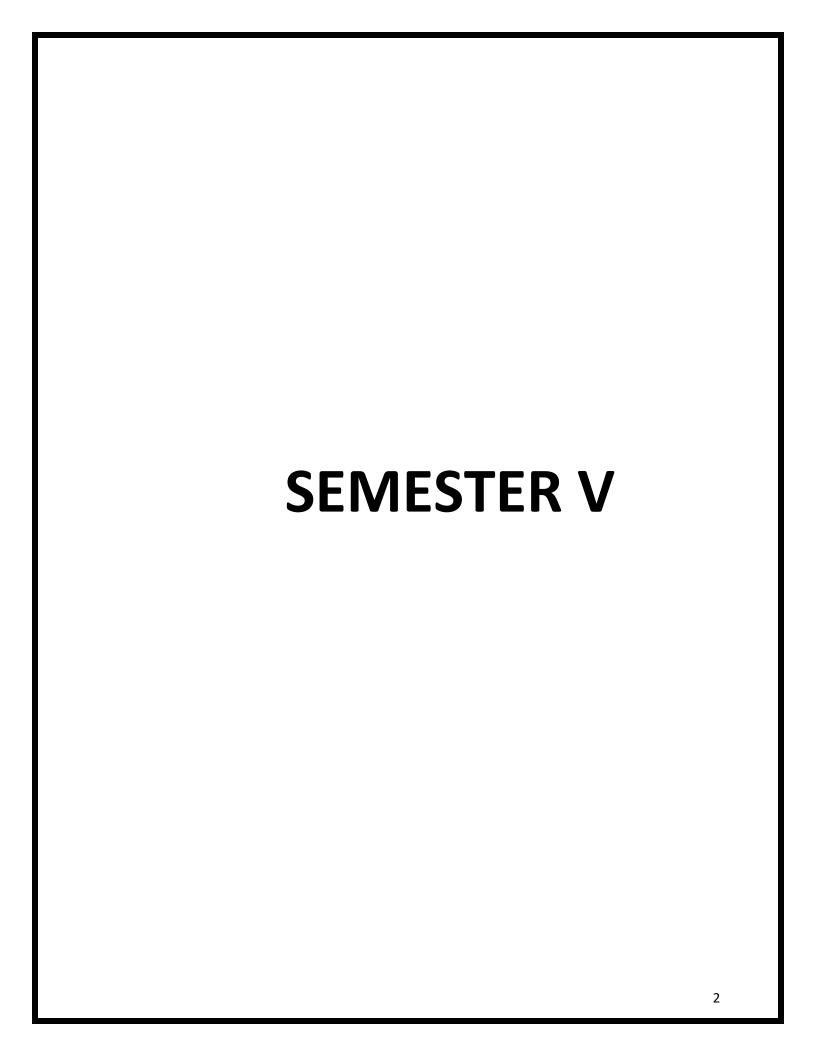# UNIVERSITY OF MUMBAI
SOPHIA COLLEGE (AUTONOMOUS)

Syllabus for T.Y.B.Sc.

Programme: Bachelor of Science

Course: Information Technology

with effect from the academic year
2022 – 2023

SEMESTER V

SEMESTER V		
Course Code	**Course Title**	**Credits**
SBTTEC501	Software Project Management	2
SBTTEC502	Internet of Things	2
SBTTEC503	Advanced Web Programming	2
SBTTEC504	Linux System Administration	2
SBTTEC505	Enterprise Java	2
SBTTECP501	Software Project Management Practical	2
SBTTECP502	Internet of Things Practical	2
SBTTECP503	Advanced Web Programming Practical	2
SBTTECP504	Linux System Administration Practical	2
SBTTECP505	Enterprise Java Practical	2
Total Credits		**20**

B. Sc. (Information Technology)	Semester – V
Course Name: Software Project Management	Course Code: SBTTEC501

COURSE OBJECTIVE

1. To provide idea of decomposing the given problem into Analysis, Design, Implementation, Testing and Maintenance phases.
2. To provide an idea of using various process models in the software industry according to given circumstances.
3. To learn the various aspects of cost considered by the organization while handling multiple projects
4. To Describe the various methodologies used in testing and to brief about the various cost models.
5. To emphasize the importance of quality assurance and brief with the reasons for which the projects fail.

COURSE OUTCOMES:

1. Students will be able to decompose the given project in various phases of a lifecycle.
2. Students will be able to choose appropriate process models depending on the user requirements.
3. What is net profit and IRR? Explain by giving an example.
4. Compare static and dynamic testing.
5. Describe metrics for measuring software quality

Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2½	75
	Internal	--	25

Unit	Details	Lectures

I	**Introduction:** What is software engineering? Software Development Life Cycle, Requirements Analysis, Software Design, Coding, Testing, Maintenance etc.
	Software Requirements: Requirement's elicitation and analysis, Requirements Validations, Requirements Management.
	Functional and Non-functional requirements, User Requirements, System Requirements, Interface Specification, Documentation of the software requirements.
	Software Development Process Models.
	    • Waterfall Model.
	    • Prototyping.
	    • Spiral Model
	    • Iterative Development.
	    • Rational Unified Process.
	    • The RAD Model
	    • Atern/Dynamic Systems Development Method
	    • Scrum
	    • Lean Software Development
			12

II	**Introduction to Software Project Management:** Introduction, why is Software Project Management Important? What is a Project? Software Projects versus Other Types of Projects, Contract Management and Technical Project Management, Activities Covered by Software Project Management, Plans, Methods and Methodologies, Some Ways of Categorizing Software Projects, Project Charter, Stakeholders, Setting Objectives, The Business Case, Project Success and Failure, what is Management? Management Control, Project Management Life Cycle, Traditional versus Modern Project Management Practices. **Project Evaluation and Programme Management: Introduction,** Feasibility study, Business Case, Project Portfolio Management, Evaluation of Individual Projects, Cost–benefit Evaluation Techniques, Risk Evaluation, Programme Management, Managing the Allocation of Resources		
		within Programmes, Strategic Programme Management, Creating a Programme, Aids to Programme Management, Some Reservations about Programme Management, Benefits Management**.**	**12**

III	**Activity Planning:** Introduction, Objectives of Activity Planning, When to Plan, Project Schedules, Projects and Activities, Sequencing and Scheduling Activities, Network Planning Models, formulating a Network Model, Adding the Time Dimension, The Forward Pass, Backward Pass, Identifying the Critical Path, Activity Float, Shortening the Project Duration, Identifying Critical Activities, Activity-on-Arrow Networks**.** **Resource Allocation:** Introduction, Nature of Resources, Identifying Resource Requirements, Scheduling Resources, Creating Critical Paths, Counting the Cost, Being Specific, Publishing the Resource Schedule, Cost Schedules, Scheduling Sequence **Risk Management:** Introduction, Risk, Categories of Risk, Risk Management Approaches, A Framework for Dealing with Risk, Risk Identification, Risk Assessment, Risk Planning, Risk Management, Evaluating Risks to the Schedule, Boehm's Top 10 Risks and Counter Measures, Applying the PERT Technique, Monte Carlo Simulation, Critical Chain Concepts	**12**
IV	**Verification and Validation:** Planning Verification and Validation, Software Inspections, Automated Static Analysis, Verification and Formal Methods. **Software Testing:** System Testing, Component Testing, Test Case Design, Test Automation. **Software Measurement:** Size-Oriented Metrics, Function-Oriented Metrics, Extended Function Point Metrics **Software Cost Estimation**: Software Productivity, Estimation Techniques, Algorithmic Cost Modelling, Project Duration and Staffing	**12**
V	**Monitoring and Control:** Introduction, Creating the Framework, Collecting the Data, Review, Visualizing Progress, Cost Monitoring, Earned Value Analysis, Prioritizing Monitoring, Getting the Project Back to Target, Change Control, Software Configuration Management (SCM). **Quality Management:** Process and Product Quality, Quality assurance and Standards, Quality Planning, Quality Control, Software Measurement and Metrics **Project Closeout:** Introduction, Reasons for Project Closure, Project Closure Process, Performing a Financial Closure, Project Closeout Report	**12**

Reference Books:

Sr. No.	Title	Author/s	Publisher	Edition
1.	Software Engineering, edition,	Ian Somerville	Pearson Education.	Ninth
2.	**Software Project Management**	**Bob Hughes, Mike Cotterell, Rajib Mall**	**TMH**	**6th**

B. Sc. (Information Technology)	Semester – V
Course Name: Internet of Things	**Course Code: SBTTEC502**

COURSE OBJECTIVE

1. student will be able to DESCRIBE THE BASIC EQUATION OF IOT AND DEFINE ITS RELATION WITH UBIQUITOUS COMPUTING
2. STUDENT WILL BE ABLE TO RECOGNIZE THE CHARACTERISTICS OF DIFFERENT BOARD USED IN PROTOTYPING.
3. STUDENT WILL BE ABLE TO DESCRIBE AND DIFFERENTIATE BETWEEN THE VARIOUS METHODS USED FOR PHYSICAL DESIGN PROTOTYPING AND SOFTWARE PROTOTYPING
4. THE STUDENT WILL BE ABLE TO IDENTIFY THE PURPOSE OF BUSINESS MODEL CANVAS AND WOULD BE ABLE TO PROPOSE ONE.
5. DISCOVER THE IMPORTANCE OF SECURITY WHILE PROTOTYPING.AND WILL BE ABLE TO IDENTIFY VARIOUS FUNDING METHODS FOR SCALING UP THE PRODUCTION OF PROTOTYPE.

