**	**0**

CO Attainment levels:

Weights for attainment levels: L1 - Easy-0.75 L2 - Comfortatble-0.7 L3 – Medium – 0.65
L4 – Somewhat difficult – 0.6 L5 – Difficult – 0.55

CO1 – L3, CO2 – L3, CO3 – L2, CO4 – L1, CO5 – L1 and CO6 – L3

Future Course Mapping:
High Performance Computing , Distributed Computing

Job Mapping:
Full Stack Architect-GPU , HPC GPU Application Developer & Consultant , GPU Programming Professional , GPU Performance Analysis Lead / Architect ,
GPU Advocate Associate

Syllabus Template **FF No.: 654**

CS4222:Image Processing

Credits: 2 **Teaching Scheme Theory: 2 Hours/Week**

Course Prerequisites: Digital logic Design, Microprocessor, Computer Organization.

Course Objectives:

1.To describe different color models and image processing techniques.
2.To analyze image condition and deduce enhancement algorithms.
3.To apply image segmentation to identify the region of interest
4.To develop an algorithm to recognize the specified objects in the given image.
5.To study different image morphological operation.
6.To learn different image compression techniques.

Course Relevance:

Vision sense is the most powerful human sense organ. In the world where intelligent automation is taking place, image processing is a vital domain for research and development. In Industry 4.0, image processing systems built around industrial cameras are an essential component in automated production. Throughout all steps of production, from the inspection of raw materials and production monitoring (i.e., flaw detection) to final inspections and quality assurance, they are an indispensable part of achieving high efficiency and quality standards. In the Entertainment Industry, latest trends such as 4K video streaming requires high quality compression that can provide limited/no loss image quality with high fps. In social networking, sharing images has been a vital part. Creating innovative effects and overall manipulating the images will be explored

Section 1: Topics/Contents

Unit-I Introduction **4 Hours**

Elements of image processing system, Scenes and Images, Vector Algebra, Human Visual System, color vision color model: RGB, HVS, YUV, CMYK, YCbCr and some basic relationships between pixels, linear and nonlinear operations, Image sampling and quantization.

Unit-II Image Enhancements **5 Hours**

Memory-less operations, Spatial domain image enhancements: Denoising filters, Smoothing Operation, Sharpening Operation, and Contrast stretching /enhancement, histogram and histogram equalization.

Unit-III Image segmentation **5 Hours**

Classification of image segmentation techniques: Edge-based Segmentation, Region based techniques. Binarization: Global Thresholding, Adaptive thresholding. Types of Edge detector: derivative filters, Sobel, Canny. Edge linking. Feature Extraction.

Section2:Topics/Contents

Unit-IV Morphological Operation **4 Hours**

Binary Morphology, Erosion Dilation, Opening and Closing.

Unit-V Feature Extraction and Object Recognition **5 Hours**

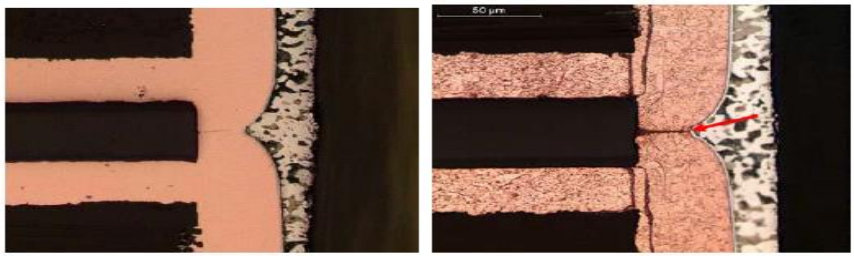
Feature points and feature detection (Line, circle and corner). Line detection: RANSAC, Hough Transform. Corner detection: Harris Corner Detector. Feature descriptors, Descriptor matching. SIFT, Boundary representation (Chain code), Boundary detection-based techniques.

