Department of Information Technology

Final Year B Tech Syllabus

Autonomous Program Structure of
Final Year B. Tech. (Information Technology)
Academic Year: 2019-2020 Onwards

Final Year B. Tech. (IT) Semester – I										
Course Code	**Course Title**	**Teaching Scheme Hours /Week**			**Examination Scheme**				**Marks**	**Credit**
		Lecture	**Tutorial**	**Practical**	**In Semester**	**End Semester**	**Oral**	**Practical**		
IT 4101	Software Architecture & Design Patterns	3	0	0	50	50	0	0	100	3
IT 4102	Cloud Computing	3	0	0	50	50	0	0	100	3
HS 4101	Green Computing*	3	0	0	50	50	0	0	100	3
OE 4101	Open Elective –I	3	0	0	50	50	0	0	100	3
IT 4103	Software Architecture & Design Patterns Laboratory	0	0	2	0	0	50	0	50	1
IT 4104	Project Phase-I	0	2	14	100	0	50	0	150	9
	Total	**12**	**2**	**16**	**300**	**200**	**100**	**0**	**600**	**22**
	Grand Total	**30**			**600**				**600**	**22**

*Advanced Entrepreneurship Development**

**Prerequisite: Basic Course ED

Final Year B. Tech. (IT) Semester – II										
Course Code	**Course Title**	**Teaching Scheme Hours /Week**			**Examination Scheme**				**Marks**	**Credit**
		Lecture	**Tutorial**	**Practical**	**In Semester**	**End Semester**	**Oral**	**Practical**		
IT 4201	Information and Cyber Security	3	0	0	50	50	0	0	100	3
PEIT 4201	Program Elective-I	3	0	0	50	50	0	0	100	3
OE 4201	Open Elective-II	3	0	0	50	50	0	0	100	3
IT 4202	Information and Cyber Security Laboratory	0	0	2	0	0	50	0	50	1
IT 4203	Project Phase-II	0	2	16	100	0	50	0	150	10
IT 4204	Project based Online Course**	2	0	0	50	0	0	0	50	2
	Total	**11**	**2**	**18**	**300**	**150**	**100**	**0**	**550**	**22**
	Grand Total	**31**			**550**				**550**	**22**

****The student shall register and complete the project based online course preferably in semester- I but may complete the same till the end of semester-II.**

Program Elective I

1. Principles of Compiler Design
2. Information Retrieval
3. Internet of Things
4. Software Defined Networks

List of the Open Elective I and II

Student shall opt / select any one course as open elective in Sem-I and Sem_II

Final Year B. Tech. Semester – I		Final Year B. Tech. Semester – II	
OE 4101 Open Elective-I	**Students from respective deptt may opt for**	**OE 4201 Open Elective-II**	**Students from respective deptt may opt for**
Software Testing and Quality assurance	C/IT/ETC/IN/M	Unified Communication	C/IT/ETC

ETC- Electro & Telecomm, C- Computer Engg
IT- Information Technology, IN- Instru and Control Engg
M- Mechanical Engg

SEMESTER 1

IT 4101 Software Architecture and Design Patterns

Teaching Scheme:
Lectures: 3 hrs/week
Tutorial: NIL

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: 3

Prerequisites: Object Oriented Paradigms, Software Engineering

Course Objectives:

Familiarize students with

1. Concepts of software architecture
2. Different types of software architectural styles
3. Concepts and applications of design patterns.
4. Different types of design patterns

Course Outcomes:

Students should be able to

1. Analyze and suggest architecture design for an application
2. Apply design patterns to software design
3. Evaluate and select appropriate design pattern for a situation
4. Compare the performance of the software on inclusion of various design patterns.

Unit – I: Software Architecture **(07)**

Overview of software Architecture, What drives software architecture, Quality attributes, Architecture design, Architecture documentation

Unit – II: Architectural Patterns **(07)**

Client server multitier architectural pattern, Even driven architectural pattern, Service Oriented Architectures, Component based architecture

Unit – III: Role of design patterns in architecture design **(07)**

Introduction to architecture design, introduction to design patterns, Types of design patterns Abstract factory, builder, factory method, singleton design patterns

Unit – IV: Creational Design Patterns **(07)**

Abstract factory, builder, factory method, singleton design patterns with case study

Unit – V: Structural Design Patterns (07)

Adapter, bridge, composite, facade, decorator, chain of responsibility with case study

Unit – VI: Behavioral Design Patterns (07)

State, Observer, Strategy, template method with case study,

Text Books:

1. Craig Larman, Applying UML and Patterns, Pearson Education, Second Edition, ISBN: 9780130925695.
2. Elizabeth Freeman, Kathy Seirra, Head first design patterns O'Reilly Media ISBN 0596007124

Reference Books:

1. Len Bass, Paul Clements, Rick Kazman Software Architecture in Practice, Pearson Education, ISBN: 978-81-7758-996-2
2. Eric Gamma and other authors Design Patterns Elements of reusable object oriented software Addison Wesley Professional Series ISBN 0-202-63361-2

IT 4102 Cloud Computing

Teaching Scheme:
Lectures: 3 hours/week
Tutorial: NIL

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: 3

Prerequisites: Operating Systems and Computer Networks

Course Objectives:
Familiarize students with
1. Distributed Systems and its ecosystem.
2. Basics of virtualization and its importance.
3. In-depth analysis of cloud computing capabilities.
4. Overview of cloud programming and services.

