



PUNE VIDYARTHI GRIHA'S
COLLEGE OF ENGINEERING AND TECHNOLOGY, PUNE-9
(AFFILIATED TO SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE)

DEPARTMENT OF INFORMATION TECHNOLOGY

CURRICULUM BOOK

ACADEMIC YEAR: 2019-20

FOR THE PROGRAMME

T.E. INFORMATION TECHNOLOGY



**PUNE VIDYARTHI GRIHA'S
COLLEGE OF ENGINEERING AND TECHNOLOGY**

VISION

TO ACHIEVE EXCELLENCE IN ENGINEERING EDUCATION

MISSION

- **To satisfy all stakeholders**
- **To develop ethical, highly motivated engineering professionals with good human values, requisite skills and competencies**
- **To adopt innovative teaching mechanisms**
- **To promote research culture**
- **To contribute to country's economic development**
- **To be responsive to changes in technology, socio-economic and environmental conditions**

Curriculum Book

DEPARTMENT OF INFORMATION TECHNOLOGY

VISION

To Empower Students to Face the Technological Challenges of 21st Century by Imparting Quality Education in the Field of Information Technology

MISSION

- 1) To impart knowledge through innovative teaching-learning process to cater the needs of industries and higher education.**
- 2) To inculcate good human values, professional competencies and create awareness about global technologies in the field of Computer Engineering.**
- 3) To respond to rapid changes in the field of Information Technology.**

PROGRAM EDUCATIONAL OBJECTIVES

PEO1: Possess strong fundamental concepts in Engineering Science and Technology to address future technological challenges of Information Technology.

PEO2: Possess knowledge and skills in the field of Information Technology for engineering problems with innovative approaches.

PEO3: Possess behavioral aspects for research, entrepreneurship and higher studies in the field of Computer Science and Information Technology.

PEO4: Have commitment to ethical practices in the field of Information Technology and, societal contributions through communities and life-long learning.

PEO5: Possess better interpersonal and presentation skills to cope up with the rapid changes in the field of Information Technology at global level.

PROGRAMME OUTCOMES

The Program Outcomes of the Department of Information Technology are:

PO1)An ability to apply knowledge of computing, engineering mathematics, statistics, science, and engineering and technology.

PO2)An ability to identify and analyze the problem, provide a systematic solution by conducting experiments, interpreting the data and drawing substantial conclusion.

PO3)An ability to design, implement, and evaluate a software or a software/hardware system, component, or process to meet desired needs within realistic constraints.

PO4)An ability to identify, formulate, and provide systematic solutions to complex engineering problems and validate the solution.

PO5)An ability to apply appropriate resources, skills, modern engineering tools and technologies necessary for practice as a IT professional.

PO6)An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems with necessary constraints and assumptions.

PO7)An ability to analyze the local and global impact of computing on individuals, organizations and society.

PO8)An ability to understand professional, ethical, legal, security and social issues and responsibilities.

PO9)An ability to function effectively as an individual or as a team member to accomplish a desired goal(s) in multidisciplinary environment.

PO10)An ability to engage in life-long learning and continuing professional development to cope up with fast changes in the technologies/tools with the help of open electives, professional organizations and extra-curricular activities.

PO11)An ability to communicate effectively in engineering community at large by means of effective presentations, report writing, paper publications, demonstrations.

PO12)An ability to understand engineering, management, financial aspects, performance, optimizations and time complexity necessary for professional practice.

PO13)An ability to apply design and development principles in the construction of software systems of varying complexity.

PROGRAMME SPECIFIC OUTCOMES

At the Completion Graduates will be competently -

PSO1. Analyze and develop effective and efficient software solution in the field of data base management system ,web technology ,networking etc. by applying the core concepts of Information Technology.

PSO2. Work in teams in various roles to manage IT projects with the help of project management techniques.

Curriculum Book

INDEX

Sr. No.	Course Number	Titles	Page No.
1	314441	Theory Of Computation	4
2	314442	Database Management Systems	7
3	314443	Software Engineering & Project Management	12
4	314444	Operating System	15
5	314445	Human-Computer Interaction	18
6	314450	Computer Network Technology	22
7	314451	Systems Programming	26
8	314452	Design and Analysis of Algorithms	29
9	314453	Cloud Computing	32
10	314454	Data Science & Big Data Analytics	36

Third Year

Curriculum Book

Syllabus Structure of Savitribai Phule Pune University, Pune

Semester I

Subject Code	Subject	Teaching Scheme			Examination Scheme					Total Marks	Credits
		Lecture	Tutorial	Practical	In-Sem. Paper	End-Sem. Paper	TW	PR	OR		
314441	Theory of Computation	4	--	--	30	70	--	--	--	100	4
314442	Database Management Systems	4	--	--	30	70				100	4
314443	Software Engineering &Project Management	3	--	--	30	70	--	--	--	100	3
314444	Operating System	4	--	--	30	70	--	--	--	100	4
314445	Human-Computer Interaction	3	--	--	30	70	--	--	--	100	3
314446	Software Laboratory-I		--	4	--	--	25	50	50	125	2
314447	Software Laboratory-II	--	--	4	--	--	25	50	--	75	2
314448	Software Laboratory-III	--	--	2	--	--	50	--	--	50	1
314449	Audit Course 3	--	--		--	--	--	--	--	Grade	
	Total	18	--	10	150	350	100	100	50	750	23
	Total of Part-I	28 Hours				750					

Semester II

Subject Code	Subject	Teaching Scheme			Examination Scheme					Total Marks	Credits
		Lecture	Tutorial	Practical	In-Sem. Paper	End-Sem. Paper	TW	PR	OR		
314450	Computer Network	3	-	--	30	70	--	--	--	100	3
314451	Systems Programming	4	-	--	30	70	--	--	--	100	4
314452	Design and Analysis of Algorithms	4	-	-	30	70	--	--	--	100	4
314453	Cloud Computing	3	-	-	30	70	--	--	--	100	3
314454	Data Science & Big Data Analytics	4	-	-	30	70	--	--	--	100	4
314455	Software Laboratory-IV	--	--	2	--	--	25	--	25	50	1
314456	Software Laboratory-V	--	--	4	--	--	50	50	--	100	2
314457	Software Laboratory-VI	--	--	2	--	--	25	25	--	50	1
314458	Project Based Seminar	--	01	--	--	--	--	--	50	50	1
314459	AuditCourse4	--	--	--	--	--	--	--	--	Grade	
	Total	18	01	08	150	350	100	75	75	750	23
	Total of Part-II	27 Hours			750						

TE IT(2015 Course)

Sem I

THEORY OF COMPUTATION

Course Title:	THEORY OF COMPUTATION	Course Number: 314441	Credits: 4
Designation of Course	Professional Core		
Teaching Scheme: 4 Hrs/Week		Laboratories: --	
Course Assessment Methods	Direct methods	On-line/In-sem Examination: 10/30 Marks	Theory/End Semester Examination: 28/70 Marks
		Term-work	Practical/Oral
	Indirect Methods	Assignments, Presentations	Seminars, Quiz, Q&A session, Group Discussion
Prerequisites	1. Discrete Structures. 2. Data structures and problem solving..		
Introduction of Course			
Course Objectives			
1	To understand problem classification and problem solving by machines.		
2	To understand the basics of automata theory and its operations.		
3	To study computing machines by describing, classifying and comparing different types of computational models.		
4	Encourage students to study theory of computability and complexity.		
5	To understand the P and NP class problems and its classification.		
6	To understand the fundamentals of problem decidability and reducibility.		
Course Outcomes			
CO1	To construct finite state machines to solve problems in computing.		
CO2	To write mathematical expressions for the formal languages		
CO3	To apply well defined rules for syntax verification.		
CO4	To construct and analyze Push Down, Post and Turing Machine for formal languages.		
CO5	To express the understanding of the decidability and decidability problems.		
CO6	To express the understanding of computational complexity.		
Course Contents			
Unit-I	FINITE STATE MACHINES Basic Concepts: Symbols, Strings, Language, Formal Language, Natural Language. Basic Machine and Finite State Machine. FSM without output: Definition and Construction-DFA, NFA, NFA with epsilon-Moves, Minimization Of FA, Equivalence of NFA and DFA, Conversion of NFA with epsilon moves to NFA, Conversion of NFA With epsilon Move to DFA. FSM with output: Definition and Construction of Moore and Mealy Machines, Inter-conversion between Moore and Mealy Machines.		
Unit-II	REGULAR EXPRESSIONS		