Unit-VI Image Compression **5 Hours**

Introduction and need, Coding redundancy, classification of compression techniques (Lossy and lossless- JPEG, Run Length Coding, Huffman Coding, Shannon Fano coding).

List of Design based Home Assignments:

1.Design an algorithm to identify fault in a “PCB inspection system” as shown below



2.Design an algorithm to perform segmentation of the image below to extract the mango from its background.

3.Design an algorithm to get from image 1 to image 2.

4.Design an algorithm to recognize character “0” in the image below.

5.Design an algorithm to compress a 300x300 pixel image with horizontal black to white gradient as shown below.

List of Case Study based Home Assignments:

1. Cam-scanner: Document scanning app
2. Tesseract OCR library
3. Instagram filters
4. OpenCV
5. Google Street View

List of Blog based Home Assignments:

1. Image processing on Embedded platforms
2. Face recognition system security analysis for authentication
3. Image processing in MSME for effective automation
4. H.264 codec for image streaming
5. Role of mathematics in image processing

List of Survey based Home Assignments:

1. Image quality metrics
2. Vision based self-driving car safety
3. Compression techniques & codecs
4. State of the art applications such as AR/ XR
5. Human recognition in social networking apps like Facebook

Assessment Scheme: Ensures 360 degree assessment and covers all aspects of Bloom's Taxonomy.
MCQ Exam – Section I - Mid Semester 30 Marks converted to 30 equivalent Marks
Home Assignment - End of Semester 100 Marks converted to 10 equivalent Marks
MCQ Exam – Section II - End of Semester 30 Marks converted to 30 equivalent Marks
Comprehensive Viva Voce -End of Semester 100Marks converted to 30 equivalents Marks

Text Books: ***(As per IEEE format)***

1. *Rafael Gonzalez & Richard Woods, "Digital Image Processing," 3rd Edition, Pearson publications, ISBN 0132345633.*
2. *Anil K. Jain, "Fundamental of Digital Image Processing," 5th Edition, PHI publication, ISBN 13: 9780133361650.*

Reference Books: ***(As per IEEE format)***

1.*Pratt, "Digital Image Processing," Wiley Publication, 3rdEdition, ISBN 0-471- 37407-5.*
2. *K.R. Castleman, "Digital Image Processing,"3rd Edition, Prentice Hall: Upper Saddle River, NJ, 3, ISBN 0-13-211467 -4.*

MOOCs Links and additional reading material:

1. https://nptel.ac.in/courses/117/105/117105135/
2. https://nptel.ac.in/courses/106/105/106105032/
3. https://nptel.ac.in/noc/courses/noc19/SEM2/noc19-cs58/
4. https://www.coursera.org/learn/computer-vision-basics

Course Outcomes:

The student will be able to –
1.Recognize different color models and image processing techniques. (1)
2.Select image enhancement algorithm to improve the quality of image. (2)

3.Build image segmentation techniques to identify region of interest. (4)
4.Predict image morphological techniques to resize the image. (3)
5.Construct an algorithm to recognize the specified objects in the given image. (5)
6. Identify different image compression techniques to reduce the size of image. (3)

CO-PO Map:

	Program Outcomes (PO)												PSO			
CO/PO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**	**PSO3**	**PSO4**
CO1	3	2		1	-	-	-	-	-	-	-	2	2	2	3	3
CO2	3	2	2	1	3	-	-	2	-	-	-	2	2	2	3	3
CO3	2	2	3	2	3	-	-	-	1	-	-	2	2	2	3	3
CO4	3	3	3	2	3	2	-	-	-	-	-	2	2	2	3	3
CO5	3	2		2	3	2	-	-	1	1	2	2	2	2	3	3
CO6	3	2		2	3	2	2	2		1	2	2	2	2	3	3
Average	**2.8**	**2.16**	**2.7**	**1.88**	**3**	**2.0**	**2.0**	**2.0**	**1.0**	**1.0**	**2.0**	**2.0**	**2.0**	**2.0**	3.0	3.0

CO Attainment levels:

Weights for attainment levels: L1 - Easy-0.75 L2 - Comfortatble-0.7 L3 – Medium – 0.65
L4 – Somewhat difficult – 0.6 L5 – Difficult – 0.55

CO1 – L2, CO2 – L2, CO3 – L4, CO4 – L3, CO5 – L5 and CO6 – L3

Future Course Mapping:

1. Augmented Reality
2. Multimedia Processing

Job Mapping:

1. Augmented Reality Experience Designer
2. Automation Engineer
3. Embedded Software Developer
4. Image Processing Expert

Syllabus Template **FF No. : 654**

CS4275: Introduction to Machine Learning

(https://onlinecourses.nptel.ac.in/noc24_cs101/preview - offered by IIT Madras)

Credits: 2

Course Prerequisites:

Statistical Mathematics, Artificial Intelligence

Course Relevance:

With the increased availability of data from varied sources there has been increasing attention paid to the various data driven disciplines such as analytics and machine learning. In this course we intend to introduce some of the basic concepts of machine learning from a mathematically well motivated perspective. We will cover the different learning paradigms and some of the more popular algorithms and architectures used in each of these paradigms..

All students have to complete this course from NPTEL-SWAYAM under the guidelines provided by dean academics and to be coordinated by department coordinator assigned for this course.

Course Outcomes:

1. Understand the features of machine learning to apply on real world problems.-01
2. To develop various supervised machine learning algorithms in a wide range of real-world applications.-02
3. Understand a wide variety of Statistical learning algorithms and apply on different applications-03
4. Analyze the concept of Support Vector Machines, Naive Bayes Classifiers for learning linear and non-linear Classifiers.-05
5. To analyze Decision Tree algorithms to solve problems of real world -04
6. Demonstrate understanding of unsupervised learning, Reinforcement learning and their applications.-03

CO-PO Mapping:

	Program Outcomes (PO)												PSO			
CO/PO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**	**PSO3**	**PSO4**
CO1	3	3	3	2	3	1	0	1	1	1	0	2	2	2	3	0
CO2	3	3	3	2	3	2	0	1	1	1	0	2	2	3	3	0
CO3	2	3	3	3	3	2	0	1	1	1	0	2	2	3	3	0
CO4	3	3	3	3	3	2	0	1	1	1	0	2	2	3	3	0
CO5	3	3	3	3	3	2	0	1	1	1	0	2	2	3	3	0
CO6	3	3	3	3	3	2	0	1	1	1	0	2	2	3	3	0
Average	**2.83**	**3**	**3**	**2.67**	**3**	**2**	**0**	**1**	**1**	**1**	**0**	**2**	**2**	**2.83**	**3.0**	**0**

Attainment Levels: 1,2,3,5,4,3

Syllabus **Template** **FF No. : 654**

CS4276: Deep Learning

(https://onlinecourses.nptel.ac.in/noc24_cs114/preview - offered by IIT Ropar**)**

Credits: 2

Course Prerequisites:

Statistical Mathematics, Artificial Intelligence, Machine Learning

Course Relevance:

The availability of huge volume of Image and Video data over the internet has made the problem of data analysis and interpretation a really challenging task. Deep Learning has proved itself to be a possible solution to such Computer Vision tasks. Not only in Computer Vision, Deep Learning techniques are also widely applied in Natural Language Processing tasks. In this course we will start with traditional Machine Learning approaches, e.g. Bayesian Classification, Multilayer Perceptron etc. and then move to modern Deep Learning architectures like Convolutional Neural Networks, Auto encoders etc. On completion of the course students will acquire the knowledge of applying Deep Learning techniques to solve various real life problems.

All students have to complete this course from NPTEL-SWAYAM under the guidelines provided by dean academics and to be coordinated by department coordinator assigned for this course.

Course Outcome:

1. Demonstrate understanding of a logistic regression model, structured as a shallow Neural network.
2. Build and train a deep Neural Network.
3. Apply techniques to improve neural network performance.
4. Demonstrate understanding of functionality of all layers in a convolutional neural network.
5. Implement convolutional networks for image recognition/classification tasks.
6. Demonstrate Understanding of Recurrent nets and their applications.