COURSE OUTCOMES
1. STATE EXAMPLES IOT DEVICE SHOW EACH COMPONENT OF EQUATION IN THE STATED EXAMPLE-COMPREHENSION APPLICATION
2. DIFFERENTIATE BETWEEN ARDUINO RASBERRYPI AND BEAGLEBONE BLACK
3. EXPLAIN THE METHODS OF SOFTWARE PROTOTYPING WITH ITS ADVANTAGES
4. CONSTRUCT A BUSINESS MODEL CANVAS TAKING A SUITABLE EXAMPLE
5. EXPLAIN THE FACTORS ON DECIDING WHICH FUNDING METHODOLOGY TO USE FOR MASS PRODUCTION

Periods per week (1 Period is 50 minutes)	**5**
Credits	**2**

		Hours	Marks
	Theory Examination	**2½**	**75**
Evaluation System	**Internal**	**--**	**25**

Unit	Details	Lectures

I	**The Internet of Things: An Overview** : The Flavour of the Internet of Things, The "Internet" of "Things", The Technology of the Internet of Things, Enchanted Objects, Who is Making the Internet of Things? **Design Principles for Connected Devices:** Calm and Ambient Technology, Magic as Metaphor, Privacy, Keeping Secrets, Whose Data Is It Anyway? Web Thinking for Connected Devices, Small Pieces, Loosely Joined, First-Class Citizens On The Internet, Graceful Degradation, Affordances. **Internet Principles:** Internet Communications: An Overview, IP, TCP, The IP Protocol Suite (TCP/IP), UDP, IP Addresses, DNS, Static IP Address Assignment, Dynamic IP Address Assignment, IPv6, MAC Addresses, TCP and UDP Ports, An Example: HTTP Ports, Other Common Ports, Application Layer Protocols, HTTP, HTTPS: Encrypted HTTP, Other Application Layer Protocols.	**12**	

II	**Thinking About Prototyping:** Sketching, Familiarity, Costs versus Ease of Prototyping, Prototypes and Production, Changing Embedded Platform, Physical Prototypes and Mass Personalisation, Climbing into the Cloud, Open Source versus Closed Source, Why Closed? Why Open? Mixing Open and Closed Source, Closed Source for Mass Market Projects, Tapping into the Community. **Prototyping Embedded Devices:** Electronics, Sensors, Actuators, Scaling Up the Electronics, Embedded Computing Basics, Microcontrollers, System-on-Chips, Choosing Your Platform, Arduino, Developing on the Arduino, Some Notes on the Hardware, Openness, Raspberry Pi, Cases and Extension Boards, Developing on the Raspberry Pi, Some Notes on the Hardware, Openness.	**12**	
III	**Prototyping the Physical Design:** Preparation, Sketch, Iterate, and Explore, Nondigital Methods, Laser Cutting, Choosing a Laser Cutter, Software, Hinges and Joints, 3D Printing, Types of 3D Printing, Software, CNC Milling, Repurposing/Recycling. **Prototyping Online Components:** Getting Started with an API, Mashing Up APIs, Scraping, Legalities, Writing a New API, Clockodillo, Security, Implementing the API, Using Curl to Test, Going Further, Real-Time Reactions, Polling, Comet, Other Protocols, MQ Telemetry Transport,	**12**	
	Extensible Messaging and Presence Protocol, Constrained Application Protocol.		

IV	**Techniques for Writing Embedded Code:** Memory Management, Types of Memory, Making the Most of Your RAM, Performance and Battery Life, Libraries, Debugging, **Business Models:** A Short History of Business Models, Space and Time, From Craft to Mass Production, The Long Tail of the Internet, Learning from History, The Business Model Canvas, Who Is the Business Model For? Models, Make Thing, Sell Thing, Subscriptions, Customisation, Be a Key Resource, Provide Infrastructure: Sensor Networks, Take a Percentage, Funding an Internet of Things Startup, Hobby Projects and Open Source, Venture Capital, Government Funding, Crowdfunding, Lean Startups.		**12**
V	**Moving to Manufacture:** What Are You Producing? Designing Kits, Designing Printed circuit boards, Software Choices, The Design Process, Manufacturing Printed Circuit Boards, Etching Boards, Milling Boards. Assembly, Testing, Mass-Producing the Case and Other Fixtures, Certification, Costs, Scaling Up Software, Deployment, Correctness and Maintainability, Security, Performance, User Community. **Ethics:** Characterizing the Internet of Things, Privacy, Control, Disrupting Control, Crowdsourcing, Environment, Physical Thing, Electronics, Internet Service, Solutions, The Internet of Things as Part of the Solution, Cautious Optimism, The Open Internet of Things Definition.		**12**

Books and References:					
Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Designing the Internet of Things	Adrian McEwen, Hakim Cassimally	WILEY	First	2014
2.	Internet of Things – Architecture and Design	Raj Kamal	McGraw Hill	First	2017
3.	Getting Started with the Internet of Things	Cuno Pfister	O'Reilly	Sixth	2018
4.	Getting Started with Raspberry Pi	Matt Richardson and Shawn Wallace	SPD	Third	2016

B. Sc. (Information Technology)	Semester – V
Course Name: Advanced Web Programming	Course Code: SBTTEC503
Periods per week (1 Period is 50 minutes)	5
Credits	2

Course Objective:
1. To explore .NET technologies for designing and developing dynamic, interactive and responsive web applications.
2. Provide a consistent, object-oriented programming environment whether object code is stored and executed locally, executed locally but web distributed, or executed remotely.
3. Make the developer experience consistent across widely varying types of apps, such as Windows based apps and Web-based apps.
4. Proficiently develop ASP.NET web applications using C#.
5. Building multi-tier enterprise applications.

Course Outcome:
Through the study of this course, students will gain a comprehensive understanding on the concepts and function of .NET Framework and ASP.NET Technology.
Students will be able to:

1. Understand the .NET framework.
2. Develop a proficiency in the C# programming language.
3. Proficiently develop ASP.NET web applications using C#.
4. Use ADO.NET for data persistence in a web application.
5. To understand the 3-tier software architecture (presentation/client tier, application tier, data tier) and develop multi-tier applications to understand and experiment with the deployment of enterprise applications.

		Hours	Marks
Evaluation System	Theory Examination	2½	75
	Internal	--	25

Unit	Details	Lectures

I	**Introducing .NET**: The .NET Framework, C#, VB, and the .NET Languages, The Common Language Runtime, The .NET Class Library. **The C# Language**: C# Language Basics, Variables and Data Types, Variable Operations, Object-Based Manipulation, Conditional Logic, Loops, Methods. **Types, Objects, and Namespaces**: The Basics About Classes, Building a Basic Class, Value Types and Reference Types, Understanding Namespaces and Assemblies, Advanced Class Programming.	**12**
II	**Web Form Fundamentals**: Writing Code, Using the Code-Behind Class, Adding Event Handlers, Understanding the Anatomy of an ASP.NET Application, Introducing Server Controls, Using the Page Class, Using Application Events, Configuring an ASP.NET Application. **Form Controls**: Stepping Up to Web Controls, Web Control Classes, List Controls, Table Controls, Web Control Events and AutoPostBack, Validation, Understanding Validation, Using the Validation Controls, Rich Controls, The Calendar, The AdRotator, Pages with Multiple Views, User Controls and Graphics, User Controls, Dynamic Graphics, The Chart Control, Website Navigation: Site Maps, URL Mapping and Routing, The SiteMapPath Control, The TreeView Control, The Menu Control.	**12**
III	**Error Handling, Logging, and Tracing** : Avoiding Common Errors, Understanding Exception Handling, Handling Exceptions, Throwing Your Own Exceptions, Using Page Tracing **State Management** : Understanding the Problem of State, Using View State, Transferring Information Between Pages, Using Cookies, Managing Session State, Configuring Session State, Using Application State, Comparing State Management Options **Styles, Themes, and Master Pages** : Styles, Themes, Master Page Basics, Advanced Master Pages,	**12**
IV	**ADO.NET Fundamentals**: Understanding Databases, Configuring Your Database, Understanding SQL Basics, Understanding the Data Provider Model, Using Direct Data Access, Using Disconnected Data Access. **Data Binding**: Introducing Data Binding, Using Single-Value Data Binding, Using Repeated-Value Data Binding, Working with Data Source Controls,	**12**
	The Data Controls: The GridView, Formatting the GridView, Selecting a GridView Row, Editing with the GridView, Sorting and Paging the GridView, Using GridView Templates, The DetailsView and FormView	

V	**XML**: XML Explained, The XML Classes, XML Validation, XML Display and Transforms. **Security Fundamentals**: Understanding Security Requirements, Authentication and Authorization, Forms Authentication, Windows Authentication. **ASP.NET AJAX**: Understanding Ajax, Using Partial Refreshes, Using Progress Notification, Implementing Timed Refreshes, Working with the ASP.NET AJAX Control Toolkit.	**12**

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Beginning ASP.NET 4.5 in C#	Matthew MacDonald	Apress		2012
2.	C# 2015	Anne Bohem and Joel Murach	Murach	Third	2016
3.	Murach's ASP.NET 4.6 Web Programming in C#2015	Mary Delamater and Anne Bohem	SPD	Sixth	2016
4.	ASP.NET 4.0 programming	J. Kanjilal	Tata McGraw-Hill		2011
5.	Programming ASP.NET	D.Esposito	Microsoft Press (Dreamtech)		2011
6.	Beginning Visual C# 2010	K. Watson, C. Nagel, J.H Padderson, J.D. Reid, M.Skinner	Wrox (Wiley)		2010

B. Sc. (Information Technology)		Semester – V	
Course Name: Linux System Administration		**Course Code: SBTTEC504**	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
Course objectives:		**Hours**	**Marks**
1. The course objective is to analyze fundamentals of the Linux operating system, including installation, configuration, administration, file management, and security. 2. Students will learn to operate running Linux systems by managing the boot process, scheduling jobs, updating the system, monitoring system performance and managing security.			
Course Outcome: 1. To create basic filesystem and manage system administration task 2. To create partitions and configure the network manager 3. To manage users, groups and permissions. 4. To setup firewalls and cryptographic services. 5. To configure DNS, mail & Apache server 6. To Design a high availability cluster 7. To setup an installation Server			
Evaluation System	**Theory Examination**	2½	75
	Internal	--	25

