SREE VIDYANIKETHAN ENGINEERING COLLEGE

(AUTONOMOUS)

Sree Sainath Nagar, Tirupati

Department of Information Technology

Supporting Document for 1.1.2

Syllabus Revision carried out in 2016

Program: M.Tech.- Software Engineering

Regulations : SVEC-16

This document details the following:

1. Courses where syllabus has been changed 20% and more.
2. Course-wise revised syllabus with changes highlighted.

Note: *For SVEC-16 revised syllabus, SVEC-14 (previous syllabus) is the reference.*

List of Courses where syllabus has been changed (20% and more)

Course Code	Name of the course	Percentage of Syllabus changed	Page Number in which Details are Highlighted
16MT12501	Cloud Computing	40	3
16MT12508	Software Reliability and Reuse	100	7
16MT12531	Advanced Software Engineering Lab-1	100	9
16MT12532	Cloud Computing Lab	100	11
16MT22504	Software Testing Techniques	100	13
16MT22531	Advanced Software Engineering Lab-2	100	15
16MT22532	Big Data Technologies Lab	100	17
Average		**91.43**	
Total No. of Courses in the Program		28	
No. of Courses where syllabus (more than 20%) has been changed		7	
Percentage of Syllabus changed in the Program		**22.86**	

DEAN (Academics)

Dean (Academics)
Sree Vidyanikethan Engg. College
Sree Sainath Nagar
TIRUPATI - 517 102, A.P., India.

PRINCIPAL

PRINCIPAL
SREE VIDYANIKETHAN ENGINEERING COLLEGE
(AUTONOMOUS)
Sree Sainath Nagar, A. RANGAMPET
Chittoor (Dist.) - 517 102, A.P., INDIA.

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. I Semester
(16MT12501) CLOUD COMPUTING

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
40	60	100	4	-	-	4

PREREQUISITES: Courses on "Operating Systems" and "Computer Networks"

COURSE DESCRIPTION: Virtualization, Case studies – XEN, VMware, Microsoft Hyper-V; Cloud architecture; Services and Applications; Cloud Programming; Industry practices and Case studies –Amazon Web Services, Google App Engine, and Microsoft Azure.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Demonstrate knowledge on Virtualization models, Cloud Architecture, Services and Programming concepts.
2. Analyze the problems in existing cloud architectures.
3. Apply concurrent programming, throughput computing and Data intensive computing in Cloud programming.
4. Develop research insights into emerging technologies and energy management.
5. Apply virtualization techniques to optimize resource sharing.

DETAILED SYLLABUS

UNIT-I: INTRODUCTION TO VIRTUALIZATION **(Periods: 09)**

Characteristics of Virtualized Environments, Taxonomy of Virtualization Techniques, Virtualization and Cloud Computing, Pros and Cons of Virtualization, Technology Examples – XEN, VMware, Microsoft Hyper-V.

UNIT-II: CLOUD ARCHITECTURE **(Periods: 11)**

Introduction to Cloud: Defining Cloud Computing, Cloud Types - The NIST model, The Cloud Cube Model, Deployment models, Service models, Examining the Characteristics of Cloud Computing, Paradigm shift, Benefits of cloud computing, Disadvantages of cloud computing, Assessing the Role of Open Standards.

Cloud Architecture: Exploring the Cloud Computing Stack, Composability, Infrastructure, Platforms, Virtual Appliances, Communication Protocols, and Applications.

UNIT-III: DEFINING CLOUD SERVICES **(Periods: 10)**

Defining Infrastructure as a Service (IaaS) – IaaS workloads, Pods, aggregation, and silos, **Defining Platform as a Service (PaaS), Defining Software as a Service (SaaS)** – SaaS characteristics, Open SaaS and SOA, Salesforce.com and CRM SaaS, **Defining Identity as a Service (IDaaS)** – what is an identity? Networked identity service classes, Identity system codes of conduct, IDaaS interoperability, **Defining Compliance as a Service (CaaS)**.

