Semester-V

Course (Category) Code	Course Name	Teaching Scheme (Hrs/week)					Credits Assigned			
		L	T	P	O	E	L	T	P	Total
(PC)	Theory of Computation	3	0	0	6	9	3	0	0	3
		Examination Scheme								
		Component		ISE		MSE		ESE		Total
CS301/IT301		Theory		75		75		150		300
		Laboratory		--		--		--		--

Pre-requisite Course Codes, if any.	CS201/IT201: Discrete Structures and Graph Theory
Course Objective: To give an overview of the theoretical foundations of computer science from the perspective of formal languages which provides the mathematical foundation of formal models of computation, and fundamentals of formal grammars and languages that is used in most areas of computer science.	
Course Outcomes (CO): ***At the end of the course students will be able to***	
CS301.1	Design finite automaton for a regular expressions and languages.
CS301.2	Apply the properties of regular languages.
CS301.3	Construct the grammar for a language and convert it into normal forms.
CS301.4	Design and Evaluate Pushdown Automata and Turing Machine for a language.

Theory Component

Module No.	Unit No.	Topics	Ref.	Hrs.
1	Title	**Sets, Relations and Languages**	1,5	3
	1.1	Relations and functions		
	1.2	Alphabets and languages		
	1.3	Types of proof		
2	Title	**Finite Automata**	1,3,5	7
	2.1	Regular languages and regular expressions		
	2.2	Finite Automata, Nondeterministic Finite Automata, Nondeterministic Finite Automata with ϵ-transitions		
	2.3	Kleene’s theorem		
	2.4	NFA to DFA Conversion		
	2.5	Finite Automata with output (Moore and Mealy Machine)		
3	Title	**Regular Languages**	1,4	6
	3.1	The pumping lemma for regular languages, Applications of the pumping lemma		
	3.2	Closure properties for regular languages		
	3.3	Equivalence and minimization of automata: Testing equivalence of states, Minimization of DFA’s		
4	Title	**Context-Free Grammars and Languages**	1,5	5

	4.1	Context free grammars: Definition of context free grammars, Derivations using a grammar, The language of a grammar, Sentential forms		
	4.2	Parse trees: Constructing parse trees, From inferences to trees, From trees to derivations, From derivations to recursive inferences		
	4.3	Ambiguity in grammars and languages: Ambiguous grammars, Removing ambiguity from grammars		
5	**Title**	**Pushdown Automata**	1,2	**6**
	5.1	Definition of the pushdown automaton: The formal definition of pushdown automata, A graphical notation for PDA's, Instantaneous descriptions of a PDA		
	5.2	The languages of a PDA: Acceptance by final state, Acceptance by empty stack, From empty stack to final state, From final state to empty stack		
	5.3	Equivalence of PDA's and CFG's: From grammars to pushdown automata, From PDA's to Grammar		
	5.4	Deterministic pushdown automata: Definition of a deterministic PDA, Regular languages and deterministic PDA's, DPDA's and context free languages		
6	**Title**	**Properties of Context-Free Languages**	1,2,3	**5**
	6.1	Eliminating useless symbols, Computing the generating and reachable symbols, Chomsky normal form, Greibach normal form		
	6.2	The Pumping lemma for context free languages: Applications of the pumping lemma for CFL's		
7	**Title**	**Introduction to Turing Machines**	1,2,	**6**
	7.1	Turing machines: Formal definition of a Turing machine, Examples of Turing machines		
	7.2	Halting Problem, Post Correspondence Problem (PCP)		
	7.3	Variants of Turing machines: Multitape Turing Machines		
	7.4	Church-Turing hypothesis		
8	**Title**	**Recursively Enumerable Languages**	3	
	8.1	Recursively Enumerable and recursive		**4**
	8.2	Enumerating a language		
	8.3	Context sensitive languages and the Chomsky hierarchy		
	Self Study	**Tractable and Intractable Problems:** Tractable and Possibly Intractable Problems: P and NP, Polynomial-Time Reductions and NP-Completeness, Cook's Theorem	3	**5***
Total (* Not included)				**42**

Text Books

Sr. No	**Title**	**Edition**	**Authors**	**Publisher**	**Year**
1	Introduction to Automata Theory, Languages, and Computation	Third	John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman	Pearson	2008