	Definition and Identities of Regular Expressions, Construction of Regular Expression of the given L, Construction of Language from the RE, Construction of FA from the given RE using direct method, Conversion of FA to RE using Arden's Theorem, Pumping Lemma for RL, Closure properties of RLs, Applications of Regular Expressions.		
Unit-III	CONTEXT FREE GRAMMAR AND LANGUAGES		
	Introduction, Formal Definition of Grammar, Notations, Derivation Process: Leftmost Derivation, Rightmost Derivation, derivation trees, Context Free Languages, Ambiguous CFG, Removal of ambiguity, Simplification of CFG, Normal Forms, Chomsky Hierarchy, Regular grammar, equivalence of RG(LRG and RLG) and FA.		
Unit-IV	PUSHDOWN AUTOMATA AND POST MACHINES		
	Push Down Automata: Introduction and Definition of PDA, Construction (Pictorial/ Transition diagram) of PDA, Instantaneous Description and ACCEPTANCE of CFL by empty stack and final state, Deterministic PDA Vs Nondeterministic PDA, Closure properties of CFLs, pumping lemma for CFL. Post Machine- Definition and construction.		
Unit- V	TURING MACHINES		
	Formal definition of a Turing machine, Recursive Languages and Recursively Enumerable Languages, Design of Turing machines, Variants of Turing Machines: Multi-tape Turing machines, Universal Turing Machine, Nondeterministic Turing machines. Comparisons of all automata.		
Unit-VI	COMPUTATIONAL COMPLEXITY		
	Decidability: Decidable problems concerning regular languages, Decidable problems concerning context-free languages, Un-decidability, Halting Problem of TM, A Turing-unrecognizable language. Reducibility: Un-decidable Problems from Language Theory, A Simple Un-decidable Problem PCP, Mapping Reducibility. Time Complexity: Measuring Complexity, The Class P, Examples of problems in P, The Class NP, Examples of problems in NP, NP-completeness.		
Text Books	Author	Title of Book	Publication
T1	Michael Sipser,	"Introduction to the Theory of Computation"	CENGAGE Learning.
T2	Vivek Kulkarni	"Theory of Computation"	Oxford University Press
T3			
T4			
Reference Books			
R1	HopcroftUlman	"Introduction to Automata Theory, Languages and Computations"	Pearson Education
R2	Daniel. A. Cohen	"Introduction to Computer Theory",	Wiley-India
R3	K.L.P Mishra, N. Chandrasekaran	"Theory of Computer Science (Automata, Languages and Computation)"	Prentice Hall India
R4	Kavi Mahesh	Theory of Computation	Wiley-India
R5	Basavaraj S. Anami, Karibasappa K.G	Formal Languages and Automata Theory,	Wiley-India

Self-Learning Facilities	NPTEL Lecture Series by Dr. Kamala Srivastva		
Web Resources			
Research papers for reference	Author	Title of Paper	Journal/Transaction
1			
2			
3			
Contents beyond Syllabus			
Additional Experiments			
Bridging Courses			
Assignments			
1			
2			
3			
4			
5			
Tutorials			
Presentations			

DATABASE MANAGEMENT SYSTEMS

Course Title:	Database Management Systems	Course Number: 314442	Credits:04
Designation of Course	Professional Core		
Teaching Scheme: 4 Hours/Week		Laboratories: 4 Hrs/Week	
Course Assessment Methods	Direct methods	In-sem Examination: 30 Marks	Theory/End Semester Examination:70 Marks
		Term-work	Practical/Oral
	Indirect Methods	Assignments, Class Test	Quiz, Q&A session,
Prerequisites	Data Structure		
Introduction of Course: This subject contains fundamental concepts of database management. These concepts include aspects of database design, database languages, and database-system implementation.			
Course Objectives			
1	To understand the fundamental concepts of database management. These concepts include aspects of database design, database languages, and database-system implementation.		
2	To provide a strong formal foundation in database concepts, technology and practice.		
3	To give systematic database design approaches covering conceptual design, logical design and an overview of physical design.		
4	Be familiar with the basic issues of transaction processing and concurrency control.		
5	To learn and understand various Database Architectures and Applications.		
6	Understand how analytics and big data affect various functions now and in the future.		
Course Outcomes			
CO1	Students shall be able to understand what are basic functions of DBMS & RDBMS		
CO2	Students shall be able to analyze database models & entity relationship models.		
CO3	Students shall be able to design and implement a database schema for a given problem-domain		
CO4	Populate and query a database using SQL DML/DDL commands.		
CO5	Programming PL/SQL including stored procedures, stored functions, cursors and packages		
CO6	Appreciate the impact of analytics and big data on the information industry and the external ecosystem for analytical and data services		
Course Contents			
Unit-I	INTRODUCTION TO DBMS		
	Introduction: Database Concepts, Database System Architecture, Data Modeling: Data Models, Basic Concepts, entity, attributes, relationships, constraints, keys. E-R and EER diagrams: Components of E-R Model, conventions, converting E-R diagram into tables, EER Model components, converting EER diagram into tables, legacy system model. Relational Model: Basic concepts, Attributes and Domains, Codd's Rules. Relational Integrity: Domain, Entity, Referential Integritys, Enterprise Constraints, Schema Diagram. Relational Algebra: Basic Operations, Selection, projection, joining, outer join, union, difference, intersection, Cartesian product, division operations (examples of queries in relational algebraic using symbols).		
	Practical		
	1. Design University database with at least 3 entities and relationships between them.		

	Apply DCL and DDL commands. Draw suitable ER/EER diagram for the system.
Unit-II	DATABASE DESIGN AND SQL
	<p>Database Design: Functional Dependency, Purpose of Normalization, Data Redundancy and Update Anomalies, Single Valued Normalization: 1NF, 2NF, 3NF, BCNF. Decomposition: lossless join decomposition and dependency preservation, Multi valued Normalization (4NF), Join Dependencies and the Fifth Normal Form.</p> <p>Introduction to SQL: Characteristics and advantages, SQL Data Types and Literals, DDL, DML, DCL, SQL Operators, Tables: Creating, Modifying, Deleting, Views: Creating, Dropping, Updating using Views, Indexes, Nulls</p> <p>SQL DML Queries: SELECT Query and clauses, Set Operations, Predicates Functions and Joins, Set membership, Tuple Variables, Set comparison, Ordering of Tuples, Aggregate, Nested Queries, Database Modification using SQL Insert, Update and Delete Queries.</p> <p>Practical</p> <ol style="list-style-type: none"> 1. Design University database with at least 3 entities and relationships between them. Draw suitable ER/EER diagram for the system. And normalize that database upto 3NF. 2. Design at least 10 SQL/NoSQL queries for suitable database application using SQL/NoSQL DML Statements: Insert, Select, Update, Delete with operators, functions, and set operator. <ul style="list-style-type: none"> A) SELECT Statement with all kind of Clauses B) DELETE, UPDATE, INSERT Command C) C) Use of Boolean and Arithmetic operators 3. Design at least 10 SQL/NoSQL queries for suitable database application using SQL DML statements: <ul style="list-style-type: none"> A) Aggregate Function & other built-in Functions B) Use group by and having clauses. C) SET OPERATION 4. Design at least 10 SQL/NoSQL queries for suitable database application using SQL DDL /DML statements: <ul style="list-style-type: none"> A) Perform Set membership (in, not in) B) Perform Set comparison (<some, >=some, <all etc.) C) C) Perform Set cardinality (unique, not unique). 5. Execute DDL statements which demonstrate the use of views. Try to update the base table using its corresponding view. Also consider restrictions on updatable views and perform view creation from multiple tables.
Unit-III	QUERY PROCESSING AND DATABASE TRANSACTIONS
	<p>Query Processing: Overview, Measures of query cost, Evaluation of expression, Materialization and Pipelining algorithm.</p> <p>Transaction: Basic concept of a Transaction, Transaction Management, Properties of Transactions, Concept of Schedule, Serial Schedule, Serializability: Conflict and View, Cascaded Aborts, Recoverable and No recoverable Schedules.</p> <p>PL/SQL: Concept of Stored Procedures, Cursors, Triggers, assertions, roles and privileges</p> <p>Programmatic SQL: Embedded SQL, Dynamic SQL, and Advanced SQL-Programming in MYSQL.</p> <p>Practical</p> <ol style="list-style-type: none"> 1. Write a PL/SQL block or Stored Procedure to calculate the Grade of minimum 10 students. 2. Consider following Student(Exam_No,NAME,SUB1,SUB2,SUB3) table in which marks for each student are recorded Write a cursor in which percentages for each student are calculated. If it is greater

	<p>than or equal to 60, make that entry into stud_First (Exam_no, Name, Sub1, Sub2, Sub3, Percent) table.</p> <p>If less than 60, make that entry into stud_Pass (Exam_no, Name, Sub1, Sub2, Sub3, and Percent) table.</p> <p>If student is fail in any subject(less than 40 in any subject), make that entry into stud_fail (Exam_no, Name, sub1, sub2, sub3) table.</p>
	<p>3. Write a PL/SQL function which accepts basic salary of an employee and returns a Gross salary of an employee.</p> <p>Gross salary = Basic + HRA + DA + TA</p> <p>a. Where HRA = 50% OF BASIC</p> <p>i. DA = 100% OF BASIC</p> <p>ii. TA = 20% OF (BASIC + HRA)</p>
	<p>4. Write a Trigger that maintains a log of a account (acc_no, name, balance) table. Whenever there is any update of account balance, LogTable maintains AccountNo, Name, amount that is deposited or withdrawn, Type of transaction- if deposite then 'D' – if withdraw then 'W', and Timestamp which includes Day, Date, Time.</p>
Unit-IV	CONCURRENCY CONTROL AND ADVANCED DATABASES
	<p>Concurrency Control: Need, Locking Methods, Deadlocks, Time-stamping Methods, and Optimistic Techniques.</p> <p>Recovery Methods: Shadow-Paging and Log-Based Recovery, Checkpoints, Performance Tuning, Query Optimization with respect to SQL Database.</p> <p>Database Architectures: Centralized and Client-Server Architectures, 2 Tier and 3 Tier Architecture, Introduction to Parallel Databases, Key elements of Parallel Database Processing, Architecture of Parallel Databases, Introduction to Distributed Databases, Architecture of Distributed Databases, Distributed Database Design.</p>
	Practical
	<p>1. Study and design a database with suitable example using following database systems</p> <p>a. A)MySQL</p> <p>b. B)MongoDB</p> <p>Compare above database management systems based on points like efficiency, scalability, characteristics and performance.</p>
	2. Install and configure client and server for MySQL and MongoDB (Show all commands and necessary steps for installation and configuration).
	3. Study the SQLite database and its uses. Also elaborate on building and installing of SQLite.
Unit- V	LARGE SCALE DATA MANAGEMENT
	<p>Emerging Database Technologies: Introduction to No SQL Databases- Internet Databases, Cloud Databases, Mobile Databases, SQLite Database, XML Databases, and MongoDB.</p> <p>Introduction to Big Data and XML: DTD, XML Schemas, XQuery, XPath.</p> <p>JSON: Overview, Data Types, Objects, Schema, JSON with Java/PHP/Ruby/Python.</p> <p>Hadoop: HDFS, Dealing with Massive Datasets-Map Reduce and Hadoop.</p> <p>Introduction to HBase: Overview, HBase Data Model, HBase Region, Hive.</p>
	Practical
	<p>1. Create a database with suitable example using MongoDB and implement</p> <ul style="list-style-type: none"> • Inserting and saving document (batch insert, insert validation) • Removing document • Updating document (document replacement, using modifiers, upserts, updating multiple documents, returning updated documents)