Course Outcomes:
Students should be able to
1. Recognize need of cloud based solutions.
2. Justify the importance of distributed systems.
3. Determine effective techniques to program cloud systems.
4. Evaluate current challenges and trade-offs in cloud computing.

Unit – I Introduction to Distributed Systems (07)
Scalable Computing over the Internet, Technologies for Network-Based Systems, System Models for Distributed and Cloud Computing, Software Environments for Distributed Systems and Clouds, Performance, Security, and Energy Efficiency

Unit – II Computer Clusters for Scalable Parallel Computing (07)
Clustering for Massive Parallelism, Computer Clusters and MPP Architectures, Design Principles of Computer Clusters, Cluster Job and Resource Management, Case Study: Top Supercomputer Systems

Unit – III Virtual Machines and Virtualization of Clusters and Data Centers (07)
Implementation Levels of Virtualization, Virtualization Structures/Tools and Mechanisms, Virtualization of CPU, Memory, and I/O Devices, Virtual Clusters and Resource Management, Virtualization for Data-Center Automation

Unit – IV Cloud Platform Architecture over Virtualized Data Centers (07)
Cloud Computing and Service Models, Data-Center Design and Interconnection Networks, Architectural Design of Compute and Storage Clouds, Public Cloud Platforms: GAE, AWS, and Azure, Inter-cloud Resource Management, Cloud Security and Trust Management

Unit – V Cloud Programming and Software Environments (07)

Features of Cloud and Grid Platforms, Parallel and Distributed Programming Paradigms, Programming Support of Google App Engine, Programming on Amazon AWS and Microsoft Azure, Emerging Cloud Software Environments,

Unit – VI Grids, P2P, and the Future Internet (07)

Grid Architecture and Service Modeling, Grid Projects and Grid Systems Built, Peer-to-Peer Computing Systems, Cloud Trends in Supporting Ubiquitous Computing, Enabling Technologies for the Internet of Things

Text Books

1. Jack J. Dongarra, Kai Hwang, Geoffrey C. Fox, Distributed and Cloud Computing: From Parallel Processing to the Internet of Things, Elsevier, First Edition

Reference Books

1. Thomas Erl, Zaigham Mahmood and Ricardo Puttini, Cloud Computing: Concepts, Technology & Architecture, Pearson, First Edition
2. Rajkumar Buyya, Christian Vecchiola, S.Thamarai Selvi, Mastering Cloud Computing: Foundations and Applications Programming, McGraw Hill, First Edition
3. A. Srinivasan, J. Suresh, Cloud Computing: A practical approach for learning and implementation, Pearson, First Edition
4. Anthony T. Velte, Cloud Computing: Practical Approach, McGraw Hill, First Edition
5. Ronald L. Krutz and Russell D. Vines, Cloud Security: A Comprehensive guide to Secure Cloud Computing, Wiley, First Edition

HS 4101 Green Computing

Teaching Scheme:
Lectures: 3 hrs/week
Tutorial: NIL

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: 3

Prerequisites: Basic Sciences

Course Objectives:

Familiarize students with

1. Knowledge of green computing practices to minimize negative impacts on the environment.
2. Principles of green computing.
3. Green Computing and how it can help improve environmental sustainability.
4. Green Computing in enterprises and its impact.

Course Outcomes:

Students should be able to

1. Relate to the socio cultural aspects of green computing.
2. Create awareness about green computing and promote green agenda in their working environments leading to green movement.
3. Apply green computing skills such as energy efficiency, IT assets disposal, carbon footprint estimation, reporting and development of green products.
4. Justify green initiatives while developing applications and services in enterprises.