Unit	Details	Lectures
I	**Introduction to Red Hat Enterprise Linux:** Linux, Open Source and Red Hat, Origins of Linux, Distributions, Duties of Linux System Administrator. **Command Line:** Working with the Bash Shell, Getting the Best of Bash, Useful Bash Key Sequences, Working with Bash History, Performing Basic File System Management Tasks, Working with Directories, Piping and Redirection, Finding Files **System Administration Tasks:** Performing Job Management Tasks, System and Process Monitoring and Management, Managing Processes with ps, Sending Signals to Processes with the kill Command, using top to Show Current System Activity, Managing Process Niceness, Scheduling Jobs, Mounting Devices, Working with Links, Creating Backups, Managing Printers, Setting Up System Logging, Setting Up Rsyslog, Common Log Files, Setting Up Logrotate **Managing Software:** Understanding RPM, Understanding Meta Package Handlers, Creating Your Own Repositories, Managing Repositories, Installing Software with Yum, Querying Software, Extracting Files from RPM Packages	12

II	**Configuring and Managing Storage:** Understanding Partitions and Logical Volumes, Creating Partitions, Creating File Systems, File Systems Overview, Creating File Systems, Changing File System Properties, Checking the File System Integrity, Mounting File Systems Automatically Through fstab, Working with Logical Volumes, Creating Logical Volumes, Resizing Logical Volumes, Working with Snapshots, Replacing Failing Storage Devices, Creating Swap Space, Working with Encrypted Volumes **Connecting to the Network:** Understanding NetworkManager, Working with Services and Runlevels, Configuring the Network with NetworkManager, Working with system-config-network, NetworkManager Configuration Files, Network Service Scripts, Networking from the Command Line, Troubleshooting Networking, Setting Up IPv6, Configuring SSH, Enabling the SSH Server, Using the SSH Client, Using PuTTY on Windows Machines, Configuring KeyBased SSH Authentication, Using Graphical Applications with SSH, Using SSH Port Forwarding, Configuring VNC Server Access	**12**
	Working with Users, Groups, and Permissions: Managing Users and Groups, Commands for User Management, Managing Passwords, Modifying and Deleting User Accounts, Configuration Files, Creating Groups, Using Graphical Tools for User, and Group Management, Using External Authentication Sources, the Authentication Process, sssd, nsswitch, Pluggable Authentication Modules, Managing Permissions, the Role of Ownership, Basic Permissions: Read, Write, and Execute, Advanced Permissions, Working with Access Control Lists, Setting Default Permissions with umask, Working with Attributes	

III	**Securing Server with iptables:** Understanding Firewalls, Setting Up a Firewall with system-config-firewall, Allowing Services, Trusted Interfaces, Masquerading, Configuration Files, Setting Up a Firewall with iptables, Tables, Chains, and Rules, Composition of Rule, Configuration Example, Advanced iptables Configuration, Configuring Logging, The Limit Module, Configuring NAT **Setting Up Cryptographic Services:** Introducing SSL, Proof of Authenticity: the Certificate Authority, Managing Certificates with openssl, Creating a Signing Request, Working with GNU Privacy Guard, Creating GPG Keys, Key Transfer, Managing GPG Keys, Encrypting Files with GPG, GPG Signing, Signing RPM Files **Configuring Server for File Sharing:** What is NFS? Advantages and Disadvantages of NFS, Configuring NFS4, Setting Up NFSv4, Mounting an NFS Share, Making NFS Mounts Persistent, Configuring Automount, Configuring Samba, Setting Up a Samba File Server, Samba Advanced Authentication Options, Accessing Samba Shares, Offering FTP Services.	**12**

IV	**Configuring DNS and DHCP:** Introduction to DNS, The DNS Hierarchy, DNS Server Types, The DNS Lookup Process, DNS Zone Types, Setting Up a DNS Server, Setting Up a Cache-Only Name Server, Setting Up a Primary Name Server, Setting Up a Secondary Name Server, Understanding DHCP, Setting Up a DHCP Server **Setting Up a Mail Server:** Using the Message Transfer Agent, the Mail Delivery Agent, the Mail User Agent, Setting Up Postfix as an SMTP Server, Working with Mutt, Basic Configuration, Internet Configuration, Configuring Dovecot for POP and IMAP **Configuring Apache on Red Hat Enterprise Linux:** Configuring the Apache Web Server, Creating a Basic Website, Understanding the Apache Configuration Files, Apache Log Files, Working with Virtual Hosts, Securing the Web Server with TLS Certificates, Configuring Authentication, Setting Up Authentication with .htpasswd, Configuring LDAP Authentication, Setting Up MySQL	**12**	
V	**Introducing Bash Shell Scripting:** Introduction, Elements of a Good Shell Script, Executing the Script, Working with Variables and Input, Understanding Variables, Variables, Subshells, and Sourcing, Working with Script Arguments, Asking for Input, Using Command Substitution, Substitution Operators, Changing Variable Content with Pattern Matching, Performing Calculations, Using Control Structures, Using if...then...else, Using case, Using while, Using until, Using for, Configuring booting with GRUB. **High-Availability Clustering:** High-Availability Clustering, The Workings of High Availability, High-Availability Requirements, Red Hat High-Availability Add-on Software, Components, Configuring Cluster-Based Services, Setting Up Bonding, Setting Up Shared Storage, Installing the Red Hat High Availability Add-On, Building the Initial State of the Cluster, Configuring Additional Cluster Properties, Configuring a Quorum Disk, Setting Up Fencing, Creating Resources and Services, Troubleshooting a Nonoperational Cluster, Configuring GFS2 File Systems **Setting Up an Installation Server:** Configuring a Network Server as an Installation Server, Setting Up a TFTP and DHCP Server for PXE Boot, Installing the TFTP Server, Configuring DHCP for PXE Boot, Creating the TFTP PXE Server Content, Creating a Kickstart File, Using a Kickstart File to Perform an Automated, Installation, Modifying the Kickstart File with, system-config-kickstart, Making Manual Modifications to the Kickstart File	**12**	

Books and References:					
Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Red Hat Enterprise Linux 6 Administration	Sander van Vugt	John Wiley and Sons		2013
2.	Red hat Linux Networking and System Administration	Terry Collings and Kurt Wall	Wiley	3rd	

| 3. | Linux Administration: A Beginner's Guide | Wale Soyinka | TMH | Fifth Edition | |

B. Sc. (Information Technology)	Semester – V
Course Name: Enterprise Java	Course Code: SBTTEC505

Course objectives:

1. To gain knowledge about Enterprise Application, Java EE Technologies, Java EE Architecture, Server and Containers, Servlet Technology
2. To learn the concept of Request Dispatcher, Cookies, Sessions, Non-Blocking I/O
3. To learn the concept of Java Server Page, JSP Action Elements and Implicit Objects, JSTL.
4. To learn the concept of Enterprise JavaBeans, Session Beans, Message Driven Beans, Java Naming and Directory Interface
5. To learn the concept Persistence, Object/Relational Mapping And JPA, Hibernate, Writing Hibernate Application.

Course Outcomes:

1. Create dynamic web pages using Servlets, Databases application with servlet.
2. Apply the concepts of Request Dispatcher, Cookies, Session, Working with Non-Blocking I/O with servlet.
3. Create dynamic web pages using Java Server Pages.
4. Make a reusable software component, using Java Bean.
5. Map Java classes and object associations to relational database tables with Hibernate mapping files.

Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	**Marks**
Evaluation System	**Theory Examination**	2½	75
	Internal	--	25

Unit	Details	Lectures
I	**Understanding Java EE:** What is an Enterprise Application? What is java enterprise edition? Java EE Technologies, Java EE evolution, Glassfish server **Java EE Architecture, Server and Containers:** Types of System Architecture, Java EE Server, Java EE Containers. **Introduction to Java Servlets:** The Need for Dynamic Content, Java Servlet Technology, Why Servlets? What can Servlets do? **Servlet API and Lifecycle:** Java Servlet API, The Servlet Skeleton, The Servlet Life Cycle, A Simple Welcome Servlet **Working With Servlets:** Getting Started, Using Annotations Instead of Deployment Descriptor. **Working with Databases:** What Is JDBC? JDBC Architecture, Accessing Database, The Servlet GUI and Database Example.	12

II	**Request Dispatcher:** Resquestdispatcher Interface, Methods of Requestdispatcher, Requestdispatcher Application. **COOKIES:** Kinds Of Cookies, Where Cookies Are Used? Creating Cookies Using Servlet, Dynamically Changing The Colors Of A Page **SESSION:** What Are Sessions? Lifecycle Of Http Session, Session Tracking With Servlet API, A Servlet Session Example **Working With Files:** Uploading Files, Creating an Upload File Application, Downloading Files, Creating a Download File Application. **Working With Non-Blocking I/O:** Creating a Non-Blocking Read Application, Creating The Web Application, Creating Java Class, Creating Servlets, Retrieving The File, Creating index.jsp	**12**	
III	**Introduction To Java Server Pages:** Why use Java Server Pages? Disadvantages Of JSP, JSP v\s Servlets, Life Cycle of a JSP Page, How does a JSP function? How does JSP execute? About Java Server Pages **Getting Started With Java Server Pages:** Comments, JSP Document, JSP Elements, JSP GUI Example. **Action Elements:** Including other Files, Forwarding JSP Page to Another Page, Passing Parameters for other Actions, Loading a Javabean. **Implicit Objects, Scope And El Expressions:** Implicit Objects, Character Quoting Conventions, Unified Expression Language [Unified El], Expression Language.	**12**	
	Java Server Pages Standard Tag Libraries: What is wrong in using JSP Scriptlet Tags? How JSTL Fixes JSP Scriptlet's Shortcomings? Disadvantages Of JSTL, Tag Libraries.		
IV	**Introduction To Enterprise Javabeans:** Enterprise Bean Architecture, Benefits of Enterprise Bean, Types of Enterprise Bean, Accessing Enterprise Beans, Enterprise Bean Application, Packaging Enterprise Beans **Working With Session Beans:** When to use Session Beans? Types of Session Beans, Remote and Local Interfaces, Accessing Interfaces, Lifecycle of Enterprise Beans, Packaging Enterprise Beans, Example of Stateful Session Bean, Example of Stateless Session Bean, Example of Singleton Session Beans. **Working with Message Driven Beans:** Lifecycle of a Message Driven Bean, Uses of Message Driven Beans, The Message Driven Beans Example. **Interceptors:** Request And Interceptor, Defining An Interceptor, AroundInvoke Method, Applying Interceptor, Adding An Interceptor To An Enterprise Bean, Build and Run the Web Application. **Java Naming and Directory Interface**: What is Naming Service? What is Directory Service? What is Java Naming and Directory interface? Basic Lookup, JNDI Namespace in Java EE, Resources and JNDI, Datasource Resource Definition in Java EE.	**12**	