2	Introduction to the Theory of computation	Third	Michael Sipser	Cengage	2013

Reference Books

Sr. No	Title	Edition	Authors	Publisher	Year
3	Introduction to Languages and the Theory of Computation	Fourth	John C. Martin	McGraw-Hill	2010
4	Elements of the Theory of Computation	Second	Harry R. Lewis, Christos H. Papadimitriou	Pearson	2015
5	Automata and Computability	--	Dexter C. Kozen	Springer	1997

<table>
<tr><th rowspan="2">Course (Category) Code</th><th rowspan="2">Course Name</th><th colspan="5">Teaching Scheme (Hrs/week)</th><th colspan="4">Credits Assigned</th></tr>
<tr><th>L</th><th>T</th><th>P</th><th>O</th><th>E</th><th>L</th><th>T</th><th>P</th><th>Total</th></tr>
<tr><td rowspan="3"><b>(PC)</b></td><td rowspan="5"><b>Software Engineering</b></td><td>3</td><td>0</td><td>2</td><td>5</td><td>10</td><td>3</td><td>0</td><td>1</td><td>4</td></tr>
<tr><td colspan="9"><b>Examination Scheme</b></td></tr>
<tr><td colspan="2"><b>Component</b></td><td colspan="2"><b>ISE</b></td><td colspan="2"><b>MSE</b></td><td colspan="2"><b>ESE</b></td><td><b>Total</b></td></tr>
<tr><td rowspan="2"><b>CS302/IT302</b></td><td colspan="2"><b>Theory</b></td><td colspan="2"><b>50</b></td><td colspan="2"><b>50</b></td><td colspan="2"><b>100</b></td><td><b>200</b></td></tr>
<tr><td colspan="2"><b>Laboratory</b></td><td colspan="2"><b>50</b></td><td colspan="2"><b>--</b></td><td colspan="2"><b>50</b></td><td><b>100</b></td></tr>
</table>

Pre-requisite Course Codes, if any.	Object-oriented programming language -CS102, DBMS- IT/CS204
Course Objective: To understand the best practices in software engineering and gain knowledge to analyze, design, implement and test software project.	
Course Outcomes (CO): ***At the End of the course students will be able to***	
CS302.1/IT302.1	Analyze software requirements.
CS302.2/IT302.2	Apply UML models for a project.
CS302.3/IT302.3	Evaluate system architecture and develop detailed task schedule from the overall estimates and planning.
CS302.4/IT302.4	Illustrate different coding principles with unit test process.
CS302.5/IT302.5	Understand the need for DevOps.

Theory Component

Mo dule No.	Unit No.	Topics	Ref.	Hr s.
1		**Introduction**		06
	1.1	Software Development Challenges, Software Scope, The Human Side of Software Development	1,2	

	1.2	Software Methodologies and Related Process Models with applications, Traditional Life Cycle Models, Waterfall, Incremental, Iterative models, Agile Software Engineering Process Models, SCRUM, Extreme Programming	1,2	
2		**Requirements Management and Project Planning**		**10**
	2.1	Requirements Development Methodology, Specifying Requirements, Eliciting Accurate Requirements, Documenting Business Requirements, SRS, Defining User Requirements, Validating Requirements, Achieving Requirements Traceability, Managing Changing Requirements, Agile Requirements Engineering	1,2	
	2.2	Scheduling, Work Breakdown Structure, Gantt Chart, Pert Chart, Critical Path, Earned Value Analysis, Schedule & Cost slippage, Estimation, Decomposition techniques, Empirical estimation models, Software Risk Management**:** Risk Identification, Risk Projection, Risk Refinement, RMMM Plan	1,2	
3		**Software Analysis**		**08**
	3.1	Difference between Structured & Object-Oriented analysis, Structured Analysis, Data Flow Diagrams	4,5	
	3.2	Object Oriented Analysis, Uses Case, Class diagram, Interaction diagrams, Activity diagram, State Chart diagram, Component & Deployment diagram	4,5	
4		**Software Design & Development**		**08**
	4.1	Software Architecture, Architectural and Pattern-Based Design, Model Driven Architectures	1,2	
	4.2	Software Development, Component Infrastructures, Refactoring, Test Driven Development (TDD)	1,2	
	4.3	DevOps, Continuous Integration, Continuous Deployment, System Provisioning and Configuration Management	3	
	4.4	Software Change Management, Change Control, Version Control	1,2	
5		**Software Quality & Testing**		**10**
	5.1	Software Quality Concepts, Quality Assurance, Quality Control, Formal Technical Reviews	1,2	
	5.2	Software Metrics, Product Metrics – McCall's Quality Factor, Metrics for Analysis Model and Design Model, Project Metrics, Process Metrics, Metrics for Source Code	1,2	
	5.3	Software Testing, Unit Testing, Integration Testing, System Testing	1,2	
6	**Self Study**	Advance Topic in software Engineering		**05***
		○ Design Pattern		
			Total(* Not included)	**42**