Unit – I: Introduction to Green Computing **(07)**

Environmental Impacts of IT, Need of green computing, Green IT Standards, Enterprise Green IT Strategy, Hardware: Reuse, Recycle and Dispose, present scenario in industry, health issues relevance, Software: Energy-Saving Software Techniques

Unit – II: Software Development and Green Data Centers **(07)**

Sustainable Software, Software Sustainability Attributes, Software Sustainability Metrics, Sustainable Software Methodology, Data Centres and Associated Energy Challenges, Data Centre IT Infrastructure, Data Centre Facility Infrastructure: Implications for Energy Efficiency, IT Infrastructure Management, Green Data Centre Metrics

Unit – III: Green Data Storage and Networks **(07)**

Storage Media Power Characteristics, Energy Management Techniques for Hard Disks, System-Level Energy Management, Objectives of Green Network Protocols, Green Network Protocols and Standards

Unit – IV: Enterprise Green IT Strategy **(07)**

Approaching Green IT Strategies, Business Drivers of Green IT Strategy, Business Dimensions for Green IT Transformation, Multilevel Sustainable Information,

Sustainability Hierarchy Models, Product Level Information, Individual Level Information, Functional Level Information, Organizational Level Information, Regional/City Level Information

Unit – V: Green Computing Services and Roles (07)

Factors Driving the Development of Sustainable IT, Sustainable IT Services (SITS), Sustainable IT Roadmap, Organizational and Enterprise Greening, Information Systems in Greening Enterprises, Greening the Enterprise

Unit – VI: Regulating Green Computing (07)

The Regulatory Environment and IT Manufacturers, Nonregulatory Government Initiatives, Industry Associations and Standards Bodies, Green Building Standards, Green Data Centres, Social Movements

Text Books:

1. San Murugesan, G. R. Gangadharan: Harnessing Green IT, WILEY, 1st Edition-2013.

Reference Books:

1. Woody Leonhard, Katherrine Murray, "Green Home computing for dummies", August 2009, WILEY
2. Bhuvan Unhelkar, "Green IT Strategies and Applications-Using Environmental Intelligence", CRC Press, June 2011
3. Alin Gales, Michael Schaefer, Mike Ebbers, "Green Data Center: steps for the Journey", Shroff/IBM redbook, 2011.
4. Jason Harris, "Green Computing and Green IT-Best Practices on regulations & industry", Lulu.com, 2008
5. Carl Speshocky, "Empowering Green Initiatives with IT", John Wiley & Sons, 2010.
6. Wu Chun Feng (Editor), "Green computing: Large Scale energy efficiency", CRC Press, 2012.

OE 4101 Software Testing and Quality Assurance

Teaching Scheme:
Lectures: 3 hrs/week
Tutorial: NIL

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: 3

Prerequisites: Any programming language

Course Objectives:

Familiarize students with

1. Application of testing strategies in projects.
2. Test management strategies and tools for testing
3. Various quality assurance models

Course Outcomes:

Students should be able to

1. Analyze the project scenario and to select proper testing technique
2. Apply testing techniques to deliver a project
3. Choose quality assurance models for the project
4. Choose one of the latest testing tools suitable for the project

Unit – I: Software testing fundamentals **(07)**

Testing as a Process, Software testing principles, The tester's role in a software development organization, Origins of defects, Defect classes, Testing fundamentals, the defect repository and test design, Defect examples, Developer /Tester support for developing a defect repository. Process model to represent Different phases, Lifecycle models

Unit – II: Levels of testing **(07)**

Need for levels of testing, Unit testing, Integration testing, System Testing - Usability and Accessibility Testing, Configuration Testing, Compatibility Testing, Stress testing, Regression testing, Alpha, Beta and Acceptance testing.

Unit – III: Testing techniques **(07)**

Using White Box Approach to Test design - Static Testing, Structural Testing, Unit Functional Testing, Challenges in White box testing, Using Black Box Approaches to Test Case Design, Random Testing, Requirements based testing, Decision tables, State-based testing, Cause-effect graphing, Error guessing, Compatibility testing.

Unit – IV: Fundamentals of software quality assurance **(07)**

SQA basics, Components of the Software Quality Assurance System, software quality in

business context, planning for software quality assurance, product quality and process quality, software process models, 7 quality control Tools and Modern Tools.

Unit – V: Quality assurance models (07)

Models for Quality Assurance, ISO-9000 series, CMM, CMMI, Test Maturity Models, SPICE, Malcolm Baldrige Model- P-CMM, Clean-room software engineering ,Defect Injection and prevention, Inspections & Walkthroughs, Case Tools and their effect on Software Quality.

Unit – VI: Software test automation and current industry trends (07)

Software Test Automation, Skills needed for Automation, Scope of Automation, Design and Architecture for Automation, Requirements for a Test Tool, Challenges in Automation Tracking the Bug. Combining Manual and Automated Testing, Adoption of DevOps, Big Data Testing, IoT Testing, Introduction to testing tools.