V	**Persistence, Object/Relational Mapping And JPA:** What is Persistence? Persistence in Java, Current Persistence Standards in Java, Why another Persistence Standards? Object/Relational Mapping, **Introduction to Java Persistence API:** The Java Persistence API, JPA, ORM, Database and the Application, Architecture of JPA, How JPA Works? JPA Specifications. **Writing JPA Application:** Application Requirement Specifications, Software Requirements, The Application Development Approach, Creating Database And Tables in Mysql, Creating a Web Application, Adding the Required Library Files, Creating a Javabean Class, Creating Persistence Unit [Persistence.Xml], Creating JSPS, The JPA Application Structure, Running The JPA Application. **Introduction to Hibernate:** What is Hibernate? Why Hibernate? Hibernate, Database and The Application, Components of Hibernate, Architecture of Hibernate, How Hibernate Works? **Writing Hibernate Application:** Application Requirement Specifications, Software Requirements, The Application Development Approach, Creating Database and Tables in Mysql, Creating a Web Application, Adding The Required Library Files, Creating a Javabean Class, Creating Hibernate Configuration File, Adding a Mapping Class, Creating JSPS, Running The Hibernate Application.	**12**

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Java EE 7 For Beginners	Sharanam Shah, Vaishali Shah	SPD	First	2017
2.	Java EE 8 Cookbook: Build reliable applications with the most robust and mature technology for enterprise development	Elder Moraes	Packt	First	2018
3.	Advanced Java Programming	Uttam Kumar Roy	Oxford Press		2015

B. Sc. (Information Technology)		Semester – V	
Course Name: Software Project Management Practical		Course Code: SBTTECP501	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2½	50
	Internal	--	--

	List of Practical (To be executed using Star UML or any similar software)
1.	Study and implementation sof Entity Relationship diagrams.
2.	Study and implementation of Use Case Diagrams.
3.	Study and implementation of class Diagrams.
4.	Study and implementation of Sequence Diagrams.
5.	Study and implementation of State Transition Diagrams.
6.	Study and implementation of Data Flow Diagrams.
7.	Study and implementation of Collaboration Diagrams.
8.	Study and implementation of Activity Diagrams.
9.	Study and implementation of Component Diagrams.
10.	Study and implementation of Deployment Diagrams.
11.	Project diagrams.

B. Sc. (Information Technology)		Semester – V	
Course Name: Internet of Things Practical		Course Code: SBTTECP502	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2½	50
	Internal	--	--

Practical No	Details
0	Starting Raspbian OS, Familiarising with Raspberry Pi Components and interface, Connecting to ethernet, Monitor, USB.
1	Displaying different LED patterns with Raspberry Pi.
2	Displaying Time over 4-Digit 7-Segment Display using Raspberry Pi
3	Raspberry Pi Based Oscilloscope
4	Controlling Raspberry Pi with WhatsApp.
5	Setting up Wireless Access Point using Raspberry Pi
6	Fingerprint Sensor interfacing with Raspberry Pi
7	Raspberry Pi GPS Module Interfacing
8	IoT based Web Controlled Home Automation using Raspberry Pi
9	Visitor Monitoring with Raspberry Pi and Pi Camera
10	Interfacing Raspberry Pi with RFID.
11	Building Google Assistant with Raspberry Pi.

12	Installing Windows 10 IoT Core on Raspberry Pi

Raspberry Pi Kits and components should be made available in the ratio of 1 kit: 3 students minimum.

B. Sc. (Information Technology)		Semester – V	
Course Name: Advanced Web Programming Practical		**Course Code: : SBTTECP503**	
Periods per week (1 Period is 50 mi utes)		**3**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Practical Examination**	2½	50
	Internal	--	--

List of Practical	
1.	**Working with basic C# and ASP .NET**
a.	Create an application that obtains four int values from the user and displays the product.
b.	Create an application to demonstrate string operations.
c.	Create an application that receives the (Student Id, Student Name, Course Name, Date of Birth) information from a set of students. The application should also display the information of all the students once the data entered.
	Create an application to demonstrate following operations i. Generate Fibonacci series. ii. Test for prime numbers. iii. Test for vowels. iv. Use of foreach loop with arrays v. Reverse a number and find sum of digits of a number.
2.	**Working with Object Oriented C# and ASP .NET**
a.	Create simple application to perform following operations i. Finding factorial Value ii. Money Conversion iii. Quadratic Equation iv. Temperature Conversion
b.	Create simple application to demonstrate use of following concepts i. Function Overloading ii. Inheritance (all types) iii. Constructor overloading iv. Interfaces
c.	Create simple application to demonstrate use of following concepts i. Using Delegates and events ii. Exception handling
3.	**Working with Web Forms and Controls**
a.	Create a simple web page with various sever controls to demonstrate setting and use of their properties. (Example : AutoPostBack)

b.	Demonstrate the use of Calendar control to perform following operations.
	a) Display messages in a calendar control b) Display vacation in a calendar control
	c) Selected day in a calendar control using style d) Difference between two calendar dates
c.	Demonstrate the use of Treeview control perform following operations.
	a) Treeview control and datalist b) Treeview operations
4.	**Working with Form Controls**
a.	Create a Registration form to demonstrate use of various Validation controls.
b.	Create Web Form to demonstrate use of Adrotator Control.
c.	Create Web Form to demonstrate use User Controls.
5.	**Working with Navigation, Beautification and Master page.**
a.	Create Web Form to demonstrate use of Website Navigation controls and Site Map.
b.	Create a web application to demonstrate use of Master Page with applying Styles and Themes for page beautification.
c.	Create a web application to demonstrate various states of ASP.NET Pages.
6.	**Working with Database**
a.	Create a web application bind data in a multiline textbox by querying in another textbox.
b.	Create a web application to display records by using database.
c.	Demonstrate the use of Datalist link control.
7.	**Working with Database**
a.	Create a web application to display Databbinding using dropdownlist control.
b.	Create a web application for to display the phone no of an author using database.
c.	Create a web application for inserting and deleting record from a database. (Using Execute-Non Query).
8.	**Working with data controls**
a.	Create a web application to demonstrate various uses and properties of SqlDataSource.
b.	Create a web application to demonstrate data binding using DetailsView and FormView Control.
c.	Create a web application to display Using Disconnected Data Access and Databinding using GridView.
9.	**Working with GridView control**

a.	Create a web application to demonstrate use of GridView control template and GridView hyperlink.
b.	Create a web application to demonstrate use of GridView button column and GridView events.
c.	Create a web application to demonstrate GridView paging and Creating own table format using GridView.
10.	**Working with AJAX and XML**
a.	Create a web application to demonstrate reading and writing operation with XML.
b.	Create a web application to demonstrate Form Security and Windows Security with proper Authentication and Authorization properties.
c.	Create a web application to demonstrate use of various Ajax controls.
11.	**Programs to create and use DLL**

B. Sc. (Information Technology)		Semester – V	
Course Name: Linux System Administration		**Course Code: : SBTTECP504**	
Periods per week (1 Period is 50 minutes)		**3**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Practical Examination**	**2½**	**50**
	Internal	**--**	**--**

Practical No	Details
0	**Installation of RHEL 6.X**
1	**Graphical User Interface and Command Line Interface and Processes**
a	Exploring the Graphical Desktop
b	The Command Line Interface
c	Managing Processes
2	**Storage Devices and Links, Backup and Repository**
b	Working with Storage Devices and Links
a	Making a Backup
b	Creating a Repository

3		**Working with RPMsm Storage and Networking**
	a	Using Query Options
	b	Extracting Files From RPMs
	c	Configuring and Managing Storage
	d	Connecting to the Network
4		**Working with Users, Groups, and Permissions**
5		**Firewall and Cryptographic services**
	a	Securing Server with iptables
	b	Setting Up Cryptographic Services
6		**Configuring Server for File Sharing**
	a	Configuring NFS Server and Client
	b	Configuring Samba
	c	Configuring FTP
7		**DNS, DHCP and Mail Server**
	a	Configuring DNS
	b	Configuring DHCP
	c	Setting Up a Mail Server
8		**Web Server**
	a	Configuring Apache on Red Hat Enterprise Linux
	b	Writing a Script to Monitor Activity on the Apache Web Server
	c	Using the select Command
9		**Shell Scripts and High-Availability Clustering**
	a	Writing Shell Scripts
	b	Configuring Booting with GRUB
	c	Configuring High Availability Clustering
10		**Setting Up an Installation Server**
	a	Configuring Network Server as an Installation Server
	b	Setting Up a TFTP and DHCP Server for PXE Boot

B. Sc. (Information Technology)		Semester – V	
Course Name: Enterprise Java		Course Code: SBTTECP505	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2½	50
	Internal	--	--