Laboratory Component, if any. (Minimum 10 Laboratory experiments are expected)

Sr. No	**Title of the Experiment**
1	Gather requirements and write a project proposal for case study. Prepare SRS document. (Use IEEE template)
2	Design UML diagram -Use Case, Class diagram
3	Design UML diagram -Interaction diagrams
4	Design Data flow diagram (level 0 and1) for the case study.

5	Create work breakdown structure and schedule the activities
6	Develop Risk Mitigation, Monitoring and Management Plan for the case study.
7	Create versions of software using version control tool.
8	Implement any one Module from chosen case study.
9	Prepare test cases and perform Unit Testing (test scenario, test cases, test data)
10	Study on continuous Integration using DevOp

Text Books

Sr. No	Title	Edition	Authors	Publisher	Year
1	Software Engineering: A Practitioner’s Approach	Ninth Edition	Roger S. Pressman and Bruce Maxim	McGraw-Hill	2019
2	Fundamentals of Software Engineering	Fifth Edition	Rajib Mall	PHI Learning	2018

Reference Books

Sr. No	Title	Edition	Authors	Publisher	Year
3	The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations	--	Gene Kin, Patrick Debois, John Willis, Jez Humble, and John Allspaw	IT Revolution Press	2016
4	UML for Java Programmers	--	Robert C. Martin	Pearson	2006
5	UML Distilled: A Brief Guide to the Standard Object Modeling Language	Third Edition	Martin Fowler	Addition Wesley	2003

Course (Category) Code	**Course Name**	**Teaching Scheme (Hrs/week)**					**Credits Assigned**			
		L	**T**	**P**	**O**	**E**	**L**	**T**	**P**	**Total**
(PC)	**Foundation of Signal Processing**	**3**	**0**	**2**	**5**	**10**	**3**	**0**	**1**	**4**
		Examination Scheme								
		Component	**ISE**	**MSE**	**ESE**	**Total**				
CS303/IT303		**Theory**	**50**	**50**	**100**	**200**				
		Laboratory	**50**	**--**	**50**	**100**				

Pre-requisite Course Codes, if any.	Introduction to Signals and Systems) or equivalent
Course Objective: Foundations of Digital Signal Processing! The study of digital signal processing explores how we transform data into new representations to better understand, compress, and leverage it. The course begins with a rigorous review of tools from Signals and Systems: sampling, convolution, Fourier representations and flow graph, fast linear filtering algorithms. It also comoares DSP Processor and General-Purpose Processor.	
Course Outcomes (CO): ***At the End of the course students will be able to***	
CS303.1/IT303.1	Interpret DT signal and perform signal manipulation in Time Domain and Frequency Domain
CS303.2/IT303.2	Develop FFT flow-graph
CS303.3/IT303.3	Implement Fast Linear filtering algorithms

CS303.4/IT303.4	Compare the DSP processor with General Purpose Processor (GPP)