Text Books:

1. Srinivasan Desikan, Gopalaswamy Ramesh,Software Testing: Principles and Practices,Pearson
2. Ilene Burnstein, Practical Software Testing, Springer International edition

Reference Books:

1. Paul C. Jorgensen, Software Testing: A Craftsman's Approach, Auerbach Publications
2. William Perry, Effective Methods of Software Testing, Wiley Publishing, Third Edition
3. Stephen Kan, Metrics and Models in Software Quality, Addison – Wesley, Second Edition
4. Watts S Humphrey, Managing the Software Process, Pearson Education Inc.

IT 4103 Software Architecture and Design Pattern Laboratory

Teaching Scheme:
Practical: 2 hours/week
Tutorial: NIL

Examination Scheme:
Oral: **50** marks
Credits: 1

Prerequisites: Web Engineering Technology, Programming skill development laboratory

Course Objectives:

Familiarize students with

1. One client side programming Technology
2. One server side programming Technology
3. Developing a multiuser application

Course Outcomes:

Students should be able to

1. Apply appropriate technology to design the client side of the application
2. Apply appropriate technology to design the server side part of the application
3. Design the persistent layer classes their connection to database
4. Deploy and run the complete application

List of Assignments

1. Identify a system having three or four user expectations. Prepare its use case model
2. For the same system, prepare its analysis class model.
   Implement it using java language
3. For the same system, refine the analysis model and prepare the design class model.
   Implement it using java. Include appropriate applicable design patterns while designing the system.
4. Add view classes to your model and run the code handling appropriate events.
5. Design persistent layer classes and connect the business logic to database.
6. Deploy the application on server and ensure that it runs for various clients.
   Comment on the Quality attributes addressed in the system.

Text Books

1. Robert Sebasta, Programming the world wide web, Pearson Education, Edition 7, 2013
2. Deitel, Deitel and Nieto, Internet and World wide web how to program
   Pearson Education, Edition 5, 2013

Reference Books

1. Kogent Learning Solutions Inc Web Technologies Black Book 2009

IT 4104-PROJECT PHASE – I

Teaching Scheme:
Tutorial: 2 hrs/week
Practical:14 hrs/week

Examination Scheme:
In semester: **100** marks
Oral: **50** marks
Credits: 9

Course Objectives :
Familiarize students with:

1. The practical implementation of theoretical knowledge gained till date.
2. implementation of their ideas/real time industrial problem/ current application of Computer Science or Information Technology.

Course Outcomes :
At the end of this course the student should be able to :

1. Formulate a statement for the problem in Computer Science or Information Technology domain.
2. Prepare prototype for the identified problem.

Following activities are expected to be completed in Project Phase-I:

1. Identification of Problem
2. Feasibility study
3. Formulation of Problem Statement
4. Abstract writing
5. Literature Survey
6. Project planning and maintaining log
7. High level System Design
8. Preparation of UML diagram using Tools.
9. Study of technology/platform
10. Technical Report writing
11. Start project based online course.

All this should be done with frequent meetings with internal and external guide.
The log has to be maintained.
Every project group has to give 2 Reviews in Semester-I
In Review-I, Point 1 to 4 should be completed. Demonstration and discussion with reviewers will be done.
In Review-II, Point 5 to 11 should be completed. Demonstration and discussion with reviewers will be done.

SEMESTER 2

IT 4201 Information and Cyber Security

Teaching Scheme:
Lectures: 3 hours/week
Tutorial: NIL

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: 3

Prerequisites: Foundations of Computer Networks, Computer Networks

Course Objectives:

Familiarize students with

1. Information Security course surveys central concepts in applied information security and cyber security.
2. Make students aware of the major security risks and attack vectors.
3. Provides tools and practices for building secure systems.
4. Design, develop and support a global security system using the state of mind and reasoning on software systems security.

Course Outcomes:

Students should be able to

1. Apply the knowledge of mathematics, science and computing appropriate to the discipline.
2. Interpret the concept of end-to-end security
3. Analyze the security protocols and develop security systems.

Unit – I Cryptographic Techniques and Algorithms I (07)

Classical Encryption Techniques, Block Ciphers and DES, Basic Concepts in Number Theory and Finite Fields, Advanced Encryption Standard (AES), Block Ciphers. Operations,

Unit – II Cryptographic Techniques and Algorithms II (07)

Pseudo Random Number Generation and Stream Ciphers , Public Key Cryptography, Cryptographic Hash Functions Message Authentication Codes

Unit – III Cryptographic Protocols I (07)

Digital Signatures, Public-Key Certificates PKI, PKIX, and X.509, CA Hierarchy , User Authentication Protocols Public-Key Certificates PKI, PKIX, and X.509, CA Hierarchy

Unit – IV Cryptographic Protocols II (07)

Advanced Protocols: Zero knowledge Proofs, Identity based public key, Secure elections, Secure multi-party computation, and Digital cash.