List of Practical	
1.	**Implement the following Simple Servlet applications.**
a.	Create a simple calculator application using servlet.
b.	Create a servlet for a login page. If the username and password are correct then it says message "Hello <username>" else a message "login failed"
c.	Create a registration servlet in Java using JDBC. Accept the details such as Username, Password, Email, and Country from the user using HTML Form and store the registration details in the database.
2.	**Implement the following Servlet applications with Cookies and Sessions.**
a.	Using Request Dispatcher Interface create a Servlet which will validate the password entered by the user, if the user has entered "Servlet" as password, then he will be forwarded to Welcome Servlet else the user will stay on the index.html page and an error message will be displayed.
b.	Create a servlet that uses Cookies to store the number of times a user has visited servlet.
c.	Create a servlet demonstrating the use of session creation and destruction. Also check whether the user has visited this page first time or has visited earlier also using sessions.
3.	**Implement the Servlet IO and File applications.**
a.	Create a Servlet application to upload and download a file.
b.	Develop Simple Servlet Question Answer Application using Database.
c.	Create simple Servlet application to demonstrate Non-Blocking Read Operation.
4.	**Implement the following JSP applications.**

a.	Develop a simple JSP application to display values obtained from the use of intrinsic objects of various types.
b.	Develop a simple JSP application to pass values from one page to another with validations. (Nametxt, age-txt, hobbies-checkbox, email-txt, gender-radio button).
c.	Create a registration and login JSP application to register and authenticate the user based on username and password using JDBC.
5.	**Implement the following JSP JSTL and EL Applications.**
a.	Create an html page with fields, eno, name, age, desg, salary. Now on submit this data to a JSP page which will update the employee table of database with matching eno.
b.	Create a JSP page to demonstrate the use of Expression language.
c.	Create a JSP application to demonstrate the use of JSTL.
6.	**Implement the following EJB Applications.**
a.	Create a Currency Converter application using EJB.
b.	Develop a Simple Room Reservation System Application Using EJB.
c.	Develop simple shopping cart application using EJB [Stateful Session Bean].
7.	**Implement the following EJB applications with different types of Beans.**
a.	Develop simple EJB application to demonstrate Servlet Hit count using Singleton Session Beans.
b.	Develop simple visitor Statistics application using Message Driven Bean [Stateless Session Bean].
c.	Develop simple Marks Entry Application to demonstrate accessing Database using EJB.
8.	**Implement the following JPA applications.**
a.	Develop a simple Inventory Application Using JPA.
b.	Develop a Guestbook Application Using JPA.
c.	Create simple JPA application to store and retrieve Book details.
9.	**Implement the following JPA applications with ORM and Hibernate.**
a.	Develop a JPA Application to demonstrate use of ORM associations.
b.	Develop a Hibernate application to store Feedback of Website Visitor in MySQL Database.
c.	Develop a Hibernate application to store and retrieve employee details in MySQL Database.

10.	Implement the following Hibernate applications.
a.	Develop an application to demonstrate Hibernate One- To -One Mapping Using Annotation.
b.	Develop Hibernate application to enter and retrieve course details with ORM Mapping.
c.	Develop a five page web application site using any two or three Java EE Technologies.

SEMESTER VI

SEMESTER VI		
Course Code	Course Title	Credits
SBTTEC601	Software Quality Assurance	2
SBTTEC602	Security in Computing	2
SBTTEC603	Business Intelligence	2
SBTTEC604	Data Science	2
SBTTEC605	Project	2
SBTTECP601	Software Quality Assurance Practical	2
SBTTECP602	Security in Computing Practical	2
SBTTECP603	Business Intelligence Practical	2
SBTTECP604	Data Science Practical	2
SBTTECP605	Project Report and Viva	2
Total Credits		20

B. Sc. (Information Technology)	Semester – VI
Course Name: Software Quality Assurance	Course Code: SBTTEC601

Course objectives
1.To brief different concepts related to quality of a project and process
2. To describe the testing process challenges and the cost of testing
3. To brief various testing methodology
4. To describe the concepts related to verification and validation
5. to explore various testing strategies depending on the system in use

Course outcomes
1. What is continuous improvement quality cycle.
2. Describe traceability matrix giving an example.
3. Compare the white box and Blackbox strategies used in testing.
4. For an attendance application developed, state various V&V activities undertaken
5. Compare web application testing and mobile application testing.

Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2½	75
	Internal	--	25

Unit	Details	Lectures

I	**Introduction to Quality:** Historical Perspective of Quality, What is Quality? (Is it a fact or perception?), Definitions of Quality, Core Components of Quality, Quality View, Financial Aspect of Quality, Customers, Suppliers and Processes, Total Quality Management (TQM), Quality Principles of Total Quality Management, Quality Management Through Statistical Process Control, Quality Management Through Cultural Changes, Continual (Continuous) Improvement Cycle, Quality in Different Areas, Benchmarking and Metrics, Problem Solving Techniques, Problem Solving Software Tools. **Software Quality:** Introduction, Constraints of Software Product Quality Assessment, Customer is a King, Quality and Productivity Relationship, Requirements of a Product, Organisation Culture, Characteristics of Software, Software Development Process, Types of Products, Schemes of Criticality Definitions, Problematic Areas of Software Development Life Cycle, Software Quality Management, Why Software Has Defects? Processes Related to Software Quality, Quality Management System Structure, Pillars of Quality Management System, Important Aspects of Quality Management.	**12**

II	**Fundamentals of testing:** Introduction, Necessity of testing, What is testing? Fundamental test process, The psychology of testing, Historical Perspective of Testing, Definitions of Testing, Approaches to Testing, Testing During Development Life Cycle, Requirement Traceability Matrix, Essentials of Software Testing, Workbench, Important Features of Testing Process, Misconceptions About Testing, Principles of Software Testing, Salient Features of Good Testing, Test Policy, Test Strategy or Test Approach, Test Planning, Testing Process	

	and Number of Defects Found in Testing, Test Team Efficiency, Mutation Testing, Challenges in Testing, Test Team Approach, Process Problems Faced by Testing, Cost Aspect of Testing, Establishing Testing Policy, Methods, Structured Approach to Testing, Categories of Defect, Defect, Error, or Mistake in Software, Developing Test Strategy, Developing Testing Methodologies (Test Plan), Testing Process, Attitude Towards Testing (Common People Issues), Test Methodologies/Approaches, People Challenges in Software Testing, Raising Management Awareness for Testing, Skills Required by Tester,	**12**
	Testing throughout the software life cycle, Software development models, Test levels, Test types, the targets of testing, Maintenance testing	

III	**Unit Testing: Boundary Value Testing:** Normal Boundary Value Testing, Robust Boundary Value Testing, Worst-Case Boundary Value Testing, Special Value Testing, Examples, Random Testing, Guidelines for Boundary Value Testing, **Equivalence Class Testing:** Equivalence Classes, Traditional Equivalence Class Testing, Improved Equivalence Class Testing, Edge Testing, Guidelines and Observations. **Decision Table– Based Testing:** Decision Tables, Decision Table Techniques, Cause-and-Effect Graphing, Guidelines and Observations, **Path Testing:** Program Graphs, DD-Paths, Test Coverage Metrics, Basis Path Testing, Guidelines and Observations, **Data Flow Testing:** Define/Use Testing, Slice-Based Testing, Program Slicing Tools.	**12**
IV	**Software Verification and Validation:** Introduction, Verification, Verification Workbench, Methods of Verification, Types of reviews on the basis od Stage Phase, Entities involved in verification, Reviews in testing lifecycle, Coverage in Verification, Concerns of Verification, Validation, Validation Workbench, Levels of Validation, Coverage in Validation, Acceptance Testing, Management of Verification and Validation, Software development verification and validation activities. **V-test Model:** Introduction, V-model for software, Testing during Proposal stage, Testing during requirement stage, Testing during test planning phase, Testing during design phase, Testing during coding, VV Model, Critical Roles and Responsibilities.	**12**
V	**Levels of Testing:** Introduction, Proposal Testing, Requirement Testing, Design Testing, Code Review, Unit Testing, Module Testing, Integration Testing, Big-Bang Testing, Sandwich Testing, Critical Path First, Sub System Testing, System Testing, Testing Stages. **Special Tests:** Introduction, GUI testing, Compatibility Testing, Security Testing, Performance Testing, Volume Testing, Stress Testing, Recovery Testing, Installation Testing, Requirement Testing, Regression Testing, Error Handling Testing, Manual Support Testing, Intersystem Testing, Control Testing, Smoke Testing, Adhoc Testing, Parallel Testing, Execution Testing, Operations Testing, Compliance Testing, Usability Testing, Decision Table Testing, Documentation Testing, Training testing, Rapid Testing, Control flow graph, Generating tests on the basis of Combinatorial Designs, State Graph, Risk Associated with New Technologies, Process maturity level of Technology, Testing Adequacy of Control in New technology usage, Object Oriented Application Testing, Testing of Internal Controls, COTS Testing, Client Server Testing, Web Application Testing, Mobile Application Testing, eBusiness eCommerce Testing, Agile Development Testing, Data Warehousing Testing.	**12**

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Software Testing and Continuous Quality Improvement	William E. Lewis	CRC Press	Third	2016
2	Software Testing: Principles, Techniques and Tools	M. G. Limaye	TMH		2017
3.	Foundations of Software Testing	Dorothy Graham, Erik van Veenendaal, Isabel Evans, Rex Black	Cengage Learning	3rd	
4.	Software Testing: A Craftsman's Approach	Paul C. Jorgenson	CRC Press	4th	2017

B. Sc. (Information Technology)	Semester – VI
Course Name: Security in Computing	Course Code: SBTTEC602

COURSE OBJECTIVES

1. **The main objective of this course is to enable students to learn the Importance of Information Protection and study how to build a security program.**
2. **Students will be able to distinguish between Authentication and Authorization. Introduction to cryptography.**
3. **To acquire basic knowledge of Secure Network Design, Firewalls and Wireless Network Security.**
4. **Students will be able to explain the Intrusion Detection and Prevention Systems. Will learn overview of Operating System Security Models**
5. **Understanding Virtual Machines, Cloud Computing and Physical Security concepts along with learning how to Secure Application Design**