Theory Component

Module No.	**Unit No.**	**Topics**	**Ref.**	**Hrs.**
1	**Title**	**Discrete-Time Signal**	1, 2	**12**
	1.1	Introduction: Signals, Systems, and Signal, Continuous Time signal, Discrete - Time signal and representation, Digital signal, The Sampling theorem, Some elementary discrete time signals, Classification of Discrete - Time Signals, Modifications of Discrete - Time Signals.	1, 2	
	1.2	Operations on Discrete - Time Signals: Linear Convolution, Circular Convolution, Matrix Representation of Circular Convolution, Linear Convolution using Circular Convolution, Auto and Cross Correlation.	1, 2	
	1.3	Discrete - Time systems: Static and dynamic, time variant and time invariant, linear and nonlinear, causal and non causal. Representation of system using impulse response, Finite Impulse Response (FIR) and Infinite Impulse Response (IIR) system, Response of the FIR system using convolution.	1, 2	
2	**Title**	**Discrete Fourier Transform**	1, 2	**08**
	2.1	Introduction to DTFT, Relation between DFT and DTFT, DFT of DT signal, Inverse DFT.	1, 2	
	2.2	Properties of the DFT: Scaling and Linearity, Symmetry for real valued signal, Periodicity, Time Shift and Frequency Shift, Time Reversal, Convolution Property and Parsevals Energy Theorem.	1, 2	
3	**Title**	**Fast Fourier Transform**	1, 2	**08**
	3.1	Fast Fourier Transform: Need of FFT, Radix-2 DIT-FFT algorithm	1, 2	
	3.2	Flow graph for N=4 and 8 using Radix-2 DIT-FFT, Inverse FFT algorithm, Comparison of complex and real, multiplication and additions of DFT and FFT	1, 2	
4	**Title**	**DSP Algorithms**	1, 2	**08**
	4.1	Fast Circular Convolution Algorithm, Fast Linear Convolution Algorithm.	1, 2	
	4.2	Linear FIR filtering using Overlap Add Algorithm and Overlap Save Algorithm and implementation using FFT.	1, 2	
5	**Title**	DSP Processors and Applications of DSP	3	**06**
	5.1	Need DSP processor, Difference between DSP processor & General Purpose (GP) Processor.	3	
	5.2	Case study of DSP applications to Speech Signal Processing and Biomedical Signal Processing.	3	
6	**Self Study**	Multi-rate Signal Processing: Up sampling and Down sampling, Signal Compression, Carl Correlation Coefficient for measurement of degree of similarity between two signals.	1, 2, 3, 4, 5	**05***
			Total(* Not included)	**42**

Laboratory Component, if any. (Minimum 10 Laboratory experiments are expected)

Sr. No	Title of the Experiment
1	Signal Operations
2	Discrete Convolution
3	Discrete Correlation
4	Discrete Fourier Transform
5	Magnitude and Phase Spectrum
6	Fast Fourier Transform
7	Overlap Add Method using FFT
8	Overlap Save Method using FFT
9	Application of DSP Part I
10	Application of DSP Part II

Text Books

Sr. No	Title	Edition	Authors	Publisher	Year
1	Digital Signal Processing : Principles, Algorithms and Applications	Fourth Edition	Proakis Manolakis	Pearson Education, ISBN 81-317-1000-9	2007
2	Digital Signal Processing	First Edition	S. Salivahanan, A. Vallavaraj, C. Gnanapriya	TataMcgraw Hill ISBN 978-0-07-066924-6	2010
3	Digital Signal Processing: A Computer Science Perspective	First Edition published on 25th Sept, 2000	Jonathan (Y) Stein	Copyright © 2000 John Wiley & Sons, Inc Print ISBN:9780471295464 |Online ISBN:9780471200598 |DOI:10.1002/047120059X	2000

Reference Books

Sr. No	Title	Edition	Authors	Publisher	Year
4	Digital Signal Processing: A Practical Approach	--	Emmanuel C. Ifeachor, Barrie W. Jervis	Pearson Education ISBN 0-201-59619- 9	2001
5	Digital Signal Processing	Sixth Edition	P. Ramesh Babu	Scitech Publication	2014

<table>
<tr><th rowspan="2">Course (Category) Code</th><th rowspan="2">Course Name</th><th colspan="5">Teaching Scheme (Hrs/week)</th><th colspan="4">Credits Assigned</th></tr>
<tr><th>L</th><th>T</th><th>P</th><th>O</th><th>E</th><th>L</th><th>T</th><th>P</th><th>Total</th></tr>
<tr><td rowspan="3"><b>(PC)</b></td><td rowspan="5"><b>Distributed Computing</b></td><td><b>3</b></td><td><b>0</b></td><td><b>2</b></td><td><b>5</b></td><td><b>10</b></td><td><b>3</b></td><td><b>0</b></td><td><b>1</b></td><td><b>4</b></td></tr>
<tr><td colspan="9"><b>Examination Scheme</b></td></tr>
<tr><td colspan="2"><b>Component</b></td><td colspan="2"><b>ISE</b></td><td colspan="2"><b>MSE</b></td><td colspan="2"><b>ESE</b></td><td><b>Total</b></td></tr>
<tr><td rowspan="2"><b>CS304/IT304</b></td><td colspan="2"><b>Theory</b></td><td colspan="2"><b>75</b></td><td colspan="2"><b>75</b></td><td colspan="2"><b>150</b></td><td><b>300</b></td></tr>
<tr><td colspan="2"><b>Laboratory</b></td><td colspan="2"><b>50</b></td><td colspan="2"><b>--</b></td><td colspan="2"><b>50</b></td><td><b>100</b></td></tr>
</table>