Unit – V **Network Security** **(07)**

IP Security , Transport Level Security (TLS) HTTPS, HTTPS Use, Secure Shell (SSH), SSH Protocol Stack, Wireless Network Security, Wireless Network Threats, Countermeasures Mobile Device Security

Unit – VI **Cyber security** **(07)**

Electronic Mail Security: Email Security Enhancements, Pretty Good Privacy (PGP), S/MIME Intrusion Detection Malicious Software

Text Books

1. William Stallings, "Cryptography and Network Security: Principles and Practice," 6th Edition, Pearson.

Reference books

1. D. R. Stinson: Cryptography: Theory and Practice (Discrete Mathematics and Its Applications), 3e, CRC Press.
2. B. Schneier: Applied cryptography: protocols, algorithms, and source code in C, 2e, John Wiley & Sons.
3. Bernard Menezes: Network Security & Cryptography, 1st Edition, Cengage Learning, Delhi, 2011.

PEIT 4201 Principles of Compiler Design

Teaching Scheme:
Lectures: 3 hrs / week
Tutorial: NIL

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: 3

Prerequisites: Data Structures, Theory of Computation, Operating System

Course Objectives:
Familiarize students with

1. Process of compilation.
2. Tools used for the development of compilers and other language translation softwares.
3. Basic issues in code generation and optimization.

Course Outcomes:
Students should be able to

1. Design a lexical analyzer for a subset of C language.
2. Design a syntax analyzer for a subset of C language.
3. Generate intermediate code for the given programming language construct.
4. Apply different code optimization & generation techniques for a given code.

Unit – I: Introduction to Compiler & Lexical Analysis (07)

Introduction to compilers Design issues, passes, phases, symbol table Preliminaries, Memory management, Lexical Analysis Tokens, Regular Expressions, Process of Lexical analysis, Block Schematic, Automatic construction of lexical analyzer using LEX, LEX features and specification

Unit – II: Syntax Analysis (07)

Syntax Analysis Grammar (ambiguous, unambiguous, CFG), top-down parser (RDP, Predictive) and bottom-up parsers (SLR, LR-1, LALR), Error detection and recovery, automatic construction of parsers using YACC

Unit – III: Semantic Analysis (07)

Introduction to Semantic analysis, Need of semantic analysis, type checking, Syntax directed translation scheme, Intermediate code - need, types: Syntax Trees, DAG, Three Address codes: Quadruples, Triples and Indirect Triples, Intermediate code generation of declaration statement and assignment statement.

Unit – IV: Runtime Storage Management **(07)**

What is run-time support? Parameter passing methods, Storage allocation, Activation records, Static scope and dynamic scope, Heap memory management, Garbage Collection

Unit – V: Code optimization **(07)**

Machine Independent: Peephole optimizations: Common Sub-expression elimination, Removing of loop invariants, Induction variables and Reduction in strengths, use of machine idioms

Unit – VI: Code Generation **(07)**

Basic block, Register allocation and Assignment, Simple code generator, Sethi Ulman algorithm for code generation

Text Books:

1. Alfred V. Aho, Ravi Sethi, Jeffrey D. Ullman, "Compilers Principles, Techniques and Tools Addison Wesley, ISBN:981–235–885-4.
2. Kenneth C. Louden , "Compiler Construction: Principles and Practice", Course Technology In ISBN-10: 0534939724, ISBN-13: 978-0534939724.

Reference Books:

1. Dick Grune, Bal, Jacobs, Langendoen, "Modern Compiler Design", Wiley, ISBN 81-265-041 8.
2. J R Levin, T Mason, D Brown, "Lex and Yacc", O'Reilly, 2000 ISBN 81-7366-061-X.
3. K Muneeswaran, "Compiler Design", Oxford University press, ISBN 0-19-806664-3.
4. Allan Holub, "Compiler design in C", Prentice Hall, ISBN-13: 978-0131550452, ISBN-1 0131550454.

PEIT 4201 Information Retrieval

Teaching Scheme:
Lectures: 3 hrs/week
Tutorial: NIL

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: 3

Prerequisites: Data structures

Course Objectives:

Familiarize students with

1. Concepts if Information Retrieval System.
2. Indexing techniques of Information retrieval System
3. Clustering in information retrieval system
4. Understand information sharing on semantic web

Course Outcomes:

Students should be able to

1. Model the working of information retrieval search system
2. Analyze Search Strategies used in Information retrieval system
3. Evaluate Information retrieval system using different statistical measures
4. Design techniques for information retrieval system

Unit – I: Introduction **(07)**

Basic Concepts of Information Retrieval, IR system architecture. Automatic Text Analysis: Luhn's ideas, Conflation Algorithm, Porter Stemmer, Retrieval Evaluation: Precision, Recall, F-Score, Mean Average Precision, Mean Reciprocal Rank, User oriented measures