	2. Execute at least 10 queries on any suitable MongoDB database that demonstrates following querying techniques: <ul style="list-style-type: none"> • find and findOne (specific values) • Query criteria (Query conditionals, OR queries, \$not, Conditional semantics) • Type-specific queries (Null, Regular expression, Querying arrays) 		
	3. Execute at least 10 queries on any suitable MongoDB database that demonstrates following: <ul style="list-style-type: none"> • where queries • Cursors (Limits, skips, sorts, advanced query options) • Database commands 		
	4. Implement Map reduce example with suitable example.		
	5. Implement the aggregation and indexing with suitable example in MongoDB. Demonstrate the following: <ul style="list-style-type: none"> • Aggregation framework • Create and drop different types of indexes 		
Unit- VI	DATA WAREHOUSING AND DATA MINING		
	Data Warehousing: Introduction, Evolution of Data Warehouse, Characteristics, Benefits, Limitation of Data Warehousing, Architecture and Components of Data Warehouse, Conceptual Models, Data Mart, OLAP.		
	Data Mining: Process, Knowledge Discovery, Goals of Data Mining, Data Mining Tasks, Association, Classification, Clustering, Big Data (Terminology and examples) Introduction to Machine learning for Big Data and Business Intelligence.		
	Practical		
	1. Implement any Machine learning algorithm for BIG data		
Text Books	Author	Title of Book	Publication
T1	Silberschatz A., Korth H., Sudarshan S	“Database System Concepts”,	6 th Edition, McGraw Hill Publishers, ISBN 0-07-120413-X
T2	S.K.Singh	“Database Systems: Concepts, Design and Application”	Pearson Publication, ISBN-978-81-317-6092-5.
Reference Books			
R1	Kristina Chodorow, Michael Dirolf	“MongoDB: The Definitive Guide”,	O’Reilly Publications
R2	Tom White	“Hadoop: The Definitive Guide”	O’Reilly Publications
R3	Jiawei Han, MichelineKamber, Jian Pei	“Data Mining: Concepts and Techniques”,	Elsevier Publications
R4	Bill Schmarzo	“Big Data:Understanding How Data Powers Big Business”,	Wiley,ISBN:978-81-265-4545-2
R5	Alex Holmes	“Hadoop in Practice”,	DreamTech Press, ISBN : 978-93-5119-150-6
Self-Learning Facilities	Text Books,Reference Books, PPTs		

Web Resources	http://www.tutorialspoint.com		
Research papers for reference	Author	Title of Paper	Journal/Transaction
1			
2			
3			
Contents beyond Syllabus			
Additional Experiments			
Bridging Courses	1. Implement any one Concurrency Control Protocol using MongoDB and .net/Java		
Assignments	2.Design and Implement Student Data Management System using Java &MongoDB database Connectivity. Student data consist of Sid, Sname, Class, City, Marks. and perform Insert, Delete, Update & Search Operation		
	3.Design and Implement Student Data Management System using Java & MySQL database Connectivity. Student data consist of Sid, Sname, Sub1, Sub2, Sub3 marks. And perform Insert, Delete, Update & Search Operation		
	4. Design and Implement Student Data Management System to collect marks of all the students and store it in database. After collecting marks of all students execute a Stored Procedure from Java to Calculate the Percentage marks and store it back to the database.		
Tutorials			
Presentations			

SOFTWARE ENGINEERING & PROJECT MANAGEMENT

Course Title:	Software Engineering & Project Management	Course Number:	Credits: 3
Year: 2018-19		Semester: I	
Designation of Course		Professional Core	
Teaching Scheme: 3Hrs/Week		Tutorial:	
Course Assessment Methods	Direct methods	On-line/In-semester Examination: 50/30 Marks	End Semester Examination: 50/70 Marks
			Practical/Oral/Term Work
	Indirect Methods	Assignments, Presentations	Seminars, Quiz, Q&A session, Group Discussion
Prerequisites	1. Problem solving and object oriented programming. 2. Fundamental of data structures.		
Course Objectives			
1	To understand the nature of software complexity in various application domains, disciplined way of software development and software lifecycle process models.		
2	To introduce principles of agile software development, the SCRUM process and agile practices.		
3	To know methods of capturing, specifying, visualizing and analysing software requirements.		
4	To understand project management through life cycle of the project.		
5	To understand current and future trends and practices in the IT industry.		
6	To learn about project planning, execution, tracking, audit and closure of project.		
Course Outcomes			
CO1	To identify unique features of various software application domains and classify software applications.		
CO2	To choose and apply appropriate lifecycle model of software development		
CO3	To describe principles of agile development, discuss the SCRUM process and distinguish agile process model from other process models.		
CO4	To analyse software requirements by applying various modelling techniques.		
CO5	To list and classify CASE tools and discuss recent trends and research in software engineering.		
CO6	To understand IT project management through life cycle of the project and future trends in IT Project Management.		
Course Contents			
Unit-I	INTRODUCTION TO SOFTWARE ENGINEERING		
	Nature of Software, Software Process, Software Engineering Practice, Software Myths, Generic Process model, Analysis and comparison of Process Models: Waterfall Model, Incremental Models, Evolutionary Models, Concurrent, Specialized Process Models, Personal and Team Process Models, Introduction to Clean Room Software Engineering. Software Quality Assurance (SQA): Verification and Validation, SQA Plans, Software Quality Frameworks, ISO 9000 Models, CMM Model		
Unit-II	REQUIREMENT ANALYSIS		

	Requirements Capturing: requirements engineering (elicitation, specification, validation, negotiation, prioritizing requirements (Kano diagram) - real life application case study. Requirements Analysis: basics, scenario based modeling, UML models: use case diagram and class diagram, data modeling, data and control flow model, behavioral modeling using state diagrams - real life application case study, software Requirement Specification		
Unit-III	PROJECT PLANNING		
	Project initiation, Planning Scope Management, Creating the Work Breakdown Structure, Effort estimation and scheduling: Importance of Project Schedules, Estimating Activity Resources, Estimating Activity Durations, Developing the Schedule using Gantt Charts, Adding Milestones to Gantt Charts, Using Tracking Gantt Charts to Compare Planned and Actual Dates, Critical Path Method, Program Evaluation and Review Technique (PERT) with examples. Planning Cost Management, Estimating Costs, Types of Cost Estimates, Cost Estimation Tools and Techniques, Typical Problems with IT Cost Estimates.		
Unit-IV	AGILE DEVELOPMENT PROCESS		
	Agile Development: Agile manifesto, agility and cost of change, agility principles, myth of planned development, toolset for the agile process. Extreme Programming: XP values, process, industrial XP, SCRUM - process flow, scrum roles, scrum cycle description, product backlog, sprint planning meeting, sprint backlog, sprint execution, daily scrum meeting, maintaining sprint backlog and burn-down chart, sprint review and retrospective. Agile Practices: test driven development, refactoring, pair programming, continuous integration, exploratory testing versus scripted testing.		
Unit- V	PROJECT MANAGEMENT		
	Project monitoring and control: tools for project management, Software tools like Microsoft project management or any other open source tools. The Importance of Project Quality Management: Planning Quality Management, Performing Quality Assurance, Controlling Quality, Tools and Techniques for Quality Control (statistical control, six sigma) The Importance of Project Risk Management, Planning Risk Management, Common Sources of Risk in IT Projects.		
Unit-VI	RECENT TRENDS IN SOFTWARE ENGINEERING AND PROJECT MANAGEMENT		
	Software configuration management: SCM basics, SCM repository, SCM process, SCM tools such as GitHub, CASE – taxonomy, tool-kits, workbenches, environments, components of CASE, categories (upper, lower and integrated CASE tools). Emerging software engineering trends: technology evolution, process trends, collaborative development, test-driven development, global software development challenges Project Management trends: CRM, ERP: Basic concepts, Advantages and limitations, SAP, Business process reengineering, International Project Management, Case studies.		
Text Books	Author	Title of Book	Publication
T1	Roger S Pressman	Software Engineering: A Practitioner's Approach	Mcgraw-Hill
T2	Joseph Phillips	IT Project Management –On Track From Start to Finish	Tata McGraw-Hill
Reference Books			
R1	Pankaj Jalote	Software Engineering: A Precise Approach	Wiley India

R2	Marchewka	Information Technology Project Management	Wiley India
R3	Chris Dawson with Ben Straub	Building Tools with GitHub	O'Reilly, Shroff publishers,
R4	P.C. Tripathi, P.N. Reddy	Principles of Management	Tata McGraw Hill
R5			
Self-Learning Facilities, Web Resources, Research papers for reference			
Contents beyond Syllabus			
Additional Experiments			
Bridging Courses			
Tutorials			
Presentations			