UNIT-IV: CLOUD PROGRAMMING CONCEPTS **(Periods: 12)**

Concurrent Programming – Introduction to Parallelism for Single Machine Computation, Programming Applications with Threads, **High Throughput Computing** – Task Programming, Task based Application Models, **Data Intensive Computing** – What is Data Intensive Computing and Technologies for Data Intensive Computing.

UNIT-V: INDUSTRIAL PLATFORMS AND TRENDING DEVELOPMENTS **(Periods: 13)**

Case Studies on Cloud Platforms – Amazon Web Services, Google App Engine, and Microsoft Azure, Case Studies on Cloud Applications – Scientific Applications, Business and Consumer Applications.

Enhancements in Cloud – Energy Efficiency in Clouds, Market based Management of Clouds, Federated Clouds / InterCloud, Third Party Cloud Services.

[Total Periods: 55]

TEXT BOOKS:

1. Rajkumar Buyya, Christian Vecchiola, S. Thamarai Selvi, "*Mastering Cloud Computing: Foundations and Applications Programming,*" 1st Edition, McGraw Hill, 2013.
2. Barrie Sosinsky, "*Cloud Computing Bible,*" 1st Edition, Wiley India Pvt Ltd, 2011.

REFERENCE BOOKS:

1. Anthony T. Velte, Toby J. Velte Robert Elsenpeter, "*Cloud Computing: A Practical Approach,*" 1st Edition, Tata Mc Graw Hill, 2010.
2. George Reese, "*Cloud Application Architectures,*" 1st Edition, O'Reilly Publishers, 2010.

M.Tech. (SE) I Semester
(14MT12508) CLOUD COMPUTING

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
40	60	100	4	-	-	4

PREREQUISITES: Courses on "Operating Systems" and "Computer Networks"

COURSE DESCRIPTION:
Virtualization, Cloud architecture, assessing the value proposition; Understanding services & applications, capacity planning; Platform as a Service case studies- Google & Amazon Services; Cloud management and their applications- Microsoft cloud services, cloud security.

COURSE OUTCOMES:
On successful completion of this course, the student will be able to:

1. Acquire knowledge on services and types of infrastructural models in Cloud.
2. Identify the Cloud Architectures in developing the Web Applications.
3. Gain problem solving skills to assess & improve quality attributes, security, privacy concerns and performance in Cloud Computing.

DETAILED SYLLABUS

UNIT-I: VIRTUALIZATION **(Periods:10)**

Introduction to Virtualization, Objectives of Virtualization, Benefits of Virtualized Technology, Adding guest Operating System. Introduction to Virtualization Technologies - Ubuntu, VMware, Microsoft Hyper-V.

UNIT-II: DEFINING CLOUD **(Periods:12)**

Defining Cloud Computing, Cloud Types - The NIST model, The Cloud Cube Model, Deployment models, Service models, Examining the Characteristics of Cloud Computing, Paradigm shift, Benefits of cloud computing, Disadvantages of cloud computing, Assessing the Role of Open Standards.

Assessing the Value Proposition: Measuring the Cloud's Value, Early adopters and new applications, The laws of cloudonomics, Cloud computing obstacles, Behavioral factors relating to cloud adoption, Measuring cloud computing costs, Avoiding Capital Expenditures, Right-sizing, Computing the Total Cost of Ownership, Specifying Service Level Agreements, Defining Licensing Models.

Understanding Cloud Architecture: Exploring the Cloud Computing Stack, Composability, Infrastructure, Platforms, Virtual Appliances, Communication Protocols, Applications, Connecting to the Cloud, The Jolicloud Netbook OS, Chromium OS: The Browser as an Operating System.