Unit – II: Indexing and Clustering **(07)**

Indexing and Index Term Weighing, Probabilistic Indexing, Inverted file, Suffix trees & suffix arrays, Signature Files, Clustered files, Cluster Hypothesis, Clustering Algorithms: Single Pass Algorithm, Single Link Algorithm

Unit – III: Search Strategies **(07)**

Retrieval strategies: Vector Space model, Probabilistic retrieval strategies, Language models, Inference networks, Extended Boolean retrieval, Latent semantic indexing, Fuzzy set retrieval

Unit – IV: Web Mining **(07)**

Searching the Web: Challenges, Characterizing the Web, Search Engines, Browsing, Mata-searchers, Web crawlers, Meta-crawler, Web data mining, Finding needle in the Haystack, Searching using Hyperlinks

Unit – V: Semantic Search Systems **(07)**

Semantic Search systems, Semantic Web, Ontology, Searching across ontologies, semantic web search, Google knowledge graphs

Unit – VI: Trends In Information Retrieval **(07)**

Case Study: Google Analytics, Search Engine Optimization, Ranking Algorithms, Recommendation Systems: Collaborative Filtering.

Text Books:

1. Yates & Neto, Modern Information Retrieval, Pearson Education, ISBN:81-297-0274-6
2. C.J. Rijsbergen, Information Retrieval, (www.dcs.gla.ac.uk).,2ndISBN:978-408709293

Reference Books:

1. Grigoris Antoniou and Frank van Harmelen, A semantic Web Primer, Massachusetts.
2. Christopher D. Manning, Prabhakar Raghavan and Hinrich Schutzen, Introduction to Information Retrieval, Cambridge University Press, Online book, ISBN:978-0-521-86571-5.

PEIT 4201 Internet of Things

Teaching Scheme:
Lectures: 3 hours/week
Tutorial: NIL

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: 3

Prerequisites: Computer Networks

Course Objectives:

Familiarize students with

1. Logical and Physical design of IOT
2. IOT architecture and its structural aspects
3. Various IOT Protocols
4. IOT solutions and applications

Course Outcomes:

Students will be able to:

1. Interpret logical and physical design of IOT enabling technologies
2. Link IOT architecture with its different structural aspects
3. Differentiate various IOT protocols
4. Propose IOT solutions for various applications

Unit – I Introduction (07)

Definition and characteristics of IOT, Physical design of IOT: Things in IOT, IOT Protocols, Logical Design of IOT: IOT functional blocks, Logical Design of IoT: Functional block, communication Model, Communication API's, IoT Enabling Technologies

Unit – II IOT Network Architecture (07)

IOT Architecture, IoT levels and Deployment templates: Level 1 to Level 5.Introduction to M2M, Difference between IoT and M2M, IoT protocol stack, Fog Computing, Edge Computing

Unit – III IOT Physical Devices and Objects (07)

Basic building blocks of IOT Device, Sensors, Actuators, and Smart Objects, Exemplary Devices: Raspberry Pi, Raspberry Pi Interfaces, pcDuino, Beagle Bone Black, CubieBoard, ARDUINO, SCADA

Unit – IV IOT Networking and Addressing techniques (07)

RFID technology, Wireless Sensor Networks, IPv6 Protocol Overview, comparison of IPv4 and IPv6, IPV6 tunneling, IPsec in IPv6, Quality of Service in IPv6

Unit – V IOT Protocols and Cloud offerings (07)

IoT Access Technologies: IEEE 802.15.4, IEEE 802.15.4g and 802.15.4e, IEEE 1901.2a, LoRaWAN, MQTT protocol

Introduction to cloud storage models and communication API's, web services for IoT

Unit – VI IOT Applications (07)

Smart City, Agriculture, healthcare, Retails, Environment

Text Books

1. “Internet of Things: A Hands-On Approach”, Arshdeep Bahga, Vijay Madisetti, University Press, 2015, ISBN: 978- 8173719547.
2. “Building the Internet of Things with IPv6 and MIPv6: The Evolving World of M2M Communications”, Daniel Minoli, Wiley Publications, 2013, ISBN: 978-1-118- 47347-4

Reference Books

1. “The Internet of Things: Connecting Objects to the Web”, Hakima Chaouchi, Wiley Publications, ISBN: 978-1- 84821- 140-7.
2. “IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things”, David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Cisco Press, 16 Aug 2017, ISBN: 978-1- 58714-456- 1 599.

PE 4201 Software Defined Networks

Teaching Scheme:
Lectures: 3 hours/week
Tutorial: NIL

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: 3

Prerequisites: Foundations of Computer Networks, Computer Networks

Course Objectives:

1. Appraise SDN
2. To comprehend role of data, control, and management planes and their separation
3. Differentiate between network virtualization and network function virtualization
4. Analyze Openflow protocol.