COURSE OUTCOMES

1. **State and explain the Importance of Information Protection. Define CIA triad.**
2. **Explain the importance of Authentication and Authorization. Discuss Encryption in brief. Explain General Database and Storage Security Concepts**
3. **State Network Device and wireless security measures and Explain Firewalls**
4. **State and explain the features of Intrusion detection and Security Information and Event Management. Explain VoIP Vulnerabilities and Countermeasures**
5. **State and explain Virtual Machines and Cloud Computing concept in detail. Explain Secure Development Lifecycle**

Periods per week (1 Period is 50 minutes)	**5**
Credits	**2**

Learning objectives: Students will learn the basic concepts in **computer security** including software vulnerability analysis and defense, networking and wireless **security**, applied cryptography, as well as ethical, legal, social and economic facets of **security**		**Hours**	**Marks**
Evaluation System	**Theory Examination**	2½	75
	Internal	--	25

Unit	Details	Lectures
I	**Information Security Overview** : The Importance of Information Protection, The Evolution of Information Security, Justifying Security Investment, Security Methodology, How to Build a Security Program, The Impossible Job, The Weakest Link, Strategy and Tactics, Business Processes vs. Technical Controls. **Risk Analysis**: Threat Definition, Types of Attacks, Risk Analysis. **Secure Design Principles**: The CIA Triad and Other Models, Defense Models, Zones of Trust, Best Practices for Network Defense.	**12**

II	**Authentication and Authorization**: Authentication, Authorization **Encryption**: A Brief History of Encryption, Symmetric-Key Cryptography, Public Key Cryptography, Public Key Infrastructure. **Storage Security**: Storage Security Evolution, Modern Storage Security, Risk Remediation, Best Practices. **Database Security**: General Database Security Concepts, Understanding Database Security Layers, Understanding DatabaseLevel Security, Using Application Security, Database Backup and Recovery, Keeping Your Servers Up to Date, Database Auditing and Monitoring.	**12**
III	**Secure Network Design**: Introduction to Secure Network Design, Performance, Availability, Security. **Network Device Security**: Switch and Router Basics, Network Hardening. **Firewalls**: Overview, The Evolution of Firewalls, Core Firewall Functions, Additional Firewall Capabilities, Firewall Design. **Wireless Network Security**: Radio Frequency Security Basics, DataLink Layer Wireless Security Features, Flaws, and Threats, Wireless Vulnerabilities and Mitigations, Wireless Network Hardening Practices and Recommendations, Wireless Intrusion Detection and Prevention, Wireless Network Positioning and Secure Gateways.	**12**
IV	**Intrusion Detection and Prevention Systems**: IDS Concepts, IDS Types and Detection Models, IDS Features, IDS Deployment Considerations, Security Information and Event Management (SIEM). **Voice over IP (VoIP) and PBX Security**: Background, VoIP Components, VoIP Vulnerabilities and Countermeasures, PBX, TEM: Telecom Expense Management. **Operating System Security Models**: Operating System Models, Classic Security Models, Reference Monitor, Trustworthy Computing, International Standards for Operating System Security.	**12**
V	**Virtual Machines and Cloud Computing**: Virtual Machines, Cloud Computing. **Secure Application Design**: Secure Development Lifecycle, Application Security Practices, Web Application Security, Client Application Security, Remote Administration Security. **Physical Security**: Classification of Assets, Physical Vulnerability Assessment, Choosing Site Location for Security, Securing Assets: Locks and Entry Controls, Physical Intrusion Detection.	**12**

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	The Complete Reference: Information Security	Mark RhodesOusley	McGraw-Hill	2nd	2013
2.	Essential Cybersecurity Science	Josiah Dykstra	O'Reilly	Fifth	2017

3.	Principles of Computer Security: CompTIA Security+ and Beyond	Wm.Arthur Conklin, Greg White	McGraw Hill	Second	2010

B. Sc. (Information Technology)		Semester – VI	
Course Name: Business Intelligence		Course Code : SBTTEC603	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
Course Objectives:	**Hours**	**Marks**	
1. To introduce students with concepts of business intelligence by creating a background of data warehousing and data mining.			

	Hours	Marks
Course Objectives: 1. To introduce students with concepts of business intelligence by creating a background of data warehousing and data mining. 2. The entire purpose of Business Intelligence is to support and facilitate better business decisions. 3. BI allows organizations access to information that is critical to the success of multiple areas including sales, finance, marketing, and a multitude of other areas and departments. 4. Identify the technological architecture that makes up BI systems 5. Plan the implementation of a BI system		
Course Outcomes: 1. Define Business intelligence architectures and Ethics 2. Critically evaluate use of BI for supporting decision making in an organization 3. Evaluate classification models 4. To design business intelligence applications 5. To construct an expert system		

Evaluation System		Hours	Marks
	Theory Examination	2½	75
	Internal	--	25

Unit	Details	Lectures
I	**Business intelligence:** Effective and timely decisions, Data, information and knowledge, The role of mathematical models, Business intelligence architectures, Ethics and business intelligence **Decision support systems:** Definition of system, Representation of the decision-making process, Evolution of information systems, Definition of decision support system, Development of a decision support system	12
II	**Mathematical models for decision making:** Structure of mathematical models, Development of a model, Classes of models **Data mining:** Definition of data mining, Representation of input data , Data mining process, Analysis methodologies **Data preparation**: Data validation, Data transformation, Data reduction	12

III	**Classification**: Classification problems, Evaluation of classification models, Bayesian methods, Logistic regression, Neural networks, Support vector machines **Clustering:** Clustering methods, Partition methods, Hierarchical methods, Evaluation of clustering models	**12**
IV	**Business intelligence applications:** **Marketing models**: Relational marketing, Sales force management, **Logistic and production models**: Supply chain optimization, Optimization models for logistics planning, Revenue management systems. **Data envelopment analysis**: Efficiency measures, Efficient frontier, The CCR model, Identification of good operating practices	**12**
V	**Knowledge Management:** Introduction to Knowledge Management, Organizational Learning and Transformation, Knowledge Management Activities, Approaches to Knowledge Management, Information Technology (IT) In Knowledge Management, Knowledge Management Systems Implementation, Roles of People in Knowledge Management **Artificial Intelligence and Expert Systems:** Concepts and Definitions of Artificial Intelligence, Artificial Intelligence Versus Natural Intelligence, Basic Concepts of Expert Systems, Applications of Expert Systems, Structure of Expert Systems, Knowledge Engineering, Development of Expert Systems	**12**

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Business Intelligence: Data Mining and Optimization for Decision Making	Carlo Vercellis	Wiley	First	2009
2.	Decision support and Business Intelligence Systems	Efraim Turban, Ramesh Sharda, Dursun Delen	Pearson	Ninth	2011
3.	Fundamental of Business Intelligence	Grossmann W, Rinderle-Ma	Springer	First	2015

B. Sc. (Information Technology)		Semester – VI	
Course Name: Data Science		Course Code: SBTTEC604	
Periods per week (1 Period is 50 minutes),		5	
Credits		2	
Course Objectives: Upon completion of this course, the students will be able to 1. Identify the need for data science and solve basic problems using Python built-in data types and their methods 2. Employ efficient storage and data operations using NumPy arrays. 3. Do data pre-processing and visualization using Pandas 4. Gain practical, hands-on experience with statistics programming languages and big data tools through coursework and applied research experiences. 5. Develop in depth understanding of the key technologies in data science and business analytics: data mining, machine learning, visualization techniques, predictive modeling, and statistics.		Hours	Marks

Course Outcomes:
    1. Define Data science framework
    2. Classifying the management layers
    3. Understanding functionality of Numpy and Pandas
    4. Designing Plotting with Pandas
    5. Building process and transform setup			

Evaluation System	Theory Examination	2½	75
	Internal	--	25

Unit	Details	Lectures
I	**<u>Introduction to Data Science and Python Programming</u>** Introduction to Data Science Decision Making- Looping- Loop Control statement- Math and Random number functions. User defined functions - function arguments & its types using Numpy and Pandas Data Science Technology Stack: Rapid Information Factory Ecosystem, Data Science Storage Tools, Data Lake, Data Vault, Data Warehouse Bus Matrix, Data Science Processing Tools ,Spark, Mesos, Akka , Cassandra, Kafka, Elastic Search, R ,Scala, Python, MQTT, The Future Layered Framework: Definition of Data Science Framework, CrossIndustry Standard Process for Data Mining (CRISP-DM), Homogeneous Ontology for Recursive Uniform Schema, The Top Layers of a Layered Framework, Layered Framework for High-Level Data Science and Engineering	12
II	**<u>Three Management Layers</u>** Three Management Layers: Operational Management Layer, ProcessingStream Definition and Management, Audit, Balance, and Control Layer, Balance, Control, Yoke Solution, Cause-and-Effect, Analysis System, Functional Layer, Data Science Process Retrieve Superstep : Data Lakes, Data Swamps, Training the Trainer Model, Understanding the Business Dynamics of the Data Lake, Actionable Business Knowledge from Data Lakes, Engineering a Practical Retrieve Superstep, Connecting to Other Data Sources	12
III	**<u>Introduction to Numpy and Pandas</u>** NumPy Basics: Arrays and Vectorized Computation- The NumPy ndarray-Creating ndarrays- Data Types for ndarrays- Arithmetic with NumPy Arrays- Basic Indexing and Slicing - Boolean Indexing-Transposing Arrays and Swapping Axes. Universal Functions: Fast Element-Wise Array Functions-Mathematical and Statistical Methods-SortingUnique and Other Set Logic Introduction to pandas Data Structures: Series, DataFrame, Essential Functionality: Dropping EntriesIndexing, Selection, and Filtering-Function Application and Mapping- Sorting and Ranking. Summarizing and Computing Descriptive Statistics- Unique Values, Value Counts, and Membership. Reading and Writing Data in Text Format.	12
IV	**<u>DATA CLEANING, PREPARATION AND VISUALIZATION</u>** Data Cleaning and Preparation: Handling Missing Data - Data Transformation: Removing Duplicates, Transforming Data Using a Function or Mapping, Replacing Values, Detecting and Filtering Outliers- String Manipulation: Vectorized String Functions in pandas. Plotting with pandas: Line Plots, Bar Plots, Histograms and Density Plots, Scatter or Point Plots.	12