UNIT-III: UNDERSTANDING SERVICES AND APPLICATIONS BY TYPE **(Periods:12)**

Defining Infrastructure as a Service (IaaS), IaaS workloads, Pods, aggregation, and silos, Defining Platform as a Service (PaaS), Defining Software as a Service (SaaS), SaaS characteristics, Open SaaS and SOA, Salesforce.com and CRM SaaS, Defining Identity as a Service (IDaaS), what is an identity? Net-

worked identity service classes, Identity system codes of conduct, IDaaS interoperability, Defining Compliance as a Service (CaaS).

Understanding Abstraction and Virtualization: Using Virtualization Technologies, Load Balancing and Virtualization, Advanced load balancing, The Google Cloud Understanding Hypervisors, Virtual machine types, VMware vSphere, Understanding Machine Imaging, Porting Applications, The Simple Cloud API, AppZero Virtual Application Appliance.

Capacity Planning: Capacity Planning, Defining Baseline and Metrics, Baseline measurements, System metrics, Load testing, Resource ceilings, Server and instance types, Network Capacity, Scaling.

UNIT-IV: EXPLORING PLATFORM AS A SERVICE (Periods:13)

Defining Services, Salesforce.com versus Force.com, SaaS versus PaaS, Application development, Using PaaS Application Frameworks, Drupal, Eccentex AppBase 3.0, LongJump, Squarespace, WaveMaker, Wolf Frameworks.

Using Google Web Services: Exploring Google Applications, Surveying the Google Application Portfolio, Indexed search, The dark Web, Aggregation and disintermediation, Productivity applications and services, Enterprise offerings, AdWords, Google Analytics, Google Translate, Exploring the Google Toolkit, The Google APIs, Working with the Google App Engine.

Using Amazon Web Services: Understanding Amazon Web Services, Amazon Web Service Components and Services, Working with the Elastic Compute Cloud (EC2), Amazon Machine Images, Pricing models, System images and software, Creating an account and instance on EC2, Working with Amazon Storage Systems, Amazon Simple Storage System (S3), Amazon Elastic Block Store (EBS), CloudFront, Understanding Amazon Database Services, Amazon SimpleDB, Amazon Relational Database Service (RDS), Choosing a database for AWS.

UNIT-V: USING MICROSOFT CLOUD SERVICES (Periods:12)

Exploring Microsoft Cloud Services, Defining the Windows Azure Platform, The software plus services approach, The Azure Platform, The Windows Azure service, Windows Azure AppFabric, Azure Content Delivery Network, SQL Azure, Windows Azure pricing, Windows Live services, Using Windows Live, Windows Live Essentials, Windows Live Home, windows live for mobile.

Managing the Cloud: Administrating the Clouds, Management responsibilities, Lifecycle Management, Cloud Management Products, Emerging Cloud Management Standards, DMTF Cloud Management standards, Cloud Commons and SMI.

Understanding Cloud Security: Securing the Cloud, The security boundary, Security service boundary, Security mapping, Securing Data, Brokered cloud storage access, Storage location and tenancy, Encryption, Auditing and compliance, Establishing Identity and Presence, Identity protocol standards, Windows Azure identity standards.

Total Periods:59

TEXT BOOKS:

1. Barrie Sosinsky, *"Cloud Computing Bible,"* 1st Edition, Wiley India Pvt Ltd, 2011.
2. Ivanka Menken and Ivanka Menken, *"Cloud Computing Virtualization Specialist Complete Certification Kit - Study Guide Book,"* 1st Edition, Emereo Pty. Ltd., 2009.

REFERENCE BOOKS:

1. Anthony T. Velte, Toby J. Velte Robert Elsenpeter, *"Cloud Computing: A Practical Approach,"* 1st Edition, Tata Mc Graw Hill, 2010.
2. John W. Rittinghouse and James F. Ransome, *"Cloud Computing Implementation, Management and Security,"* 1st Edition, CRC Press, Taylor & Francis Group, 2010.
3. George Reese, *"Cloud Application Architectures,"* 1st Edition, O'Reilly Publishers, 2010.
4. David S. Linthicum, *"Cloud Computing and SOA Convergence in your Enterprise,"* 1st Edition, Addison- Wesley, 2010.