Pre-requisite Course Codes, if any.	Operating Systems, Computer Networks and Communications
Course Objective: To familiarize students with the fundamental concepts, techniques and design of Distributed Systems and use of distributed computing applications domains.	
Course Outcomes (CO): ***At the End of the course students will be able to***	
CS304.1/IT304.1	Understand the principles and desired properties of distributed systems.
CS304.2/ T304.2	Apply the various communication techniques for distributed communication.
CS304.3/IT304.3	Apply the concepts of process, naming, consistency, replication and faults tolerance in distributed environment.
CS304.4/IT304.4	Apply the algorithms such as clock synchronization, election, and mutual exclusion in distributed applications.
CS304.5/IT304.5	Identify the challenges in developing distributed applications.

Theory Component

Module No.	**Unit No.**	**Topics**	**Ref.**	**Hrs.**
1	**Title**	**Introduction to Distributed Systems**		
	1.1	Definition, Type, Goals, Distributed Computing Models, Issues in Distributed Systems.	1,2	**08**
	1.2	Hardware Concepts, Software Concepts, The Client-Server Model, Positioning Middleware, Models of Middleware, Services offered by Middleware, models of Distributed Algorithms and some fundamental problems.	1,2	
2	**Title**	**Communication In Distributed Systems**		**12**
	2.1	Introduction to Message Passing, Desirable Features of a Good Message-Passing System, Issues in IPC by Message Passing, Synchronization, Buffering, Multi-datagram Messages, Group Communication.	1,2	
	2.2	Remote Procedure Call (RPC): Basic RPC Operations, Parameter Passing, Extended RPC Models. Remote Object Invocation: Distributed Objects, Binding a Client to an Object, Static Vs Dynamic RMI Message Oriented Communication: Persistence and synchronicity in communication, Message Oriented Transient and Persistent Communications	1,2	
3	**Title**	**Process in Distributed Systems**		**6**
	3.1	Introduction to Threads, Threads in Distributed Systems, Clients, Server	1,2	
	3.2	Code Migration: Approaches to Code Migration, Models, Migration and Local Resources, Migration in Heterogeneous Systems	1,2	

4	**Title**	**Synchronization in Distributed Systems**		**10**
	4.1	Clock Synchronization: Physical Clocks, Global Positioning System, Clock Synchronization Algorithms; Logical Clocks: Lamport's Logical Clocks, Vector Clocks	1,2	
	4.2	Election Algorithms: Bully and Ring; Mutual Exclusion: Centralized Algorithm, Decentralized Algorithm, Distributed Algorithm, Token Ring Algorithm, Comparison of Algorithms; Load Balancing: Goals, Types, Strategies.	1,2	
5	**Title**	**Consistency and Replication**		**6**
	5.1	Reasons for Replication, Object Replication, Replication as Scaling Technique Data Replication in Distributed Systems, Goals, Types, Schemes,	1	
	5.2	Data-Centric Consistency Models, Client Centric Consistency Models Continuous Consistency, Consistent Ordering of Operations	1	
6	**Self Study**	Naming Entities, Locating Mobile Entities, Distribution Protocols, Consistency Protocols, Faults Tolerance: Process Resilience, Distributed Commit, Recovery	1,2	**8***
			Total(* Not included)	**42**

Laboratory Component

Sr. No	**Title of the Experiments**
1	Implementation of Client Server Communication using RPC/RMI.
2	Implementation of Clock Synchronization (logical/physical).
3	Implementation of Election algorithm.
4	Implementation of Mutual Exclusion algorithm.
5	Implementation of Client Server based program to check data consistency.
6	Implement Load Balancing Algorithms
7	Mini Project

Text Books:

Sr. No	**Title**	**Edition**	**Authors**	**Publisher**	**Year**
1	Distributed Systems– Principles and Paradigms.	First Edition	Andrew S. Tanenbaum, Maarten Van Steen	PHI	2004
2	Distributed Operating Systems Concepts and Design	Second Edition	P. K. Sinha	PHI	2010