OPERATING SYSTEM

Course Title:	OPERATING SYSTEM	Course Number:314444	Credits :4
Designation of Course	Professional Core		
Teaching Scheme: 4 Hrs/Week		Laboratories: 4 Hrs/Week	
Course Assessment Methods	Direct methods	In-sem Examination: 30 Marks	Theory/End Semester Examination: 70 Marks
		Oral Mock exam	Practical Mock Exam
	Indirect Methods	Assignments	Practical
Prerequisites	Computer Organization and Data Structure Fundamentals of Data Structures		
Introduction of Course			
OPERATING SYSTEM			
Course Objectives			
1	To Introduce basic concepts and functions of modern operating systems.		
2	To Understand the concept of process, and thread management.		
3	To understand the scheduling of processes and threads.		
4	To understand the concept of concurrency control.		
5	To understand the concept of I/O and File management.		
6	To Understand various Memory management techniques.		
Course Outcomes : Students will be able to			
CO1	Fundamental understanding of the role of Operating Systems.		
CO2	To understand the concept of a process and thread.		
CO3	To apply the cons of process/thread scheduling.		
CO4	To apply the concept of process synchronization, mutual exclusion and the deadlock.		
CO5	To realize the concept of I/O management and File system.		
CO6	To understand the various memory management techniques.		
Course Contents			
Unit-I	OPERATING SYSTEM OVERVIEW		
	Operating System Objectives and Functions, The Evolution of Operating Systems, Developments Leading to Modern Operating Systems, Virtual Machines. BASH Shell scripting: Basic shell commands, shell as a scripting language.		
	Practical		
	Shell programming		
	(1) Write a program to implement an address book with options given below: a) Create address book. b) View address book. c) Insert a record. d) Delete a record. e) Modify a record. f) Exit.		
Unit-II	PROCESS DESCRIPTION AND CONTROL		
	Process: Concept of a Process, Process States, Process Description, Process Control (Process creation, Waiting for the process/processes, Loading programs into processes and Process Termination), Execution of the Operating System.		
	Threads: Processes and Threads, Concept of Multithreading, Types of Threads, Thread programming Using Pthreads.		
	Scheduling: Types of Scheduling, Scheduling Algorithms, and Thread Scheduling.		
	Practical		
	(1) Process control system calls: The demonstration of fork, execve and wait system calls		

	along with zombie and orphan states. (2) Thread management using pthread library. (3) Implementing a CPU scheduling policy in a Linux OS.		
Unit-III	CONCURRENCY CONTROL		
	Process/thread Synchronization and Mutual Exclusion: Principles of Concurrency, Requirements for Mutual Exclusion, Mutual Exclusion: Hardware Support, Operating System Support (Semaphores and Mutex), Programming Language Support (Monitors). Classical synchronization problems: Readers/Writers Problem, Producer and Consumer problem, Interprocess communication (Pipes, shared memory: system V). Deadlock: Principles of Deadlock, Deadlock Modelling, Strategies to deal with deadlock: The Ostrich Algorithm, Deadlock Prevention, Deadlock Avoidance, Deadlock detection and recovery, An Integrated Deadlock Strategy, Example: Dining Philosophers Problem.		
	Practical		
	(1).Thread synchronization using counting semaphores and mutual exclusion using mutex. Application to demonstrate: producer-consumer problem with counting semaphores and mutex. (2). Thread synchronization and mutual exclusion using mutex. Application to demonstrate: Reader-Writer problem with reader priority. (3).Deadlock Avoidance Using Semaphores: Implement the deadlock-free solution to Dining Philosophers problem to illustrate the problem of deadlock and/or starvation that can occur when many synchronized threads are competing for limited resources. (4)Inter process communication in Linux using PIPE, FIFO and SIGNALS.		
Unit-IV	MEMORY MANAGEMENT		
	Memory Management: Memory Management Requirements, Memory Partitioning: Fixed Partitioning, Dynamic Partitioning, Buddy System, Relocation, Paging, Segmentation. Virtual Memory: Hardware and Control Structures, Operating System Software		
	Practical		
	Implementing a memory management policy in a Linux OS.		
Unit- V	INPUT/OUTPUT AND FILE MANAGEMENT		
	I/O Management and Disk Scheduling: I/O Devices, Organization of the I/O Function, Operating System Design Issues, I/O Buffering, Disk Scheduling(FIFO, SSTF, SCAN, C-SCAN, LOOK, C-LOOK), Disk Cache. File Management: Overview, File Organization and Access, File Directories, File Sharing, Record Blocking, Secondary Storage Management.		
	Practical		
	Implement an assignment using File Handling System Calls (Low level system calls like open, read, write, etc).		
Unit-VI	The LINUX Operating System		
	Linux Design Principles, Linux Booting Process, Kernel Modules, Process Management, Scheduling, Memory Management, File Systems, Input and Output, Inter-process Communication.		
	Practical		
	(1) Linux Kernel configuration, compilation and rebooting from the newly compiled kernel. (2) Implement a new system call, add this new system call in the Linux kernel (any kernel source, any architecture and any Linux kernel distribution) and demonstrate the use of same.		
Text Books	Author	Title of Book	Publication
T1	William Stallings	Operating System: Internals and Design Principles, 8th Edition.	Prentice Hall

T2	Abraham Silberschatz, Peter Baer Galvin and Greg Gagne	Operating System Concepts, 9th Edition.	John Wiley & Sons
T3	Andrew S. Tanenbaum & Herbert Bos,	Operating System 4th Edition	Pearson
Reference Books			
R1	Tom Adelstein and Bill Lubanovic	Linux System Administration, Inc., 1st Edition, 2007	O'Reilly Media
R2	Harvey M. Deitel	Operating Systems, , 3rd Edition.	Prentice Hall
R3	Thomas W. Doeppner	Operating System in depth: Design and Programming	WILEY
R4	Mendel Cooper	Advanced Shell Scripting	Linux Documentation Project
Self-Learning Facilities			
Web Resources			
Research papers for reference	Author	Title of Paper	Journal/Transaction
1			
2			
3			
Contents beyond Syllabus			
Additional Experiments			
Bridging Courses			
Assignments			
1			
2			
3			
4			
5			
Tutorials	Assignment 1 on unit 1		
	Assignment 2 on unit 2		
Presentations			

HUMAN-COMPUTER INTERACTION

Course Title:	Human-Computer Interaction	Course Number:314445	Credits : 3
Designation of Course	Professional Core		
Teaching Scheme: 3 Hrs/Week			
Course Assessment Methods	Direct methods	In-sem Examination: 30 Marks	Theory/End Semester Examination: Marks:70
		Term-work	Practical/Oral
	Indirect Methods	Assignments, Presentations	Seminars, Quiz, Q&A session, Group Discussion
Prerequisites	Problem Solving and Object Oriented Technologies.		
Introduction of Course			
Course Objectives			
1	To introduce to the field of human-computer-interaction study.		
2	To gain an understanding of the human part of human-computer-interactions.		
3	To learn to do design and evaluate effective human-computer-interactions.		
4	To study HCI models and theories.		
5	To understand HCI design processes.		
6	To apply HCI to real life use cases.		
Course Outcomes			
CO1	To explain importance of HCI study and principles of user-centred design (UCD) approach.		
CO2	To develop understanding of human factors in HCI design.		
CO3	To develop understanding of models, paradigms and context of interactions.		
CO4	To design effective user-interfaces following a structured and organized UCD process.		
CO5	To evaluate usability of a user-interface design.		
CO6	To apply cognitive models for predicting human-computer-interactions.		
Course Contents			
Unit-I	INTRODUCTION		
	What is HCI?, Disciplines involved in HCI, Why HCI study is important? The psychology of everyday things, Principles of HCI, User-centred Design.		
	Practical		
Unit-II	UNDERSTANDING THE HUMAN		
	Input-output channels, Human memory, Thinking: Reasoning and Problem Solving, Human emotions, Individual differences, Psychology and Design.		
	Practical		
Unit-III	UNDERSTANDING THE INTERACTION		
	Models of interaction, Ergonomics, Interaction styles, WIMP Interface, Interactivity, Context of interaction, User experience, Paradigms of Interactions.		
	Practical		
Unit-IV	HCI - DESIGN PROCESS		
	What is interaction design?, The software design process, User focus, Scenarios, Navigation Design, Screen Design, Prototyping techniques, Wire-Framing, understanding the UI Layer		

	and Its Execution Framework, Model-View-Controller (MVC) Framework.		
	Practical		
Unit- V	HCI - DESIGN RULES , GUIDELINES AND EVALUATION TECHNIQUES		
	Principles that support usability, Design standards, Design Guidelines, Golden rules and heuristics, Using toolkits, User interface management system (UIMS), Goals of evaluation, Evaluation Criteria, Evaluation through expert analysis, Evaluation through user participation, Choosing an Evaluation Method.		
	Practical		
Unit-VI	HCI MODELS AND THEORIES		
	Goal and task hierarchy model, Linguistic model, Physical and device models, Cognitive architectures, Hierarchical task analysis (HTA), Uses of task analysis, Diagrammatic dialog design notations, Computer mediated communication, Ubiquitous Computing, Finding things on web Future of HCI.		
	Practical		
Text Books	Author	Title of Book	Publication
T1	Alan Dix.	Human Computer Interaction.	Pearson Education, ISBN 978-81-317-1703-5.
T2	Gerard Jounghyun Kim.	Human-Computer Interaction: Fundamentals and Practice	CRC Press. ISBN 978-1-4822-3390-
Reference Books			
R1	Ben Shneiderman; Catherine Plaisant; Maxine Cohen; Steven Jacobs	Designing the User Interface: Strategies for Effective Human-Computer Interaction	Pearson Education, ISBN 978-1-292-03701-1.
R2	Donald A. Norman	The Design of Everyday Things Basic Books.	ISBN 978-0-465-07299-6.
R3	Jeff Johnson	Designing with the Mind in Mind: Simple Guide to Understanding User Interface Design Guidelines	Elsevier. ISBN 978-0-12-411556-9
R4	Alan Cooper; Robert Reimann; David Cronin; Christopher Noessel	About Face: The Essentials of Interaction Design	Wiley. ISBN 978-1-118-76658-3.
R5	Alan Cooper	The Inmates are running the Asylum, Sam's	ISBN 978-0-672-31649-4.
R6	John M. Carroll	HCI Models, Theories, and Frameworks: Toward a Multidisciplinary Science	Morgan Kaufmann. ISBN 978-0-08-049141-7.
R7	Alan Cooper, Robert Reimann, David Cronin, Christopher Noessel,	About Face: The Essentials of Interface Design	Wiley India, ISBN : 9788126559718,4th Ed
R8	Rogers, Sharp, Preece,	Interaction Design: Beyond Human Computer Interaction,	Wiley India, ISBN : 9788126544912,3ed
R9	Wilbert O.Galitz,	The Essential Guide to	Wiley India,