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. (SE) I Semester
(16MT12508) SOFTWARE RELIABILITY AND REUSE
(PROFESSIONAL ELECTIVE-I)

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
40	60	100	4	-	-	4

PREREQUISITES: Courses on "Software Engineering" and "Probability and Statistics".

COURSE DESCRIPTION: Software reliability engineering process, Software reliability strategies, availability; Software reliability modeling; Software metrics for reliability assessment; Best practice of software reliability engineering, and neural networks for software reliability, software system failures, free software intensive system and reusable components.

COURSE OUTCOMES:
On successful completion of this course, the student will be able to:

1. Gain knowledge on:
   - Software Reliability Modeling.
   - Software metrics for Reliability Assessment.
   - Software Reliability Estimation.
   - Best practices of Software Reliability Engineering.
2. Analyze software system failures and operational profile.
3. Solve Software system reliability issues using optimum reliability models.
4. Initiate research in producing failure free software intensive system.
5. Apply advanced methods to analyze complex legacy software systems and identify reusable components.

DETAILED SYLLABUS

UNIT – I: SOFTWARE RELIABILITY (Periods: 10)

Software Reliability Ideas of Software Reliability, Computation of software reliability, Classes of software reliability Models.
Time Dependent Software Reliability Models: Time between failure reliability Models, Fault Counting Reliability Models.

UNIT – II: TIME INDEPENDENT SOFTWARE RELIABILITY MODELS (Periods: 08)

Fault injection model of Software Reliability, Input Domain Reliability Model, Orthogonal defect classification, Software availability Models.
Software Reliability Modeling: A general procedure for reliability modeling.

UNIT – III: SOFTWARE REUSE (Periods: 10)

Introduction Software Reuse and Software Engineering, Concepts and Terms, Software Reuse products, Software Reuse processes, Software reuse paradigms. State of the Art and the Practice: Software Reuse Management, Software Reuse Techniques, Aspects of Software Reuse, Organizational Aspects, Technical Aspects and Economic Aspects.

UNIT –IV: PROGRAMMING PARADIGMS AND REUSABILITY **(Periods: 10)**

Usability Attributes, Representation and Modeling Paradigms, Abstraction and Composition in development paradigm.

Object - Oriented Domain Engineering: Abstraction and parameterization techniques, Composition techniques in Object Orientation.

UNIT-V: APPLICATION ENGINEERING **(Periods: 10)**

Component Storage and Retrieval, Reusable Asset Integration. Software Reuse technologies: Component Based Software Engineering, COTS based development, Software Reuse Metrics, Tools for Reusability.

[Total Periods: 48]

TEXT BOOKS:

1. Michael R. Lyu, "*Handbook of Software Reliability Engineering*," IEEE Computer Society Press, McGraw-Hill Book Company, 2005.
2. Ivar Jacobson, Martin Gress, Patrick Johnson, "*Software Reuse*," Pearson Education, 2004.

REFERENCE BOOKS:

1. John D. Musa, "*Software Reliability Engineering*," 2nd Edition, Tata Mc GrawHill, 2011.
2. Eve-Andre Karisson, "*Software Reuse – A Holistic Approach*," John Wiley and Sons, 1996.

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. (SE) I Semester
(16MT12531) ADVANCED SOFTWARE ENGINEERING LAB-1

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
50	50	100	-	-	4	2

PREREQUISITES: Courses on "Software Development Methodologies", "Software Requirements and Estimation", Data Structures and Algorithms" and "Object Oriented Analysis & Design"

COURSE DESCRIPTION: Software development life cycle activities- requirements specification using open source Requirement documentation tool, modeling using AgroUML tool; Implementation of various linear and non-linear data structures using C++; Refactoring using InsRefactor and SafeRefactor Eclipse Plugins.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Demonstrate hands-on experience on:
   - Requirements Engineering and Management
   - Estimation of software-size, effort, schedule and cost.
2. Identify key entities and relationships in the problem domain and write succinct textual descriptions of problems, modeling and Implement linear and non-linear data structures using C++.
3. Identify and apply designing, estimating tools and methodologies for complex engineering problems.
4. Apply algorithm design approaches and C++ programming skills to solve real world applications.
5. Work individually and in teams collaboratively in implementing mini projects.
6. Demonstrate communication skills both oral and written for preparing and presenting reports.
7. Engage in life-long learning and enthusiasm to upgrade knowledge and skills in latest technologies and tools.