CO-PO Mapping:

	Program Outcomes (PO)												PSO			
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4
CO1	3	3	3	2	3	1	0	1	1	1	0	2	0	3	3	0
CO2	3	3	3	2	3	2	0	1	1	1	0	2	0	3	3	0
CO3	2	3	3	3	3	2	0	1	1	1	0	2	0	3	3	0
CO4	3	3	3	3	3	2	0	1	1	1	0	2	0	3	3	0
CO5	3	3	3	3	3	2	0	1	1	1	0	2	0	3	3	0
CO6	3	3	3	3	3	2	0	1	1	1	0	2	0	3	3	0
Average	3	3	3	2.67	3	1.83	0	1	1	1	0	2	0	3	3	0

Attainment Levels: 3,3,5,4,5,4

Syllabus Template **FF No. : 654**

CS4225, CS4226: Major Project

Credits:.13..... **Teaching Scheme Theory: …26... Hours/Week**

Course Prerequisites: Project Based Learning

Aim

This course addresses the issues associated with the successful management of a project. The course emphasizes project life cycle phases requirement engineering, system analysis and system design. A further aim is for students to heighten personal awareness of the importance of developing strategies for themselves and working with peers to create desired outcomes. The Project Work can lead to:

- Transform existing Ideas into conceptual models.
- Transform conceptual models into determinable models.
- Use determinable models to obtain system specifications.
- Select optimum specifications and create physical models.
- Apply the results from physical models to create real target systems.

Project Group and Topic Selection and Synopsis:

The project work needs to be undertaken by a group of maximum FOUR and minimum of THREE students. The Project work will be jointly performed by the project team members. The student needs to identify a technological problem in the area of Computer Engineering or Information Technology of their choice and address the problem by formulating a solution for the identified problem. The Project Group will prepare a synopsis of the project work which will be approved by the concerned faculty member. The project should not be a reengineering or reverse engineering project. In some cases, reverse engineering projects will be permissible based on the research component involved in it. The project work aims at solving a real world technical problem. Hence ample literature survey is required to be done by the students. Application-oriented projects will not be acceptable. Low-level custom User Interface development and its allied mapping with a particular technology will not be accepted.

Overview of the Course:

1. The Student Project Group is expected to make a survey of situation for identifying the requirements of selected Technological Problem. The Student Project Group will be monitored by Internal Guides and External Guides (if any).
2. The project requires the students to conceive, design, implement and operate a mechanism (the design problem). The mechanism may be entirely of the student's own design, or it may incorporate off-the-shelf parts. If the mechanism incorporates off-the-shelf parts, the students

must perform appropriate analysis to show that the parts are suitable for their intended purpose in the mechanism.

3. The project must be based on a Fresh Idea or Implementation of a Theoretical Problem – meaning that there is not a known Solution to the design problem Or Create a Better Solution.
4. The project must have an experimental component. Students must conceive, design, implement and operate an appropriate experiment as part of the project. The experiment might be to collect data about some aspect of the design (i.e., to verify that the design will work as expected). Alternatively, the experiment could be to verify that the final mechanism performs as expected.
5. Upon receiving the approval, the Student Project Group will prepare a preliminary project report consisting , Feasibility Study Document, System Requirement Specification, System Analysis Document, Preliminary System Design Document. All the documents indicated will have a prescribed format.

6. Upon project completion, the Student Project Group will prepare a detailed Project Report consisting Semester I Preliminary Project document along with Detailed System Design Document, Implementation and Testing Document with conclusion and future scope of the Project Work. All the documents indicated will have a prescribed format. The Project Report ideally should consist of following documents : (Exceptions may be there based on the nature of the project, especially if some of the following documents are not applicable to a particular project as determined by the project guide, coordinator and head of department).