V	Process Setup , Transform Setup	12
	Process Superstep : Data Vault, Time-Person-Object-Location-Event Data Vault, Data Science Process, Data Science Transform Superstep : Transform Superstep, Building a Data Warehouse, Transforming with Data Science, Hypothesis Testing, Overfitting and Underfitting, Precision-Recall, Cross-Validation Test.Univariate Analysis, Bivariate Analysis, Multivariate Analysis, Linear Regression, Logistic Regression, Clustering Techniques, ANOVA, Principal Component Analysis (PCA), Decision Trees, Support Vector Machines, Networks, Clusters, and Grids, Data Mining, Pattern Recognition, Machine Learning, Bagging Data,Random Forests, Computer Vision (CV) , Natural Language Processing (NLP), Neural Networks, TensorFlow	

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1	Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython	Wes McKinney	O'Reilly	2nd Edition	2018
2	Python Data Science Handbook: Essential Tools for Working with Data	Jake VanderPlas	O'Reilly		2017
3	Practical Data Science	Andreas François Vermeulen	APress		2018
4	Principles of Data Science	Sinan Ozdemir	PACKT		2016

B. Sc. (Information Technology)		Semester – VI	
Course Name: Project		Course Code: SBTTEC605	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Project Examination	2½	100
	Internal	--	-

Project Implementation Semester VI

Semester VI report has to be hard bound with golden embossing. Students will be evaluated based on the implementation, report and viva voce in Semester VI.

I. OBJECTIVES

- Describe the Systems Development Life Cycle (SDLC).
- Evaluate systems requirements. • Complete a problem definition.
- Evaluate a problem definition.
- Determine how to collect information to determine requirements.
- Perform and evaluate feasibility studies like cost-benefit analysis, technical feasibility, time feasibility and Operational feasibility for the project.
- Work on data collection methods for fact finding.
- Construct and evaluate data flow diagrams.
- Construct and evaluate data dictionaries.
- Evaluate methods of process description to include structured English, decision tables and decision trees.
- Evaluate alternative tools for the analysis process.
- Create and evaluate such alternative graphical tools as systems flow charts and state transition diagrams.
- Decide the S/W requirement specifications and H/W requirement specifications.
- Plan the systems design phase of the SDLC.
- Distinguish between logical and physical design requirements.

- Design and evaluate system outputs.
- Design and evaluate systems inputs.
- Design and evaluate validity check for input data. Design and evaluate user interfaces for input.
- Design and evaluate file structures to include the use of indexes.
- Estimate storage requirements.
- Explain the various file update processes based on the standard file organizations.
- Decide various data structures.
- Construct and evaluate entity-relationship (ER) diagrams for RDBMS related projects.
- Perform normalization for the unnormalized tables for RDBMS related projects
- Decide the various processing systems to include distributed, client/server, online and others.
- Perform project cost estimates using various techniques.
- Schedule projects using both GANTT and PERT charts.
- Perform coding for the project.
- Documentation requirements and prepare and evaluate systems documentation.
- Perform various systems testing techniques/strategies to include the phases of testing.
- Systems implementation and its key problems.
- Generate various reports.
- Be able to prepare and evaluate a final report.
- Brief the maintenance procedures and the role of configuration management in operations.
- To decide the future scope and further enhancement of the system.
- Plan for several appendices to be placed in support with the project report documentation.
- Decide the various processing systems to include distributed, client/server, online and others.
- Perform project cost estimates using various techniques.
- Schedule projects using both GANTT and PERT charts.
- Perform coding for the project.
- Documentation requirements and prepare and evaluate systems documentation.
- Perform various systems testing techniques/strategies to include the phases of testing.
- Work effectively as an individual or as a team member to produce correct, efficient, well-organized and documented programs in a reasonable time.

- Recognize problems that are amenable to computer solutions, and knowledge of the tool necessary for solving such problems.

- Develop of the ability to assess the implications of work performed.

- Get good exposure and command in one or more application areas and on the software

- Develop quality software using the software engineering principles

- Develop of the ability to communicate effectively.

II. Type of the Project

The majority of the students are expected to work on a real-life project preferably in some industry/ Research and Development Laboratories/Educational Institution/Software Company. Students are encouraged to work in the areas listed below. However, it is *not mandatory* for a student to work on a real-life project. The student can formulate a project problem with the help of her/his Guide and submit the project proposal of the same. **Approval of the project proposal is mandatory**. If approved, the student can commence working on it, and complete it. Use the latest versions of the software packages for the development of the project.

III. SOFTWARE AND BROAD AREAS OF APPLICATION

FRONT END / GUI Tools	.Net Technologies,Java
DBMS/BACK END	Oracle, SQL Plus, MY SQL, SQL Server,
LANGUAGES	C, C++, Java, VC++, C#, R,Python
SCRIPTING LANGUAGES	PHP,JSP, SHELL Scripts (Unix), TcL/TK,
.NET Platform	F#,C#. Net, Visual C#. Net, ASP.Net
MIDDLE WARE (COMPONENT) TECHNOLOGIES	COM/DCOM, Active-X, EJB
UNIX INTERNALS	Device Drivers, RPC, Threads, Socket programming
NETWORK/WIRELESS TECHNOLOGIES	-
REALTIME OPERATING SYSTEM/ EMBEDDED SKILLS	LINUX, Raspberry Pi, Arduino, 8051
APPLICATION AREAS	Financial / Insurance / Manufacturing / Multimedia / Computer Graphics / Instructional Design/ Database Management System/ Internet / Intranet / Computer Networking-Communication Software development/ ECommerce/ ERP/ MRP/ TCP-IP programming / Routing protocols programming/ Socket programming.

B. Sc. (Information Technology)		Semester – VI	
Course Name: Software Quality Assurance Practical		Course Code: SBTTECP601	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2½	50
	Internal	--	-

1.You have got the brilliant idea of setting up a company that sells testing services to software houses. Make a strategic plan for your company, taking into account the following issues:

What is the testing process that will be followed in the company?

What is the focus of the testing services?

What kind of people are you going to hire as staff for the company?

How are you going to validate that a testing project carried out in the company has been beneficial to the customer?

What kind of automated tools will the company use?

2. Prepare a small project and submit SRS, design, coding and test plan.

3. The program reads an arbitrary number of temperatures (as integer numbers) within the range - 60°C … +60°C and prints their mean value. Design test cases for testing the program with the black-box strategy.

4. When getting a person's weight and height as input, the program prints the person's body weight index. The weight is given in kilograms (as a real number, for instance: 82,0) and the height in meters (as a real number, for instance: 1,86). The body weight index equals weight divided by height squared: weight / (height ´height). Design test cases for testing the program with the black-box strategy.

5. Let us study the following program:
x=0; read(y);
while (y > 100) { x=x+y; read(y); } **if**
(y < 200) print(x) **else** print(y);
a) Construct a control-flow graph for the program.
b) Design test cases for reaching complete branch coverage over the program. Use as few test cases as possible.

6. Design test cases for the following program with the "simple loop" strategy: x=0;
read(y);
while ((y > 100) && (x < 10)) { x=x+1; read(y); } print(y);
a) Construct a data-flow graph for the program with respect to variable *x*.

b) Which execution paths have to be traversed during testing, in order to reach complete *alldefinitions* coverage with respect to variable *x*?
c) Minimize the number of paths and tests.
d) Which execution paths have to be traversed during testing, in order to reach complete *alluses* coverage with respect to variable *x*? Minimize the number of paths and tests.
e) Design test cases for reaching the (minimal) complete *all-uses* coverage with respect to variable *x*.

7. MANUAL TESTING for the project
a) Walkthrough
b) Whitebox Testing
c) Blackbox Testing
d) Unit Testing
e) Integration Testing

8. Functional Testing
a) Boundary value Testing
b) Equivalence class testing
c) Decision Table based testing
d) Cause-effect graphs

9. Regression Testing using automated testing for website.

10. AUTOMATED TESTING for websites
a) Load Testing
b) Performance Testing

B. Sc. (Information Technology)		Semester – VI	
Course Name: Security in Computing Practical		**Course Code:SBTTECP602**	
Periods per week (1 Period is 50 minutes)		**3**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Practical Examination**	2½	50
	Internal	--	-

Practical No	Details
1	**Configure Routers**
a	OSPF MD5 authentication.
b	NTP.

c	to log messages to the syslog server.	
d	to support SSH connections.	
2	**Configure AAA Authentication**	
a	Configure a local user account on Router and configure authenticate on the console and vty lines using local AAA	
b	Verify local AAA authentication from the Router console and the PC-A client	
3	**Configuring Extended ACLs**	
a	**Configure, Apply and Verify an Extended Numbered ACL**	
4	**Configure IP ACLs to Mitigate Attacks and IPV6 ACLs**	
a	Verify connectivity among devices before firewall configuration.	
b	Use ACLs to ensure remote access to the routers is available only from management station PC-C.	
c	Configure ACLs on to mitigate attacks.	
d	Configuring IPv6 ACLs	
5	**Configuring a Zone-Based Policy Firewall**	
6	Configure IOS Intrusion Prevention System (IPS) Using the CLI	
a	Enable IOS IPS.	
b	Modify an IPS signature.	
7	**Layer 2 Security**	
a	Assign the Central switch as the root bridge.	
b	Secure spanning-tree parameters to prevent STP manipulation attacks.	
c	Enable port security to prevent CAM table overflow attacks.	
8	**Layer 2 VLAN Security**	
9	**Configure and Verify a Site-to-Site IPsec VPN Using CLI**	
10	**Configuring ASA Basic Settings and Firewall Using CLI**	
a	Configure basic ASA settings and interface security levels using CLI	
b	Configure routing, address translation, and inspection policy using CLI	

c	Configure DHCP, AAA, and SSH
d	Configure a DMZ, Static NAT, and ACLs

B. Sc. (Information Technology)		Semester – VI	
Course Name: Business Intelligence Practical		**Course Code:SBTTECP603**	
Periods per week (1 Period is 50 minutes)		**3**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Practical Examination**	2½	50
	Internal	--	--

Practical No	Details
1	Import the legacy data from different sources such as (Excel , SqlServer, Oracle etc.) and load in the target system. (You can download sample database such as Adventureworks, Northwind, foodmart etc.)
2	Perform the Extraction Transformation and Loading (ETL) process to construct the database in the Sqlserver.
3	a. Create the Data staging area for the selected database. b. Create the cube with suitable dimension and fact tables based on ROLAP, MOLAP and HOLAP model.
4	a.Create the ETL map and setup the schedule for execution. b. Execute the MDX queries to extract the data from the datawarehouse.
5	a. Import the datawarehouse data in Microsoft Excel and create the Pivot table and Pivot Chart. b. Import the cube in Microsoft Excel and create the Pivot table and Pivot Chart to perform data analysis.
6	Apply the what – if Analysis for data visualization. Design and generate necessary reports based on the data warehouse data.
7	Perform the data classification using classification algorithm.
8	Perform the data clustering using clustering algorithm.
9	Perform the Linear regression on the given data warehouse data.