LIST OF EXERCISES:

1. Prepare the Software Requirement Specification (SRS), High Level Design (HLD) and Detailed Design (DD) for the following experiments

   (i) Employee Information System (ii) Online Airline Reservation

   Note: For the reference of SRS, HLD and DD templates refer department manual and use any open source Requirement documentation tool.
2. Estimate project parameters such as size, effort and time for development for a Library Information system using Basic COCOMO model.
3. Model UML Use case, Sequence, Collaboration and Component diagrams for the following experiments using Argo UML tool (i) Students Marks Analyzing System (ii) Course Registration System.

4. Study and prepare a report on the following tools: (i) Raptor-Flowchart based programming tool (ii) Microsoft Visio 2010 (iii) Jenkins tool.
5. Write C++ program to implement the following data structures using a singly linked list.

   a) Stack b) Queue
6. Write C++ program to implement the operations of doubly linked list.
7. Write a C++ program to perform the following operations of BST:

   a) Node Insertion b) Node Deletion c) Key Search
8. Write C++ program to traverse the given binary tree in Pre-order, In-order and Post-order using recursion.
9. Write C++ program for the implementing BFS and DFS graph traversal techniques using queue and stack data structures.
10. Write C++ program for implementing the following search and sorting techniques.

    a) Binary search b) Fibonacci Search c) Quick Sort d) Shell Sort
11. Write C++ program to construct the Minimum Cost Spanning Tree using Kruskal's algorithm.
12. Write a C++ program to implement 0/1 Knapsack problem.
13. Mini Project on any web based application using Refactoring

    Note: Use InsRefactor and SafeRefactor Eclipse Plugins for refactoring

REFERENCE BOOKS:

1. Roger S. Pressman, "*Software Engineering, A practitioner's Approach*," 6th Edition, Tata McGraw-Hill, Edition, 2010.
2. Sommerville, "*Software Engineering*," 8th Edition, Pearson Education, 2007.
3. Rajesh Naik and Swapna Kishore, "Software Requirements and Estimation," Tata

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. I Semester
(16MT12532) CLOUD COMPUTING LAB

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
50	50	100	-	-	4	2

PREREQUISITES: Courses on "Cloud Computing" and "Operating Systems".

COURSE DESCRIPTION: Hands-on experience on creating Virtual machines on Windows and Linux platforms, Development of Service based web applications & their deployment and Mobile app development.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Gain hands-on experience on Virtualization models and Cloud Environment.
2. Analyze the given experiment and relate to existing architectures.
3. Apply API development skills in web applications for Cloud deployment.
4. Initiate research to develop novel Service based web solutions.
5. Gain knowledge on investigative approach and identify suitable Cloud platforms for SOA based problems.
6. Devise virtual environments based on virtualization techniques.
7. Develop written and oral communications in preparing and presenting reports

LIST OF EXCERCISES:

1. Create Virtual machines with given set of configuration on Hyper-V: "Ubuntu 14 LTS OS, with 2 GB RAM and 200 GB HDD". (IaaS)
2. Create Virtual machines with given set of configuration on Hyper-V: "Windows 7 OS with 4 GB RAM and 500 GB HDD". (IaaS)
3. Create Virtual machines with given set of configuration on Ubuntu OS: "Any Unix OS with 2 GB RAM and 200 GB HDD". (IaaS)
4. Create Virtual machines with given set of configuration on Ubuntu OS: "Windows 7 OS with 4 GB RAM and 500 GB HDD". (IaaS)
5. Develop a simple web application for performing Calculator operations and deploy it on cloud platform. (SaaS)
6. Develop a Design document for a web application, to perform operations based on service calls and to be deployed on cloud environment. (Design Doc)
7. Develop a web application for performing Calculator operations be selecting relevant services. Deploy it on cloud platform. (SaaS)
8. Develop a HTTPS web application with social media interfaces (Facebook / Twitter / Instagram / Google+ APIs). (SaaS)
9. Develop a mobile app on Google App Engine for uploading a resume into a website, collaborated with Drop box. The resume should be encrypted. (PaaS)

10. Develop a service call to run on Drop box resumes for picking the resumes of given skill set. (PaaS)

    i. 6 1 years of Exp in Java Development.
    ii. 10 years of experience in Automation Testing.
    iii. 15 1 years of Managerial experience with technical background.
    iv. 5 7 years of on site experience in .NET support and programming.

REFERENCE BOOKS:

1. Ivanka Menken and Ivanka Menken, *"Cloud Computing Virtualization Specialist Complete Certification Kit - Study Guide Book,"* 1st Edition, Emereo Pty. Ltd., 2009.
2. Barrie Sosinsky, *"Cloud Computing Bible,"* 1st Edition, Wiley India Pvt Ltd, 2011.

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. II Semester
(16MT22504) SOFTWARE TESTING TECHNIQUES

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
40	60	100	4	-	-	4

PREREQUISITES: A course on "Software Development Methodologies".

COURSE DESCRIPTION: Basic concepts of Software Testing; Testing Techniques – Levels of Testing; Testing Process – Test Planning; Test Metrics and Reports; Software Test Automation.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Gain knowledge on the Software Testing types and Process for different programming environments.
2. Analyze and apply the appropriate testing techniques suitable for testing the software.
3. Design and apply appropriate strategies for selecting test cases to meet requirements of the product.
4. Apply efficient modern software testing tools for automation.
5. Write test cases and perform defect reporting.

DETAILED SYLLABUS

UNIT-I: BASIC CONCEPTS OF SOFTWARE TESTING **(Periods: 12)**

Fundamentals of software testing - software verification and validation – V test model: V model for software, testing during proposal stage, testing during requirements stage, testing during test-planning phase, test during design phase, VV model, critical roles and responsibilities.

UNIT-II: TESTING TECHNIQUES **(Periods: 12)**

Levels of testing – Acceptance testing – feature based testing (special tests part – I) – Application based testing (special tests part – II)

UNIT-III: TESTING PROCESS **(Periods: 10)**

Test planning –test policy, contents, strategy, test plan, Quality plan, test plan template, guidelines, test administration and estimation, standards, building test data, test cases, scenarios, templates for test cases, test scripts, effective test cases, building test data, generation of test data, roles and responsibilities in testing life cycle, test process monitoring.

UNIT-IV: TEST METRICS AND REPORTS **(Periods: 10)**

Testing related data, defect data, efficiency data, categories of test metrics, estimated, budgeted, approved and actual, resources, effectiveness in testing, defect density, defect leakage ratio, residual defect density, test team efficiency, test case efficiency, rework, MTBF/ MTTR, test reports, status reports, integration test reports, system test reports, final test reporting, test status report, Bench marking

UNIT-V: SOFTWARE TEST AUTOMATION **(Periods: 10)**

Test Automation: Scope of Automation, Design and Architecture of automation, Process Model for Automation, challenges in automation. Load Runner, Selenium, QTP, RFT and RQM.