Sr.	Project Item
1	Project Cover Front Page
2	Project Completion Certificate [Institute]
3	Project Completion Letter [In case of Sponsored Projects]
4	Acknowledgments
5	Table of Contents
6	List of Figures
7	List of Tables
8	Project Synopsis [Problem Background, Existing System Details, Proposed Solution]
9	Feasibility Study Report
10	Project Plan
11	System Requirement Specification
12	System Analysis Document: UML Use Case Diagrams
13	System Analysis Document: UML Sequence Diagrams
14	System Analysis Document: UML State Diagrams
15	System Design Document with Module Specifications

16	System Implementation
17	System Testing and Experimental Findings
18	Conclusion
19	References

7. The Project Work will be assessed jointly by a panel of examiners consisting faculty and industry experts. The Project Groups will deliver the presentation and demonstration of the Project Work which will be assessed by the panel.
8. The Student Project Group needs to actively participate in the presentation. The panel of examiners will evaluate the candidate's performance based on presentation skills, questions based on the Project Work and overall development effort taken by the candidates.

Note:
The student needs to design and develop solution for the identified technological problem in the area of Computer Engineering or Information Technology of their choice. The Project Implementation needs to be completed using best possible use of available technologies as applicable to deal with the complexity of the project. The Project Group will prepare a detailed report of the project work which will be approved by the concerned faculty member. The Project Report need to be submitted both in Hard form and Soft form in CD. The Soft Copy of the Project Report must accompany other project deliverables as well.

Assessment: MSE and ESE
1. Mid Semester Assessment – 50 Marks to be converted to 30 Marks.
2. End Semester Assessment – 100 Marks to be converted to 70 Marks.

Mid Semester Assessment

Sr. No.	Parameter	Marks
1	Problem Statement	10
2	Literature Review	10
3	Group formation and identification of individual responsibility	10
4	Objective of Project activity	10
5	Knowledge of domain, latest technology and modern tools used /to be used	10
	TOTAL	**50**

End Semester Assessment

Sr. No.	Parameter	Marks
1	Realization of project as per problem statement	10
2	Design, Testing / Experimentation, Analysis / Validation	30
3	Documentation and Report Writing	20
4	Quality of Work	15
5	Performance in Question & Answers Session	15
6	Regular interaction with guide	10
	TOTAL	**100**

Course Outcomes:

Upon completion of the course, graduates will be able to -

CO1: Identify the real life problem from societal need point of view

CO2: Prepare the requirement engineering, feasibility analysis documents

CO3: Form the teams and share responsibilities according to individual skill strengths

CO4: Create design documents to build software solutions

CO5: Develop software solutions based on standard engineering specifications

CO6: Perform the verification and validation up to the mark

CO PO Map

	Program Outcomes (PO)												PSO			
CO/PO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**	**PSO3**	**PSO4**
CO1	2	3		2					3			3				3
CO2	2	3	3	2	2				3	3	2	3	3		3	3
CO3	2	-	-	-	-				3		2	3	3			3
CO4	2	3	3	2	2	3	3.0	2.0	3	3	2	3	3	3	3	3
CO5	2	3	3	2	2				3	3	2	3	3	3	3	3
CO6	2	2	2	3	2				3	2	2	3	3	2	3	3
Average	**2.0**	**2.8**	**2.75**	**2.83**	**2.0**	**3.0**	**3.0**	**2.0**	**3.0**	**2.75**	**2.0**	**3.0**	**3.0**	**1.75**	**3.0**	**3.0**

CO attainment levels

CO1 -4 CO2 –2 CO3-4 CO4-5 CO5 -1 CO6-3

Syllabus Template **FF No. : 654**

CS4292: Design Thinking 7

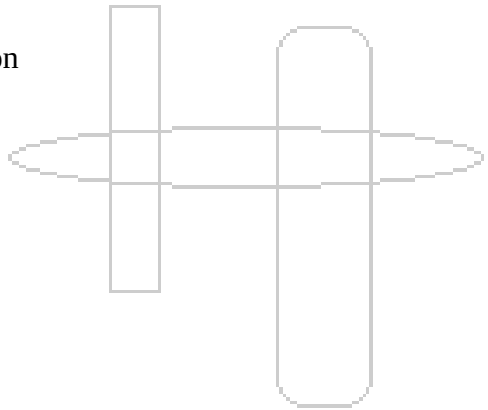
Credits: 01 **Teaching Scheme: Tutorial 01 Hr/week**