Course Outcomes:

Students should be able to

1. To comprehend role of data, control, and management planes and their separation
2. Relate network virtualization and network function virtualization
3. Interpret Openflow protocol.

Unit – I SDN architecture and Fundamentals. (07)

Introduction: The Modern Data Center, Roles and Separation of data, control and management Planes, Advantages and Disadvantages. Need of SDN, Genesis of SDN.
Working of SDN: Fundamental characteristics, SDN Devices, SDN controllers, Applications

Unit – II Openflow and Abstraction (07)

Introduction: Definition, OpenFlow architecture, Flow & Group Tables, types, Hybrid Approaches, The OpenFlow forwarding and pipeline model. OpenFlow Advantages and Limitations, OpenFlow Protocol.

Unit – III Network Virtualization (07)

Definition, Concepts, Benefits of Network Virtualization, Components of a Virtual Network, Applications, Existing Network Virtualization Framework

Unit – IV Control Plane (07)

Control Plane: Overview, Existing SDN Controllers including Floodlight and Open Daylight projects. Customization of Control Plane: Switching and Firewall Implementation using SDN Concepts.

Unit – V Data Plane (07)

Data Plane: Software-based and Hardware-based; Programmable Network, Hardware. Programming SDNs: Northbound Application Programming Interface, Current Languages and Tools, Composition of SDNs.

Unit – VI Network Function Virtualization (07)

Introduction: Concepts, Comparison of NFV and NV, Implementation and Application, Data Center Networks, Application of NFV in LTE, IMS, Content Delivery, Mobile Networks

Text Books

1. Thomas D. Nadeau, Ken Gray, SDN: Software Defined Networks, An Authoritative Review of Network Programmability Technologies, O'Reilly Media, ISBN:10:1-4493-4230-2, 978-1-4493-4230-2

2. Paul Goransson and Chuck Black, Software Defined Networks: A Comprehensive Approach, Morgan Kaufmann, ISBN:9780124166752, 9780124166844

Reference books

1. Vivek Tiwari, SDN and OpenFlow for Beginners, Digital Services, 10: 1-940686-00-8 13: 978-1-940686-00-4
2. Fei Hu, Network Innovation through OpenFlow and SDN: Principles and Design,CRC
   Press,ISBN:10: 1466572094

3. Open Networking Foundation (ONF)Documents, https://www.opennetworking.org

4. Online Reading, http://www.nec-labs.com/~lume/sdn-reading-list.html,

OE 4201 Unified Communication

Teaching Scheme:
Lectures: 3 hours/week

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: 3

Prerequisites: Foundations of Computer Networks, Computer Networks

Course Objectives:
Familiarize students with

1. Compare Circuit switching and packet switching related to performance parameters.
2. Choose VOIP protocols for unified communications.
3. Analyze contact center as application of unified communications.
4. Interpret emerging technologies/protocols in VOIP communications.

Course Outcomes:
Students should be able to

1. Understand and apply VOIP unified communications and analytics concepts to Contact Center Working.
2. Design and Implement VOIP protocols for telecommunication systems/applications.
3. Interpret and apply current or emerging knowledge in telecommunication engineering.
4. Use relevant mathematics and computer science concepts as tools.

Unit – I Introduction to digital and lP Telephony (07)

Digital Telephony: circuit switched networks, ss7, ISDN, Exchanges, E.164 Numbering Plans IP Telephony: Packet switched Networks, signaling & Media separation' Media Encapsulation ' RTP and RTCP, Audio and Video Codecs.

Unit – II VolP Protocols (07)

H.323 Network Elements, H.323 protocol, H.323 Call flows, SIP Network Elements, SIP Protocols, SIP Call Flows, H.248 protocol : Media Gateways, Media Gateway controllers, commands, Transactions, Contexts, Terminations, Descriptors' Packages

Unit – III Unified Communications (07)

Local and Network features, Voice & Data Integration, Collaboration, Mobility, Business Applications: Framework for custom applications, computer Telephony Interface, Application Sequencing.

Unit – IV **Inbound Contact Center** **(07)**

Call Centers: Introduction, Evolution and classification of Contact Centers.
Inbound Contact Center :Introduction Self Service / Interactive Voice Response, Routing, Intelligent Routing, VXML
Agent : Skills, Selection Algorithms, Modes, Service Observing, Recording

Unit – V **Outbound Contact Center and Reporting** **(07)**

Outbound contact center: Introduction, Proactive contact: voice, SMS, E-mail & chat. Contact Center Reporting: Types of Reports, Business use cases.
Analytics: Agent Performance, Occupancy

Unit – VI **Emerging technologies in Telecommunications** **(07)**

High Availability: Load balancing, Reliability, Failover & Failback, Location Redundancy, Hardware footprint, cloud Computing, Emerging Technologies: Google Glass, WebRTC, Hosting on Cloud.