Reference Books:

Sr. No	**Title**	**Edition**	**Authors**	**Publisher**	**Year**

1	Distributed Systems – Concept and Design	Fourth Edition	George Coulouris, Jean Dollimore, Tim Kindberg, & Gordon Blair	Pearson	2010
2	Distributed VOD Systems	First Edition	Sudhir D. & Bandu B.M	Research India Publication	2011

Course (Category) Code	Course Name	Teaching Scheme (Hrs/week)					Credits Assigned			
		L	T	P	O	E	L	T	P	Total
SBC	Internet Technology Lab	1	0	2	5	8	1	0	1	2
		Examination Scheme								
		Component		ISE		MSE		ESE		Total
CS305/IT305		Theory		--		--		--		--
		Laboratory		100		--		100		200

Pre-requisite Course Codes, if any.	CS208/IT208 Mini Project
Course Objective: : To impart a knowledge of different Internet Technologies.	
Course Outcomes (CO): ***At the End of the course students will be able to***	
CS305.1/IT305.1	Develop a sophisticated web UX
CS305.2/IT305.2	Create, integrate and test REST based web services
CS305.3/IT305.3	Design secured web application/ web services
CS305.4/IT305.4	Demonstrate behaviour of web crawlers and testing of web application

Theory Component

Module No.	Unit No.	Topics	Ref.	Hrs.
1		**Designing UI**		2
	1.1	Fundamentals of UX Design, Defining UX Solutions, Design Communication and Visualizing Ideas	1	
2		**Web content management system**		1
	2.1	Introduction to Web CMS, different types of Web CMS	2	
3		**Web services**		2
	3.1	Introduction to web service, REST architecture	3	
4		**Web mashups**		1
	4.1	Introduction to web mashups, server side mashups, client side mashups	2	
5		**Secured Web application**		2
	5.1	Introduction to Web Tokens, Auth2.0, OAuth, Access token	2	
6		**Integration of web services**		2
	6.1	Introduction to Mule ESB, Introduction to Anypoint studio, Integrating Web Services using Any point studio	4	
7		**Web crawlers**		2
	7.1	Introduction to web crawler, role of crawler in the internet, concept of page ranking	3	

8		**Testing web applications**		**2**
	8.1	Introduction to different types of testing, manual testing, automated testing, performance testing and functional testing, open source tools used for testing	2	
				14

Laboratory Component, if any. (Minimum 10 Laboratory experiments are expected)

Sr. No	**Title of the Experiment**
1	Design web pages using HTML, CSS and javascript
2	Design UX for a given problem definition by using open source UX tools
3	Create a website using web CMS (Node Js/Angular Js/React Js/Flask/Django/Wordpress/Joomla etc.)
4	Create a Restful webservice to demonstrate different HTTP methods
5	Testing of restful web service using Postman/ARC
6	Create a web mashup of web services using open source framework
7	Design secured Web application using web token
8	Integration of web services using open source integration tools like Mulesoft
9	Demonstrate the behavior of Web Crawlers/ spiders (use XPATH,CSS PATH),extract information and store it in the database.
10	Test the web application using open source testing tools like Selenium, Test runner and Junit

Text Books

Sr. No	**Title**	**Edition**	**Authors**	**Publisher**	**Year**
1	Sketching the User experiences	Second edition	Bill Buxton	Diane Cerra	2010
2	Rich Internet Application AJAX and Beyond	Third edition	Dana Moore, Raymond Budd, Edward Benson	WROX Publisher	2017
3	Web Technology	Second Edition	Srinivasan	Pearson	2014
4	API Recipes with Mulesoft(r) Anypoint Platform	First Edition	WHISHWORKS Editorial Board	White falcon	2017

Reference Books

Sr. No	**Title**	**Edition**	**Authors**	**Publisher**	**Year**
5	Internet Technology And Web Design	First Edition	R. K. JAIN	Khanna Book Publishing Company	2015

6	Understanding the Internet: A Clear Guide to Internet Technologies	First Edition	Keith Sutherland	A ButterworthHeinemann Title	2016
7	RESTful Web APIs: Services for a Changing World	Third edition	Leonard Richardson, Mike Amundsen, Sam Ruby	O'REILLY	2013