		user Interface Design,	ISBN: 9788126502806
Self-Learning Facilities	NPTEL Lecture Series		
Web Resources	1. http://hcibib.org/ 2. Android Design Guidelines - https://developer.android.com/guide/practices/ui_guidelines/index.html 3. iOS Human Interface Guidelines - https://developer.apple.com/ios/human-interfaceguidelines/overview/design-principles/ 4. MacOS Human Interface Guidelines - https://developer.apple.com/library/content/documentation/UserExperience/Conceptual/OSX_HIGuidelines/		
Research papers for reference	Author	Title of Paper	Journal/Transaction
1			
2			
3			
Contents beyond Syllabus			
Additional Experiments			
Bridging Courses			
Assignments			
1			
2			
3			
4			
5			
Presentations			

TE IT(2015 Course)

Sem. II

PVG'S COET, Pune

COMPUTER NETWORK TECHNOLOGY

Course Title:	Computer Network Technology	Course Number: 314450	Credits : 3
Designation of Course	Professional Core		
Teaching Scheme:3Hrs/Week		Laboratories: 2 Hrs/Week	
Course Assessment Methods	Direct methods	In-sem Examination: 30 Marks	Theory/End Semester Examination: 70 Marks
		Term-work	Practical/Oral
	Indirect Methods	Assignments, Presentations	Seminars, Quiz, Q&A session, Group Discussion
Prerequisites	Foundation of Communications and Computer Network		
Introduction of Course			
Course Objectives			
1	To understand services offered at different layers of network.		
2	To understand protocol used at different layers of network.		
3	To fathom wireless network and different wireless standards.		
4	To recognize differences in between different wireless networks and to learn different mechanism used at layers of wireless network.		
5	To know the applications of network and use the understood concepts for new application development.		
6	To explore recent trends in networking.		
Course Outcomes			
CO1	To know Responsibilities, services offered and protocol used at each layer of network.		
CO2	To understand different addressing techniques used in network.		
CO3	To know the difference between different types of network.		
CO4	To know the different wireless technologies and IEEE standards.		
CO5	To use and apply the standards and protocols learned, for application development.		
CO6	To understand and explore recent trends in network domain		
Course Contents			
Unit-I	NETWORK LAYER		
	Network Layer Services, IPv4 Addresses: Classful and Classless Addressing, Special Addresses, NAT, Subnetting, Supernetting, Delivery and Forwarding of IP Packet, Structure of Router, IPv4: Fragmentation, Options, Checksum, ARP: Address Mapping, ARP Protocol, RARP, DHCP, ICMPv4, Unicast Distance Vector Routing, Link State Routing, Unicast Routing Protocols: RIP,EIGRP,OSPF,BGP, IPv6 Addressing.		
	Practical		
	1.Explore and Study of TCP/IP utilities and Network Commands on Linux. a) Ping g) Tracert/Traceroute/Tracepath b) ipconfig / ifconfig h) NSlookup		

	<p>c) Hostname i) Arp d) Whois j) Finger e) Netstat k) Port Scan / nmap f) Route</p> <p>2. Using a Network Simulator (e.g. packet tracer) configure</p> <ul style="list-style-type: none"> • Sub-netting of a given network • Super-netting of a given networks <p>3. Using a Network Simulator (e.g. packet tracer) configure</p> <ul style="list-style-type: none"> • A router using router commands • Access Control lists – Standard & Extended <p>4. Using a Network Simulator (e.g. packet tracer) configure</p> <ul style="list-style-type: none"> • EIGRP – Explore Neighbor-ship Requirements and Conditions, its K Values Metrics Assignment and Calculation, • RIPv2 and EIGRP on same network. • WLAN with static IP addressing and DHCP with MAC security and filters <p>5. Using a Network Simulator (e.g. packet tracer) configure</p> <ul style="list-style-type: none"> • VLAN, Dynamic trunk protocol and spanning tree protocol • OSPF – Explore Neighbor-ship Condition and Requirement, Neighbor-ship states, OSPF Metric Cost Calculation. • Network Address Translation : Static, Dynamic & PAT (Port Address translation)
Unit-II	TRANSPORT LAYER
	<p>Transport Layer Services, UDP: Datagram, Services, Applications, TCP: Services, Features, Segment, TCP Connection, Window in TCP, Flow control, Congestion Control, Congestion Control Algorithms, Leaky Bucket, Token Bucket and QoS, TCP Timers, Options, TCP Package, Applications, SCTP: Features, Services, Packet Format, Socket: TCP and UDP Socket, Applications.</p> <p>Practical</p> <p>1. Socket Programming in C/C++ on Linux.</p> <ul style="list-style-type: none"> • TCP Client , TCP Server • UDP Client , UDP Server
Unit-III	APPLICATION LAYER
	<p>Client Server Paradigm: Communication using TCP and UDP, Peer to Peer Paradigm, Application Layer Protocols: DNS, FTP, TFTP, HTTP, SMTP, POP, IMAP, MIME, Network Management: SNMP.</p> <p>Practical</p> <p>1. Introduction to server administration (server administration commands and their applications) and configuration any three of below Server : (Study/Demonstration Only)</p> <ul style="list-style-type: none"> • FTP, • Web Server, • DHCP, • Telnet, • Mail, • DNS
Unit-IV	WIRELESS STANDARDS
	<p>Electromagnetic Spectrum: Spectrum Allocation, Radio Propagation Mechanism, Characteristics of Wireless Channel, Wireless LANs: Architectural Comparison, Characteristics, Access Control, IEEE 802.11: Architecture, MAC Sub Layer, Addressing Mechanism, Physical Layer, Bluetooth: Architecture, Layers, IEEE 802.16/WiMax: Services, Architecture, Layers, Differences between Bluetooth, IEEE 802.11 and</p>

	IEEE 802.16.		
	Practical		
Unit- V	ADHOC WIRELESS NEWTORK		
	Infrastructure Network and Infrastructure-less Wireless Networks, Issues in Adhoc Wireless Network, AdhocNetwork MAC Layer: Design Issues, Design Goal, Classification, MACAW, Adhoc Network Routing Layer: Issues in Designing a Routing Protocol for Ad-hoc Wireless Networks – Classifications of Routing Protocols, DSDV, AODV, DSR, Adhoc Transport Layer: Issues in Designing a Transport Layer Protocol for Ad hoc Wireless Networks – Design Goals of a Transport Layer Protocol for Ad hoc Wireless Networks –Classification of Transport Layer Solutions, TCP over Adhoc Wireless Networks.		
	Practical		
	1.Using any open source Network Simulator, Implement <ul style="list-style-type: none">• MANET / Wireless Sensor Network		
Unit-VI	RECENT TRENDS IN COMMUNICATION NETWORKS		
	Satellite Network: Operation, GEO Satellites, MEO Satellites, LEO Satellites, Wireless Sensor Network: Functioning, Characteristics, Operation, Cluster Management, Computational Grid: Design, Issues, Internet of Things: Vision, Trends, Significance, Technical Building Blocks, Issues and Challenges, Applications, IoE. Software Defined Network: SDN Implication for research and innovation, Genesis of SDN, Characteristics of SDN, SDN Operations, SDN Devices, SDN Controllers, SDN Application, OpeFlow Overview, Network Function Virtualization: Introduction, Applications, Network Neutrality: Need, Requirements (e Reference from research papers and web)		
	Practical		
	1. Write a program using Arduino / Rasberry Pi Kit for Demonstration of IOT Application on any one of the following Topics. <ul style="list-style-type: none">• Appliance Remote Control• Time Lapse Camera Controller• Security / Automation Sensors• The Traffic Light Controller• Temperature Controller		
Text Books	Author	Title of Book	Publication
T1	Behrouz A. Forouzan	TCP/IP Protocol Suite	McGraw Hill Education, 4th Edition
T2	C. Siva Ram Murthy and B. S. Manoj	Ad Hoc Wireless Networks: Architectures and Protocols	Prentice Hall
T3	Behrouz A. Forouzan	Data Communication and Networking	McGraw Hill Education, 5th Edition
Reference Books			
R1	Andrew S. Tanenbaum, David J. Wethrall	Computer Network	Pearson Education
R2	James F. Kurose and Keith W. Ross	Computer Networking: A Top-Down Approach Featuring the Internet	Pearson Education
R3	Charles E. Perkins	Adhoc Networking	Pearson Education
R4	Andrea Goldsmith	Wireless Communication	Cambridge University Press
R5	Mayank Dave	Computer Network	Cengage Learning
R6	C. K. Toh	Ad Hoc Mobile Wireless	TohPrentice Hall