Course Prerequisites: Problem Based Learning, Project Centric Learning

Course Objective:
To provide ecosystem for students and faculty for paper publication and patent filing

Section 1: Topics/Contents

What is research?
Importance of Paper Publication and Patents
Structure of Paper
Journal Publication
Publication in conference
Literature Review
Research Paper Writing
Journal Ratings and Evaluation
How to rate a Journal?
Intellectual property (IP)
Research Ethics
Entrepreneurship



Section 2: Topics/Contents

Structure of The paper
Journal List (Top 50 Journals)
Selection of the journal
Use of various online journal selection tools
Plagiarism checking
Improving contents of the paper
Patent drafting
Patent search
Filing of patent
Writing answers to reviewer questions
Modification in manuscript
Checking of publication draft

Course Outcome: **[**Publication of paper or patent]
The student will be able to
1. Understand the importance of doing Research

2. Interpret and distinguish different fundamental terms related to Research
3. Apply the methodology of doing research and mode of its publication
4. Write a Research Paper based on project work
5. Understand Intellectual property rights
6. Use the concepts of Ethics in Research
7. Understand the Entrepreneurship and Business Planning

CO-PO Map:

	Program Outcomes (PO)												PSO			
CO/PO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**PSO1**	**PSO2**	**PSO3**	**PSO4**
CO1	1	1	1	1	1	--	--	--	--	--	--	1	1	2	2	3
CO2	1	1	1	1	1	--	--	--	--	--	--	1	2	1	1	3
CO3	2	2	3	3	2	2	1	2	2	3	--	1	2	2	3	3
CO4	3	3	3	3	3	2	1	2	2	3	1	1	-	-	2	3
CO5	1	1	1	1	1	--	--	--	--	--	--	1	-	-	1	2
CO6	2	2	2	2	2	2	1	3	2	3	--	1	2	2	2	3
C07	1	1	1	1	1	--	--	--	--	--	--	1	1	1	1	1
Average	**1.57**	**1.57**	**1.71**	**1.71**	**1.57**	**2.0**	**1.0**	**2.33**	**2.0**	**3.0**	**1.0**	**1.0**	1.66	1.66	1.71	2.5

Syllabus Template **FF No. : 654**

CS4236, CS4238, CS4232, CS4202: Industry Internship, Research Internship

Credits:.15..... **Teaching Scheme Laboratory: …32... Hours/Week**

Guidelines:

Internships are educational and career development opportunities, providing practical experience in a field or discipline. Internships are far more important as the employers are looking for employees who are properly skilled and having awareness about industry environment, practices and culture. Internship is structured, short-term, supervised training often focused around particular tasks or projects with defined time scales. Core objective is to expose technical students to the industrial environment, which cannot be simulated/experienced in the classroom and hence creating competent professionals in the industry and to understand the social, economic and administrative considerations that influence the working environment of industrial organizations. Engineering internships are intended to provide students with an opportunity to apply theoretical knowledge from academics to the realities of the field work/training.

The following guidelines are proposed to give academic credits for the internship undergone as a part of the B.Tech. Engineering curriculum.

Duration:

Industry Internship will be started at the beginning of the semester 7 or semester 8 or yearlong for the duration 6 months or 12 months.