Text Books

1. Allan Sulkin, "PBX Systems for IP Telephony" McGraw-Hil! Professional

Reference books

1. ITU-T H.323 Packet-based multimedia communications systems
2. ITU-TH.225Call Signaling Protocols and media stream packetization
3. ITU-T H-245 Control protocol for multimedia communication
4. IETF RFC 326131P: Session Initiation Protocol
5. IETF RFC4566 SDP: Session Description Protocol
6. Contact Center for' Dummies, Wiley Publishing Inc.
7. Real Time Communication with WebRTC, O'Reilly Publishing

IT 4202 Information and Cyber Security Laboratory

Teaching Scheme:
Practical: 2 hours/week

Examination Scheme:
Oral exam: **50** marks
Credits: 1

Prerequisites: Foundations of Computer Networks, Computer Networks

Course Objectives:

1. Learn to implement the algorithms DES, RSA, MD5, SHA-1 etc.
2. Make students aware of the major security risks and attack vectors.
3. Provides tools and practices for building secure systems.
4. Learn to use network security like GnuPG, KF sensor, Net Strumbler

Course Outcomes:
Students should be able to

1. Implement the cipher techniques
2. Analyze the various security algorithms and protocols
3. Use different open source tools for network security and analysis

List of experiments:

1.Implement the following SUBSTITUTION & TRANSPOSITION TECHNIQUES concepts (any 2) :
a) Caesar Cipher
b) Playfair Cipher
c) Hill Cipher
d) Vigenere Cipher
e) Rail fence – row & Column Transformation

2. Implement the following algorithms (any 3)
a) DES
b) RSA Algorithm
c) Diffie-Hellman
d) MD5
e) SHA-1

3.Implement the Signature Scheme

4. Demonstrate intrusion detection system (ids) using any tool (snort or any other s/w)

5. Analysis the Security Vulnerabilities of E-commerce services. / Analysis the security vulnerabilities of E-Mail Application

6. Steps to ensure Security of any one web browser (Mozilla Firefox/Google Chrome)

7. Study assignment: (any 1)

A. Study of different wireless network components and features of any one of the Mobile Security Apps.
B. Study of the features of firewall in providing network security and to set Firewall Security in windows.
C. Study of different types of vulnerabilities for hacking a websites / Web Applications.

Text Books

1. William Stallings, "Cryptography and Network Security: Principles and Practice," 6th Edition, Pearson.

Reference books

1. B. Schneier: Applied cryptography: protocols, algorithms, and source code in C, 2e, John Wiley & Sons.
2. Bernard Menezes: Network Security & Cryptography, 1st Edition, Cengage Learning, Delhi, 2011.

IT 4203-PROJECT PHASE – II

Teaching Scheme:
Tutorial: 2 hrs/week
Practical: 16 hrs/week

Examination Scheme:
In-Semester: **100** Marks
Oral: **50** marks
Credits: 10

Prerequisites: BTech-Project Phase I – Semester I

Course Objectives:
Familiarize students with:

1. Product development cycle.
2. Paper presentation activities.
3. Technical writing.

Course Outcomes:
At the end of this course the student should be able to:

1. Propose a System model.
2. Apply technical knowledge for solving a problem.
3. Create solution for problem in Computer Science and Information Technology.
4. Test product/service.

Following activities are expected to be completed in Project Phase-II:

1. Completion of online course.
2. Additional literature survey.
3. Detailed System design.
4. Implementation of project.
5. Write test cases.
6. Test developed project using testing tools.
7. Writing journal/conference paper on the project.
8. Participations in project competitions.
9. Project report preparation.

All this should be done with frequent meetings with internal and external guide.
The log has to be maintained.
Every project group has to give 2 Reviews in Semester-II
In Review-III, Point 1 to some part of 4 should be completed. Demonstration and discussion with reviewers will be done.
In Review-IV, remaining part from Point 4 to 9 should be completed. Demonstration and discussion with reviewers will be done.
At least one paper should be published in reputed International conference/International journal.

IT 4204 PROJECT BASED ONLINE COURSE

Teaching Scheme:
Lecture: 2 hrs/week

Examination Scheme:
In-Semester: **50** marks

Credits: 2

Course Objectives:

Familiarize students with
1. Exploring technical literature with the purpose of formulating a project statement.
2. Formulate intended future work based on the course they have registered.
3. Developing a prototype for the project statement.

Course Outcomes:

Students should be able to
1. Perform focused study of technical literature relevant to a specific topic.
2. Build independent thinking abilities to approach complex problems.
3. Develop prototype to test and validate project statement

Contents

1. The Project guide will suggest one/two online courses (which students have not studied till date)
2. Multiple courses can be taken by different group member of the same group.
3. Assignments related to project should be completed.