		Networks Protocols and Systems	
R7	Paul Goransson, Chuck Black	Software Defined Networks: A Comprehensive Approach	Morgan Kaufmann
R8	Natalia Olifer, Victor Olifer	Computer Networks: Principles, Technologies and Protocols for Network Design	Wiley India
R9	KazemSohraby, Daniel Minoli, TaiebZnati	Wireless Sensor Networks: Technology, Protocols and Applications	Wiley India
R10	P. Nicopolitidis, M.S. Obaidat, G.I. Papadimitriou, A.S. Pomportsis	Wireless Networks	Wiley India
Self-Learning Facilities	NPTEL Lecture Series		
Web Resources	www.computernetworkingnotes.com		
Research papers for reference	Author	Title of Paper	Journal/Transaction
1	Aarti, Dr.S.S.Tyagi	Study of MANET: Characteristics, challenges, Application and Security Attacks	International Journal of Adv Research in CS & SE.
2	Mohit Kumar , Rashmi Mishra	An overvies of MANET: Histry, challenges, and Applications	IJCSE
Contents beyond Syllabus			
Additional Experiments	1. Socket Chatting Application		
	2. Protocol headers using Wire shark		
Bridging Courses			
Assignments			
1	Theory Assignment on Unit 1		
2	Theory Assignment on Unit 2		
3	Theory Assignment on Unit 3		
4	Theory Assignment on Unit 4		
5	Theory Assignment on Unit 5		
6	Theory Assignment on Unit 6		
Tutorials			
Presentations	PPT Prepared for all units		

SYSTEM PROGRAMMING

Course Title:	SYSTEM PROGRAMMING	Course Number: 314451	Credits : 4
Designation of Course	Professional Core		
Teaching Scheme: 4 Hrs/Week		Laboratories: 4 Hrs/Week	
Course Assessment Methods	Direct methods	In-sem Examination: 30 Marks	Theory/End Semester Examination: 70 Marks
	Indirect Methods	Assignments, Presentations, Class Test	Practical and Oral(SL-II)
Prerequisites	1. Computer Organization and architecture. 2. Processor Architecture and Interfacing. 3. Fundamentals of Data Structures, Data Structures and Files. 4. Theory of Computation: DFA, NFA, Regular expressions, Grammars.		
Introduction of Course			
Course Objectives			
1	To study and understand different system software like Assembler, Macro-processor and Loaders / Linkers.		
2	To design and develop useful system software.		
3	To study and understand compiler design.		
4	To understand semantic analysis and storage allocation in compilation process.		
5	To understand different code generation techniques.		
6	To study different code optimization methods.		
Course Outcomes			
CO1	To learn independently modern software development tools and creates novel solutions for language processing applications.		
CO2	To design and implement assemblers and macro processors.		
CO3	To use tool LEX for generation of Lexical Analyzer.		
CO4	To use YACC tool for generation of syntax analyser		
CO5	To generate output for all the phases of compiler.		
CO6	To apply code optimization in the compilation process.		
Course Contents			
Unit-I	INTRODUCTION TO SYSTEMS PROGRAMMING AND ASSEMBLERS 08 Hours		
	: Need of System Software, Components of System Software, Language Processing Activities, Fundamentals of Language Processing. Assemblers: Elements of Assembly Language Programming, A simple Assembly Scheme, Pass structure of Assemblers, Design of Two Pass Assembler, Single pass assembler.		
	Practical: 1. Write a program to implement Pass-I of Two-pass assembler for Symbols and Literal processing (For hypothetical instruction set from Dhamdhare) considering following cases i. Forward references ii. DS and DC statement iii. START, EQU, LTORG, END.		

	<p>iv. Error handling: symbol used but not defined, invalid instruction/register etc.</p> <p>2. Write a program to implement Pass-II of Two-pass assembler for output of Assignment 1 (The subject teacher should provide input file for this assignment)</p>
Unit-II	MACROPROCESSORS, LOADERS AND LINKERS
	<p>: Macro Definition and call, Macro Expansion, Nested Macro Calls and definition, Advanced Macro Facilities, Design of two-pass Macro Processor.</p> <p>Loaders: Loader Schemes, Compile and Go, General Loader Scheme, Absolute Loader Scheme, Subroutine Linkages, Relocation and linking concepts, Self-relocating programs, Relocating Loaders, Direct Linking Loaders, Overlay Structure.</p>
	<p>Practical</p> <p>3. Study Assignment for Macro Processor. (Consider all aspects of Macro Processor)</p>
Unit-III	INTRODUCTION TO COMPILER
	<p>Phase structure of Compiler</p> <p>Lexical Analyzer: The Role of the Lexical Analyzer, Input Buffering. Specification of Tokens, Recognition of Tokens, Lexical Analyzer, RE to DFA Conversion, Lexical Errors.</p> <p>LEX: LEX Specification, Generation of Lexical Analyzer by LEX.</p>
	<p>Practical:</p> <p>4. Write a program to implement Lexical Analyzer for subset of C.</p>
Unit-IV	PARSERS
	<p>Role of parsers, Top down parsers, recursive descent parser, predictive parser, LL(K) parsers, Bottom up Parsers - Operator Precedence Parser, Shift Reduce - SLR, LR(K), LALR, Error Detection and Recovery in Parser, YACC specification and features Automatic construction of Parser(YACC).</p>
	<p>Practical:</p> <p>5. Write a program to implement a Recursive Descent Parser .</p> <p>6. Write a program to implement calculator using LEX and YACC.</p>
Unit- V	SEMANTIC ANALYSIS AND STORAGE ALLOCATION
	<p>Need, Syntax Directed Translation, Syntax Directed Definitions, Translation of assignment Statements, iterative statements, Boolean expressions, conditional statements, Type Checking and Type conversion. Intermediate Code Formats: Postfix notation, Parse and syntax trees, Three address code, quadruples and triples.</p> <p>Storage Allocation: Storage organization and allocation strategies.</p>
	<p>Practical</p> <p>7. Intermediate code generation using LEX & YACC for while and Switch Case statements.</p>
Unit-VI	CODE GENERATION AND OPTIMIZATION
	<p>Code Generation: Code generation Issues. Basic blocks and flow graphs, A Simple Code Generator.</p> <p>Code Optimization: Machine Independent: Peephole optimizations: Common Sub-expression elimination, Removing of loop invariants, Induction variables and Reduction in strengths, use of machine idioms, Dynamic Programming Code Generation.</p> <p>Machine dependent Issues: Assignment and use of registers, Rearrangement of Quadruples for code optimization.</p>

Text Books	Author	Title of Book	Publication
T1	D.M. Dhamdhere	, "Systems Programming and Operating Systems"	Tata McGraw-Hill, ISBN-13:978-0-07-463579-7
T2	Alfred V. Aho, Ravi Sethi, Ralfrey D. Ullman,	"Compilers Principles, Techniques, and Tools"	Addison Wesley, ISBN 981-235-885-4
T3	John J Donovan	"Systems Programming"	Tata McGraw-Hill Edition 1991, ISBN 0-07-460482-1
Reference Books			
R1	Leland L. Beck,	"System Software An Introduction to Systems Programming" 3rd Edition,	Person Education, ISBN 81-7808-036-2
Self-Learning Facilities	Books, ppts		
Web Resources	-		
Research papers for reference	Author	Title of Paper	Journal/Transaction
1			
2			
Contents beyond Syllabus			
Additional Experiments			
Bridging Courses			
1			
2			
Tutorials			
Presentations			

DESIGN AND ANALYSIS OF ALGORITHMS

Course Title:	DESIGN AND ANALYSIS OF ALGORITHMS		Course Number: 314452	Credits : 4
Year: 2018-19			Semester: II	
Designation of Course			Professional Core	
Teaching Scheme: 4 Hrs/Week			Tutorial:	
Course Assessment Methods	Direct methods	In-semester Examination: 30 Marks	End Semester Examination: 70 Marks	
			Practical	
	Indirect Methods	Assignments	Class Test	
Prerequisites	1. Fundamentals of Data Structures, Data Structures and Files. 2. Discrete Structures. 3. Basic mathematics: Induction, probability theory, logarithms.			
Course Objectives				
1	To understand the problem solving and problem classification.			
2	To know the basics of computational complexity analysis and various algorithm design strategies			
3	To provide students with solid foundations to deal with a wide variety of computational problems			
4	To provide a thorough knowledge of the most common algorithms and data structures.			
5	To analyse a problem and identify the computing requirements appropriate for its solutions			
6	To understand the design of parallel algorithms.			
Course Outcomes				
CO1	To calculate computational complexity using asymptotic notations for various algorithms			
CO2	To apply Divide & Conquer as well as Greedy approach to design algorithms			
CO3	To practice principle of optimality			
CO4	To illustrate different problems using Backtracking			
CO5	To compare different methods of Branch and Bound strategy.			
CO6	To explore the concept of P, NP, NP-complete, NP-Hard and parallel algorithms.			
Course Contents				
Unit-I	INTRODUCTION			
	Brute Force method: Introduction to Brute Force method & Exhaustive search, Brute Force solution to 8 queens' problem. Proof Techniques: Minimum 2 examples of each: Contradiction, Mathematical Induction, Direct proofs, Proof by counterexample, Proof by contraposition. Analysis of Algorithm: Efficiency-Analysis framework, asymptotic notations – big O, theta and omega. Amortized Analysis: Aggregate, Accounting & Potential method with the example of stack operations. Analysis of Non-recursive and recursive algorithms: Solving Recurrence Equations (Homogeneous and nonhomogeneous).			
	Practical/Tutorial			
Unit-II	DIVIDE AND CONQUER AND GREEDY METHOD			
	Divide & Conquer: General method, Control abstraction, Merge sort, Quick Sort – Worst, Best and average case. Binary search, Finding Max-Min, Large integer Multiplication (for all above algorithms analysis to be done with recurrence). Greedy Method: General method and characteristics, Prim's method for MST , Kruskal's method for MST (using $n \log n$ complexity), Dijkstra's Algorithm, Optimal storage on			