[Total Periods: 54]

TEXT BOOKS:

1. M. G. Limaye, *"Software Testing : Principles and Techniques and Tools,"* Tata McGraw – Hill Education, 1st Edition, 2012.
2. Srinivasan Desikan, Gopalaswamy Ramesh, *"Software Testing: Principles and Practices"*, Pearson 2012

REFERENCE BOOKS:

1. Dr. K. V. K. K. Prasad, *"Software Testing Tools,"* Dreamtech, 1st Edition, 2004.
2. Aditya P. Mathur, *"Foundations of Software Testing"*, Pearson, 2008

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. (SE) II Semester
(16MT22531) ADVANCED SOFTWARE ENGINEERING LAB-2

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
50	50	100	-	-	4	2

PREREQUISITES: Courses on "Software Development Methodology" and "Object Oriented Programming"

COURSE DESCRIPTION: Software development life cycle activities- Implementation of design patterns using enterprise architect; Creation of web service client; Implementation of Orchestration with BPEL; Test plan document; Regression testing, functional testing using QTP, RFT and Selenium; Performance testing using Load Runner, RPT and Web Performance Tool.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Gain hands-on experience on:
   - Web services
   - Design issues and patterns
2. Analyze software design problems, complex business process and identify architectural styles, patterns and appropriate service model logic to solve.
3. Gain software testing skills and practical experience by conducting software testing processes.
4. Apply various testing phases and automate testing process for the given application using Software Engineering concepts and practices to:
   i. Identify customer's needs.
   ii. Evaluate system for feasibility.
   iii. Perform economic and technical analysis.
   iv. Allocate functions to system elements.
   v. Establish schedule, constraints and estimate cost.
   vi. Create system definitions.
5. Apply QTP and RFT tools for automation testing of software process.
6. Work individually and in teams collaboratively in implementing mini projects.
7. Gain communication skills both oral and written for preparing and presenting reports.

LIST OF EXERCISES:

1. Study and prepare a report on the Enterprise Architect 12.0.
2. Write the Java program for Abstract Factory design pattern.
3. Write the Java program for Decorator design pattern.
4. Creation of web services and client applications to consume the following services:
   (i) Addition operation (ii) Finding factorial number
5. Implementation of orchestration with BPEL for authenticating user credentials.
6. Create a test plan document for a Desktop based application.
7. Write the Functional test cases for a Desktop based application.
8. Conduct Functional testing for a Desktop based application using QTP.
9. Conduct Functional testing for a Desktop based application using RFT.
10. Conduct Performance testing for a Desktop based application using Load Runner.
11. Write the Regression test cases for a Web based application.
12. Conduct Functional Testing using Selenium for a Web based application.
13. Develop a mini project for any web based application.
    Note: Use above tools in the process of developing the project

REFERENCE BOOKS:

1. James W.Cooper, "Java Design Patterns- A Tutorial," Pearson Education, 2000
2. Eric Newcomer and Greg Lomow, "Understanding SOA with Web Services," Pearson Education, 2007.
3. Ilene Burnstein, "Practical Software Testing," Springer-Verilog Internation Edition, 2003.
4. Dr. K. V. K. K.Prasad, "Software Testing Tools," Dreamtech, 1st Edition, 2004.
5. "Introduction to IBM Rational Functional Tester 6.1,"
   http://www.ibm.com/developerworks/rational/library/04/r-3228/3228.html, drafted on July 01, 2016 at 2:30 PM.
6. "Selenium Documentation," http://docs.seleniumhq.org/docs/, drafted on July 10, 2016 at 3:30 PM.

SREE VIDYANIKETHAN ENGINEERING COLLEGE (Autonomous)
Department of Information Technology

M.Tech. (SE) II Semester
(16MT22532) BIG DATA TECHNOLOGIES LAB

Int. Marks	Ext. Marks	Total Marks	L	T	P	C
50	50	100	-	-	4	2

PREREQUISITES: Courses on "Database Management Systems", "Data Mining and Data Warehousing", and "Big Data Technologies".

COURSE DESCRIPTION: Hands on Java Programs; Data-parallel programming model-Hadoop, Hadoop I/O; MapReduce features, HDFS; Hive, HBase, Zookeeper; Sqoop and Case studies.