Identification of Internship work:

Student may choose to undergo Internship at Industry/Govt./NGO/MSME/ Innovation/IPR/Entrepreneurship. Contacting various companies for Internship and Internship work identification process should be initiated at the end of 6th semester in coordination with training and placement cell/ industry institute cell/ internship cell. This will help students to start their internship work on time. Students can take internship work in the form of online/onsite work from any of the following but not limited to:

- Working for consultancy/ research project
- Contribution in Incubation/ Innovation/ Entrepreneurship Cell/ Institutional Innovation Council/ start-ups cells of institute
- Industry / Government Organization Internship,

- Internship through Internshala
- In-house product development, intercollegiate, inter department research internship under research lab/group, micro/small/medium enterprise/online internship
- Research internship under professors, IISC, IIT's, Research organizations

Internship Documents Submission:

Students must submit internship offer letter, internship completion letter, FF 1029 (Students feedback form), FF 1030 (Industry feedback about interns).

Students must present their internship progress time to time to faculty mentors. Faculty mentors and industry mentors both can evaluate the progress of the intern combiningly.

Internship Work Evaluation:

In-semester and end semester internship evaluation and assessment will be done by internal (Faculty mentor) and external examiners - a supervisor from industry.

After completion of Internship, the student should prepare a comprehensive report to indicate what he has observed and learnt in the internship/training period. The student may contact Industrial Supervisor/ Faculty Mentor/TPO for assigning special topics and problems and should prepare the final report.

If the student remain absent without prior intimation to the department/institute/concern authority/T & P Cell, his entire training can be cancelled and he will fail.

Course Outcomes: Industry Internship

On the completion of course, students will able to-

1. Understand real-world applications, workplace environment and operating procedures
2. Adapt skill for learning and applying modern tools and technologies
3. Apply professional values and ethical standards
4. Perform as an individual and as a team member effectively to changing conditions
5. Encompass improved writing, verbal communication and documentation skills
6. Learn about career positions and occupations along with the qualities and training required to obtain those positions

CO-PO Map: Industry Internship

	Program Outcomes (PO)												PSO			
CO/PO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**	**PSO3**	**PSO4**
CO1	2	3		2					3			3				2
CO2	2	3	3	2	3	2	3	2	3		2	3	3	3	3	3
CO3	2	3	3	2				2	3		2	3	3	3	3	3
CO4	2								3			3				
CO5	2		2		3			2	3	3	2	3			3	
CO6	2					2			3			3				
Average	**2.0**	**3.0**	**2.66**	**2.0**	**3.0**	**2.0**	**3.0**	**2.0**	**3.0**	**3.0**	**2.0**	**3.0**	**3.0**	**3.0**	**3.0**	**2.66**

CO attainment levels:

CO1 – 3, CO2 –5, CO3 –2, CO4 –2, CO5 – 3, CO6 –2

Course Outcomes: Research Internship

On the completion of course, students will able to-

1. Develop an ecosystem to promote entrepreneurship and research culture among the students.
2. Learn first-hand to apply techniques, resources, modern engineering tools for prediction modelling to complex engineering activities.
3. Apply reasoning informed by the contextual knowledge to access societal, health, safety, legal and cultural issues and consequent responsibilities.
4. Perform as an individual and as a team member.
5. Understand Engineering and Management Principles.
6. Exercise R & D aptitude focusing on the knowledge creation and dissemination through engineering artifacts creation, construction and presentation.

CO-PO Map: Research Internship

	Program Outcomes (PO)												PSO			
CO/PO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**	**PSO3**	**PSO4**
CO1	2											3				3
CO2	3	3	3	2	3	2	2	2				3	3	3	3	3
CO3	3	3	3	2		3						3	3		3	3
CO4									3			3				3
CO5	3	2	3	2								3	3		3	3
CO6	3	3	2	3			3	2.0		3.0	2.0	3			3	3
Average	**2.8**	**2.75**	**2.75**	**233**	**3.0**	**2.5**	**2.5**	**2.0**	**3.0**	**2.75**	**2.0**	**3.0**	**3.0**	**3.0**	**3.0**	**3.0**

CO1 – 3, CO2 –4, CO3 –5, CO4 –2, CO5 – 3, CO6 –4