	tapes, Fractional Knapsack problem, Job Sequencing		
	Practical/Tutorial		
	1. Write a program to find Maximum and Minimum element in an array using Divide and Conquer strategy and verify the time complexity. 2. Write a program to solve optimal storage on tapes problem using Greedy approach.		
Unit-III	DYNAMIC PROGRAMMING		
	General strategy, Principle of optimality, 0/1 knapsack Problem, Bellman-Ford Algorithm , Multistage Graph problem, Optimal Binary Search Trees, Travelling Salesman Problem.		
	Practical/Tutorial		
	1. Write a program to implement Bellman-Ford Algorithm using Dynamic Programming and verify the time complexity 2. Write a program to solve the travelling salesman problem and to print the path and the cost using Dynamic Programming.		
Unit-IV	BACKTRACKING		
	General method, Recursive backtracking algorithm, Iterative backtracking method. 8-Queen problem, Sum of subsets, Graph colouring, Hamiltonian Cycle , 0/1 Knapsack Problem.		
	Practical/Tutorial		
	1. Write a recursive program to find the solution of placing n queens on chessboard so that no two queens attack each other using Backtracking		
Unit- V	BRANCH AND BOUND		
	The method, Control abstractions for Least Cost Search, Bounding, FIFO branch and bound, LC branch and bound, 0/1 Knapsack problem – LC branch and bound and FIFO branch and bound solution, Travelling sales person problem		
	Practical/Tutorial		
	1. Write a program to solve the travelling salesman problem and to print the path and the cost using Branch and Bound.		
Unit-VI	COMPUTATIONAL COMPLEXITY AND PARALLEL ALGORITHMS		
	Computational Complexity: Non Deterministic algorithms, The classes: P, NP, NP Complete, NP Hard, Satisfiability problem, Proofs for NP Complete Problems: Clique, Vertex Cover. Parallel Algorithms: Introduction, models for parallel computing, computing with complete binary tree, Pointer doubling algorithm.		
	Practical/Tutorial		
Text Books	Author	Title of Book	Publication
T1	Horowitz and Sahani	Fundamentals of computer Algorithms	Galgotia, ISBN 81-7371-612-9.
T2	S. Sridhar	Design and Analysis of Algorithms, Oxford	ISBN 10 : 0-19-809369-1

Reference Books			
R1	Thomas H Cormen and Charles E.L Leiserson	Introduction to Algorithm	PHI, ISBN:81-203-2141-3.
R2	R. C. T. Lee, SS Tseng, R C Chang, Y T Tsai Tata McGraw Hill	Introduction to Design and Analysis of Algorithms, A Strategic approach	ISBN-13: 978-1-25-902582-2. ISBN-10: 1-25-902582-9
R3	AnanyLevitin	Introduction to the Design & Analysis of Algorithm	Pearson, ISBN 81- 7758-835-4
R4	Steven S Skiena	The Algorithm Design Manua	Springer, ISBN 978-81-8489-865-1.
R5	George T. Heineman, Gary Pollice, Stanley Selkow	Algorithms in a Nutshell, A Desktop Quick Reference	O'Reilly, ISBN: 9789352133611
R6	Gilles Brassard, Paul Bratle	Fundamentals of Algorithms	Pearson, ISBN 978-81-317-1244-3.
R7	Michael T. Goodrich, Roberto Tamassia	Algorithm Design: Foundations, Analysis and Internet Examples,	Wiley India, ISBN: 9788126509867
R8	Rod Stephens, Essential	Algorithms: A Practical Approach to Computer Algorithms	Wiley India, ISBN: 9788126546138
Self-Learning Facilities, Web Resources, Research papers for reference			
Contents beyond Syllabus			
Additional Experiments			
Bridging Courses			
Tutorials			
Presentations			

CLOUD COMPUTING

Course Title:		Cloud Computing	Course Number: 314453	Credits : 3
Year: TE		Semester: IV		
Designation of Course		Professional Core		
Teaching Scheme: 3 Hrs/Week		Tutorial:		
Course Assessment Methods	Direct methods	In-semester Examination: 30 Marks	End Semester Examination: 70 Marks	
			Practical/Oral/Term Work	
	Indirect Methods	Assignments, Presentations	Seminars, Quiz, Q&A session, Group Discussion	
Prerequisites		1. Operating Systems. 2. Fundamentals of Computer Networks.		
Course Objectives				
1	To become familiar with Cloud Computing and its ecosystem.			
2	To learn basics of virtualization and its importance.			
3	To evaluate in-depth analysis of Cloud Computing capabilities			
4	To give technical overview of Cloud Programming and Services.			
5	To understand security issues in cloud computing.			
6	To be exposed to Ubiquitous Cloud and Internet of Things.			
Course Outcomes				
CO1	To understand the need of Cloud based solutions.			
CO2	To understand Security Mechanisms and issues in various Cloud Applications			
CO3	To explore effective techniques to program Cloud Systems.			
CO4	To understand current challenges and trade-offs in Cloud Computing.			
CO5	To find challenges in cloud computing and delve into it to effective solutions.			
CO6	To understand emerging trends in cloud computing.			
Course Contents				
Unit-I		FUNDAMENTALS OF CLOUD COMPUTING		
		Origins and Influences, Basic Concepts and Terminology, Goals and Benefits, Risks and Challenges, Roles and Boundaries, Cloud Characteristics, Cloud Delivery Models, Cloud Deployment Models, Federated Cloud/Intercloud, Types of Clouds. Cloud-Enabling Technology: Broadband Networks and Internet Architecture, Data Center Technology, Virtualization Technology, Web Technology, Multitenant Technology, Service Technology.		
		Practical/Tutorial		
Unit-II		VIRTUALIZATION AND COMMON STANDARDS IN CLOUD COMPUTING		
		Implementation Levels of Virtualization, Virtualization Structures/Tools and Mechanisms, Types of Hypervisors, Virtualization of CPU, Memory, and I/O Devices, Virtual Clusters and Resource Management, Virtualization for Data-Center Automation. Common Standards: The Open Cloud Consortium, Open Virtualization Format, Standards for Application Developers: Browsers (Ajax), Data (XML, JSON), Solution Stacks (LAMP and LAPP), Syndication (Atom, Atom Publishing Protocol, and RSS), Standards for Security.		
		Practical/Tutorial		
Unit-III		CLOUD PROGRAMMING, ENVIRONMENTS AND APPLICATIONS		

	Features of Cloud and Grid Platforms, Programming Support of Google App Engine, Programming on Amazon AWS and Microsoft Azure, Emerging Cloud Software Environments, Understanding Core OpenStack Ecosystem. Applications: Moving application to cloud, Microsoft Cloud Services, Google Cloud Applications, Amazon Cloud Services, Cloud Applications (Social Networking, E-mail, Office Services, Google Apps, Customer Relationship Management).		
	Practical/Tutorial		
Unit-IV	CLOUD SECURITY AND ISSUES		
	Basic Terms and Concepts, Threat Agents, Cloud Security Threats and Attacks, Additional Considerations. Cloud Security Mechanisms: Encryption, Hashing, Digital Signature, Public Key Infrastructure (PKI), Identity and Access Management (IAM), Single Sign-On (SSO), Hardened Virtual Server Images. Cloud Issues: Stability, Partner Quality, Longevity, Business Continuity, Service-Level Agreements, Agreeing on the Service of Clouds, Solving Problems, Quality of Service, Regulatory Issues and Accountability		
	Practical/Tutorial		
Unit- V	UBIQUITOUS CLOUDS AND THE INTERNET OF THINGS		
	Cloud Trends in Supporting Ubiquitous Computing, Performance of Distributed Systems and the Cloud, Enabling Technologies for the Internet of Things (RFID, Sensor Networks and ZigBee Technology, GPS), Innovative Applications of the Internet of Things (Smart Buildings and Smart Power Grid, Retailing and Supply-Chain Management, Cyber-Physical System), Online Social and Professional Networking.		
	Practical/Tutorial		
Unit-VI	FUTURE OF CLOUD COMPUTING		
	How the Cloud Will Change Operating Systems, Location-Aware Applications, Intelligent Fabrics, Paints, and More, The Future of Cloud TV, Future of Cloud-Based Smart Devices, Faster Time to Market for Software Applications, Home-Based Cloud Computing, Mobile Cloud, Autonomic Cloud Engine, Multimedia Cloud, Energy Aware Cloud Computing, Jungle Computing. Docker at a Glance: Process Simplification, Broad Support and Adoption, Architecture, Getting the Most from Docker, The Docker Workflow.		
	Practical/Tutorial		
Text Books	Author	Title of Book	Publication
T1	Jack J. Dongarra, Kai Hwang, Geoffrey C. Fox	Distributed and Cloud Computing: From Parallel Processing to the Internet of Things, ,	Elsevier ISBN :9789381269237, 9381269238, 1st Edition
T2	Thomas Erl, Zaigham Mahmood and Ricardo Puttini,	Cloud Computing: Concepts, Technology & Architecture,	Pearson ISBN :978 9332535923, 9332535922, 1st Edition