COURSE OUTCOMES:

On successful completion of this course, the students will be able to:

1. Gain hands-on experience on:
   - Map-Reduce, Sqoop, HBase and Mahout
2. Apply Big Data Technologies to solve contemporary problems
3. Gain skills on HDFS and MapReduce programming model concepts.
4. Initiate research insights into latest technologies of Big Data Analytics.
5. Apply various Big Data tools: Sqoop, HBase, MapReduce and Mahout.
6. Work individually and in teams collaboratively in implementing mini projects.
7. Demonstrate communication skills both oral and written for preparing and presenting reports.

LIST OF EXERCISES:

1. Practice Hadoop installation, Hadoop Cluster Configuration, Name node Safe mode, Secondary Name node, Hadoop File system Shell, Java API ,Configuration, Reading Data and Writing Data in Hadoop Distributed File System.
2. Practice on MapReduce to Implement first MapReduce Job, Running MapReduce Locally, Running MapReduce on Cluster, Packaging MapReduce Jobs, MapReduce CLASSPATH, Submitting Jobs, Logs and Web UI, Input and Output Formats, Implement a Streaming Job, Contrast with Java Code and Create counts in Streaming application.
3. Practice on Pig Programming with Execution Modes, Installation, Pig Latin Basics, Developing Pig Script: Most Occurred Start Letter, Resources, Joining data-sets and User Defined Functions (UDF).
4. Practice on Hive Installation, Table Creation and Deletion, Loading Data into Hive, Partitioning, Bucketing and Joins.
5. Practice on Sqoop with Importing and Exporting data from using RDBMS.
6. Practice on HBase Management Console, HBase Shell: Define Schema and Create, Read, Update and Delete, create via Put method, Read via Get method, Update via Put method, Delete via Delete method, Create Table, Drop Table, Scan API, Scan Caching, Scan Batching and Filters.

7. Practice on Oozie Installation, Write Oozie Workflow, Deploy and Run Oozie Workflow.

8. Practice on Flume properties, Flume sinks and Flume Channels

9. Practice on Zookeeper with Stand alone operation, Managing zookeeper storage and Programming to Zookeeper

10. **Case Study 1: Insurance Domain:** A US-based insurance provider has decided to launch a new medical insurance program targeting various customers. To help this customer understand the current realities and the market better, perform a series of data analytics tasks using Hadoop. The customer has provided pointers to the data set that can be used. For the Insurance company data set, perform the following analysis tasks
    i) Find maximum insurance
    ii) Find minimum insurance
    iii) Find average insurance
    iv) Find Total insurance

11. **Case Study 2: Retail Domain:** An Indian-based online retailer wants to launch a new product category and wants to understand the potential growth areas and areas that have stagnated over a period of time. It wants to use this information to ensure its product focus is aligned to opportunities that will grow over the next 5–7 years. The customer has also provided pointers to the data set that can used to:
    i) Find maximum retail
    ii) Find minimum retail
    iii) Find average retail
    iv) Find Total retail

12. **Case Study 3: Education Domain:** The company has recently bagged a large assignment from a US-based customer that is into training and development. The larger outcome deals with launching a suite of educational and skill development programs to consumers across the globe. As part of the project, the customer wants the company to analyze a series of data sets to arrive at a prudent product mix, product positioning, and marketing strategy that will be applicable for at least a decade.

    i) Find maximum students
    ii) Find minimum students
    iii) Find average students
    iv) Find Total students

REFERENCE BOOKS:

1. Tom White, "*Hadoop: The Definitive Guide*," Oreilly and Yahoo Press, 3rd Edition, 2012.
2. Frank J. Ohlhorst, "*Big Data Analytics: Turning Big Data into Big Money*," Wiley Publication, December 2012.
3. Kevin Roebuck, "*Big Data: High-Impact Strategies - What You Need to Know: Definitions, Adoptions, Impact, Benefits, Maturity, Vendors*," Tebbo Publisher, 2011.
4. Alex Holmes, "*Hadoop in Practice*," Manning Publications Publisher, 2012.