Reference Books			
R1	Srinivasan, J. Suresh,	Cloud Computing: A practical approach for learning and implementation,	Pearson ISBN :9788131776513.
R2	Brian J.S. Chee and Curtis Franklin, Jr.,	Cloud Computing: Technologies and Strategies of the Ubiquitous Data Center,	CRC Press ISBN :9781439806128
R3	Kris Jamsa	, Cloud Computing: Saas, Paas, Iaas, Virtualization, Business Models, Mobile, Security, and More	Jones and Bartlett , ISBN :9789380853772
R4	John W. Rittinghouse, James F. Ransome,	Cloud Computing Implementation, Management, and Security,	CRC Press ISBN : 978 1439806807, 1439806802.
R5	Karl Matthias, Sean P. Kane,	Docker: Up and Running, ,	OREilly ISBN:9781491917572, 1491917571.
R6	RajkumarBuyya, Christian Vecchiola, S. ThamaraiSelvi,	Mastering Cloud Computing: Foundations and Applications Programming,.	McGraw Hill, ISBN: 978 1259029950, 1259029956
R7	Barrie Sosinsky,	Cloud Computing Bible	Wiley ISBN: 978 8126529803
R8	Gautham Shroff,	Enterprise Cloud Computing,	Cambridge, ISBN: 9781107648890
R9	Ronald L. Krutz and Russell D. Vines,	Cloud Security: A Comprehensive guide to Secure Cloud Computing,	Wiley, ISBN: 9788126528097
R10	Scott Adkins, John Belamaric, Vincent Giersch, Denys Makogon, Jason E. Robinson,	OpenStack: Cloud Application Development,	Wrox, ISBN :9781119194316.
R11	KailashJayaswal, JagannathKallakurchi, Donald J. Houde,	Cloud Computing Black Book ,	Wiley Dreamtech,ISBN:9789351194187
R12	Barrie Sosinsky,	Cloud Computing Bible	Wiley India, ISBN :9788126529803
Self-Learning Facilities, Web Resources, Research papers for reference			
Contents beyond Syllabus			
Bridging Courses			

Tutorials	
Presentations	

PVG'S COET, Pune

DATA SCIENCE AND BIG DATA ANALYTICS

Course Title:	Data Science And Big Data Analytics	Course Number: 314454	Credits : 4
Designation of Course	Professional Core		
Teaching Scheme: 4 Hrs/Week		Laboratories: 2 Hrs/Week	
Course Assessment Methods	Direct methods	In-Semester Examination: 30 Marks	Theory/End Semester Examination: 70 Marks
		Term-work 25 Marks	Practical/Oral 25 Marks
	Indirect Methods	Assignments,	Seminars, Quiz, Q&A session,
Prerequisites	1. Computer Organization and architecture 2. Processor Architecture and Interfacing. 3. Fundamentals of Data Structures, Data Structures and Files. 4. Theory of Computation: DFA, NFA, Regular expressions, Grammars.		
Introduction of Course			
Course Objectives			
1.	To introduce basic need of Big Data and Data science to handle huge amount of data.		
2.	To understand the basic mathematics behind the Big data.		
3.	To understand the different Big data processing technologies.		
4.	To understand and apply the Analytical concept of Big data using R and Python.		
5.	To visualize the Big Data using different tools.		
6.	To understand the application and impact of Big Data.		
Course Outcomes			
CO1	To understand Big Data primitives.		
CO2	To learn and apply different mathematical models for Big Data.		
CO3	To demonstrate their Big Data learning skills by developing industry or research applications.		
CO4	To analyze each learning model come from a different algorithmic approach and it will perform differently under different datasets		
CO5	To understand needs, challenges and techniques for big data visualization.		
CO6	To learn different programming platforms for big data analytics		
Course Contents			
Unit-I	INTRODUCTION: DATA SCIENCE AND BIG DATA		
	Introduction to Data science and Big Data, Defining Data science and Big Data, Big Data examples, Data explosion, Data volume, Data Velocity, Big data infrastructure and challenges, Big Data Processing Architectures, Data Warehouse, Re-Engineering the Data Warehouse, Shared everything and shared nothing architecture, Big data learning approaches.		
Unit-II	MATHEMATICAL FOUNDATION OF BIG DATA		
	Probability theory, Tail bounds with applications, Markov chains and random walks, Pair wise independence and universal hashing, Approximate counting, Approximate median, The streaming models, Flajolet Martin Distance sampling, Bloom filters, Local search and testing connectivity, Enforce test techniques, Random walks and testing, Boolean functions, BLR test for linearity.		
Unit-III	BIG DATA PROCESSING		
	Big Data technologies, Introduction to Google file system, Hadoop Architecture, Hadoop Storage: HDFS, Common Hadoop Shell commands, Anatomy of File Write and Read.		

	<p>NameNode, Secondary NameNode, and DataNode, HadoopMapReduce paradigm, Map Reduce tasks, Job, Task trackers - Cluster Setup – SSH &Hadoop Configuration, Introduction to: NOSQL, Textual ETL processing.</p> <p>Practical</p> <p>Hadoop Installation on a)Single Node b)Multiple Node</p> <p>Design a distributed application using MapReduce which processes a log file of a system. List out the users who have logged for maximum period on the system. Use simple log file from the Internet and process it using a pseudo distribution mode on Hadoop platform.</p> <p>Design and develop a distributed application to find the coolest/hottest year from the available weather data. Use weather data from the Internet and process it using MapReduce.</p> <p>Write an application using HBase and HiveQL for flight information system which will include</p> <ol style="list-style-type: none"> 1. Creating, Dropping, and altering Database tables 2. Creating an external Hive table to connect to the HBase for Customer Information Table 3. Load table with data, insert new values and field in the table, Join tables with Hive 4. Create index on Flight information Table 5. Find the average departure delay per day in 2008.
Unit-IV	BIG DATA ANALYTICS
	<p>Data analytics life cycle, Data cleaning , Data transformation, Comparing reporting and analysis, Types of analysis, Analytical approaches, Data analytics using R, Exploring basic features of R, Exploring R GUI, Reading data sets, Manipulating and processing data in R, Functions and packages in R, Performing graphical analysis in R, Integrating R and Hadoop, Hive, Data analytics.</p> <p>Practical</p> <p>Perform the following operations using R/Python on the Amazon book review and facebook metrics data sets</p> <ol style="list-style-type: none"> 1) Create data subsets 2) Merge Data 3) Sort Data 4) Transposing Data 5) Melting Data to long format 6) Casting data to wide format <p>Perform the following operations using R/Python on the Air quality and Heart Diseases data sets</p> <ol style="list-style-type: none"> 1) Data cleaning 2) Data integration 3) Data transformation 4) Error correcting 5) Data model building <p>Integrate R/Python and Hadoop and perform the following operations on forest fire dataset</p> <ol style="list-style-type: none"> 1) Text mining in RHadoop 2) Data analysis using the Map Reduce in Rhadoop 3) Data mining in Hive <p>Visualize the data using R/Python by plotting the graphs for assignment no. 2 and 3</p>
Unit- V	Big Data Visualization
	<p>Introduction to Data visualization, Challenges to Big data visualization, Conventional data visualization tools, Techniques for visual data representations, Types of data visualization, Visualizing Big Data, Tools used in data visualization, Propriety Data Visualization tools, Open –source data visualization tools, Analytical techniques used in Big data visualization, Data visualization with Tableau, Introduction to: Pentaho, Flare, Jasper Reports, Dygraphs, Datameer Analytics Solution and Cloudera, Platfora, NodeBox, Gephi, Google Chart API,</p>

	Flot, D3, and Visually.		
	Practical		
	Perform the following data visualization operations using Tableau on Adult and Iris datasets		
	1) 1D (Linear) Data visualization 2) 2D (Planar) Data Visualization 3) 3D (Volumetric) Data Visualization 4) Temporal Data Visualization 5) Multidimensional Data Visualization 6) Tree/ Hierarchical Data visualization 7) Network Data visualization		
Unit-VI	BIG DATA TECHNOLOGIES APPLICATION AND IMPACT		
	Social media analytics, Text mining, Mobile analytics , Roles and responsibilities of Big data person, Organizational impact, Data analytics life cycle, Data Scientist roles and responsibility, Understanding decision theory, creating big data strategy, big data value creation drivers, Michael Porter's valuation creation models, Big data user experience ramifications, Identifying big data use cases.		
	Practical		
	Part C : Case Study Assignment		
	1) Social Media Analytics 2) Text Mining/ Text Analytics 3) Mobile Analytics		
Text Books	Author	Title of Book	Publication
T1	Krish Krishnan	Data warehousing in the age of Big Data	Elsevier, ISBN: 9780124058910, 1st Edition.
T2	DT Editorial Services	Big Data, Black Book	DT Editorial Services, ISBN: 9789351197577, 2016 Edition.
Reference Books	Mitzenmacher , Upfal	Probability and Computing: Randomized Algorithms and Probabilistic Analysis,	Cambridge University press, ISBN :521835402 hardback.
R1	Dana Ron	Algorithmic and Analysis Techniques in Property Testing, School of EE.	School of EE.
R2	Mark Gardener	Beginning R: The Statistical Programming Language	WROX(Wiley)India, ISBN
	EMC Services	Data Science and Big Data Analytics	EMC Education Services.
Self-Learning Facilities	http://nptel.ac.in/courses/106106142/		
Web Resources	http://hadoop.apache.org/		
	https://www.tableau.com		
Research papers for reference	Author	Title of Paper	Journal/Transaction
	Sanjay Ghemawat, Howard Gobioff, and Shun-Tak Leung	The Google File System	SOSP'03, October 19–22, 2003, Bolton Landing, New York, USA. Copyright 2003 ACM 1-58113-757-5/03/0010 ... \$5.00.

PVG'S COET, Pune