10	Perform the logistic regression on the given data warehouse data.

The BI tools such as Tableau / Power BI / BIRT / R / Excel or any other can be used.

B. Sc. (Information Technology)		Semester – VI	
Course Name: Data Science Practical		Course Code: SBTTECP604	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2½	50
	Internal	--	--

Practical List

1. Reading different types of data sets (.txt, csv) from web and disk and writing in specific disk location
2. Implement and demonstrate the Find-S algorithm for finding the most specific hypothesis based on a given set of training data sample. Reading the training data from the csv file.
3. Write a program in python using pandas
   a. Reading Data
   b. Panda Data Frame 1
   c. Panda Data Frame 2
   d. Panda Data Frame 3
   e. Control Structure Function
   f. Exploratory Data analysis
   g. Data Visualization
   h. Dealing with missing value
4. For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples.

5. Conversion data model using Cassandra
6. Conversion from different format to Horus
    a. CSV to Horus
    b. XML to HORUS
    c. JSON to HORUS
    d. MYSQL database to HORUS
    e. Picture to HORUS
    f. Video to Horus
7. Utilities and Auditing
    a. Fixer Utilities
    b. Data Binding or Bucketing
    c. Aggerate of Data
    d. Outlier Detection
    e. Audit
8. Retrieving Data
    a. Data Processing
    b. Data Pattern
9. Reporting Data
    a. Directed Acyclic Graph
    b. Graphics
10. Working with Power BI
    a. Importing data from Excel
    b. Importing data from OData Feed
    c. Data visualization with power BI

B. Sc. (Information Technology)		Semester – VI	
Course Name: Project Report and Viva		Course Code: SBTTECP605	
Periods per week (1 Period is 50 m)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2½	50
	Internal	--	--

Introduction

The project report should be documented with scientific approach to the solution of the problem that the students have sought to address. The project report should be prepared in order to solve the problem in a methodical and professional manner, making due references to appropriate techniques, technologies and professional standards. The student should start the documentation process from the first phase of software development so that one can easily identify the issues to be focused upon in the ultimate project report. The student should also include the details from the project diary, in which they will record the progress of their project throughout the course. The project report should contain enough details to enable examiners to evaluate the work. The important points should be highlighted in the body of the report, with details often referred to appendices.

1.1 PROJECT REPORT:

Title Page

Original Copy of the Approved Proforma of the Project Proposal

Certificate of Authenticated work

Role and Responsibility Form

Abstract

Acknowledgement

Table of Contents

Table of Figures

CHAPTER 1: INTRODUCTION

1.1 Background

1.2 Objectives

1.3 Purpose, Scope, and Applicability
1.3.1 Purpose

1.3.2 Scope

1.3.3 Applicability

1.4 Achievements

1.5 Organisation of Report

CHAPTER 2: SURVEY OF TECHNOLOGIES

CHAPTER 3: REQUIREMENTS AND ANALYSIS

3.1 Problem Definition

3.2 Requirements Specification

3.3 Planning and Scheduling

3.4 Software and Hardware Requirements

3.5 Preliminary Product Description

3.6 Conceptual Models

CHAPTER 4: SYSTEM DESIGN

4.1 Basic Modules

4.2 Data Design

4.2.1 Schema Design

4.2.2 Data Integrity and Constraints

4.3 Procedural Design

4.3.1 Logic Diagrams

4.3.2 Data Structures

4.3.3 Algorithms Design

4.4 User interface design

4.5 Security Issues

4.6 Test Cases Design

The documentation should use tools like star UML, Visuo for windows, Rational Rose for design as part of Software Project Management Practical Course. The documentation should be spiral bound for semester V and the entire documentation should be hard bound during semester VI.

CHAPTER 5: IMPLEMENTATION AND TESTING

5.1 Implementation Approaches

5.2 Coding Details and Code Efficiency
5.2.1 Code Efficiency

5.3 Testing Approach

5.3.1 Unit Testing

5.3.2 Integrated Testing

5.3.3 Beta Testing

5.4 Modifications and Improvements

5.5 Test Cases

CHAPTER 6: RESULTS AND DISCUSSION

6.1 Test Reports

6.2 User Documentation

CHAPTER 7: CONCLUSIONS

7.1 Conclusion

7.1.1 Significance of the System

7.2 Limitations of the System

7.3 Future Scope of the Project

REFERENCES

GLOSSARY

APPENDIX A

V. EXPLANATION OF CONTENTS

Title Page

Sample format of Title page is given in Appendix 1 of this block. Students should follow the given format.

Original Copy of the Approved Proforma of the Project Proposal

Sample Proforma of Project Proposal is given in Appendix 2 of this block. Students should follow the given format.

Certificate of Authenticated work

Sample format of Certificate of Authenticated work is given in Appendix 3 of this block. Students should follow the given format.

Role and Responsibility Form

Sample format for Role and Responsibility Form is given in Appendix 4 of this block.

Students should follow the given format.

Abstract

This should be one/two short paragraphs (100-150 words total), summarising the project work. It is important that this is not just a re-statement of the original project outline. A suggested flow is background, project aims and main achievements. From the abstract, a reader should be able to ascertain if the project is of interest to them and, it should present results of which they may wish to know more details.

Acknowledgements

This should express student's gratitude to those who have helped in the preparation of project.

Table of Contents: The table of contents gives the readers a view of the detailed structure of the report. The students would need to provide section and subsection headings with associated pages.

The formatting details of these sections and subsections are given below.

Table of Figures: List of all Figures, Tables, Graphs, Charts etc. along with their page numbers in a table of figures.

Chapter 1: Introduction

The introduction has several parts as given below:

Background: A description of the background and context of the project and its relation to work already done in the area. Summarise existing work in the area concerned with the project work.

Objectives: Concise statement of the aims and objectives of the project. Define exactly what is going to be done in the project; the objectives should be about 30 /40 words.

Purpose, Scope and Applicability: The description of Purpose, Scope, and Applicability are given below:

• Purpose: Description of the topic of the project that answers questions on why this project is being done. How the project could improve the system its significance and theoretical framework. • Scope: A brief overview of the methodology, assumptions and limitations. The students should answer the question: What are the main issues being covered in the project? What are the main functions of the project?

• Applicability: The student should explain the direct and indirect applications of their work.
Briefly discuss how this project will serve the computer world and people.

Achievements: Explain what knowledge the student achieved after the completion of the work. What contributions has the project made to the chosen area? Goals achieved - describes the degree to which the findings support the original objectives laid out by the project. The goals may be partially or fully achieved, or exceeded.

Organisation of Report: Summarising the remaining chapters of the project report, in effect, giving the reader an overview of what is to come in the project report.

Chapter 2: Survey of Technologies

In this chapter Survey of Technologies should demonstrate the students awareness and understanding of Available Technologies related to the topic of the project. The student should give the detail of all the related technologies that are necessary to complete the project. The should describe the technologies available in the chosen area and present a comparative study of all those Available Technologies. Explain why the student selected the one technology for the completion of the objectives of the project.

Chapter 3: Requirements and Analysis

Problem Definition: Define the problem on which the students are working in the project.
Provide details of the overall problem and then divide the problem in to sub-problems. Define each subproblem clearly.

Requirements Specification: In this phase the student should define the requirements of the system, independent of how these requirements will be accomplished. The Requirements Specification describes the things in the system and the actions that can be done on these things. Identify the operation and problems of the existing system.

Planning and Scheduling: Planning and scheduling is a complicated part of software development. Planning, for our purposes, can be thought of as determining all the small tasks that must be carried out in order to accomplish the goal. Planning also takes into account, rules, known as constraints, which, control when certain tasks can or cannot happen. Scheduling can be thought of as determining whether adequate resources are available to carry out the plan. The student should show the Gantt chart and Program Evaluation Review Technique (PERT).

Software and Hardware Requirements: Define the details of all the software and hardware needed for the development and implementation of the project.

• Hardware Requirement: In this section, the equipment, graphics card, numeric co-processor, mouse, disk capacity, RAM capacity etc. necessary to run the software must be noted.

• Software Requirements: In this section, the operating system, the compiler, testing tools, linker, and the libraries etc. necessary to compile, link and install the software must be listed.

Preliminary Product Description: Identify the requirements and objectives of the new system.

Define the functions and operation of the application/system the students are developing as project.

Conceptual Models: The student should understand the problem domain and produce a model of the system, which describes operations that can be performed on the system, and the allowable sequences of those operations. Conceptual Models could consist of complete Data Flow Diagrams, ER diagrams, Object-oriented diagrams, System Flowcharts etc.

Chapter 4: System Design

Describes desired features and operations in detail, including screen layouts, business rules, process diagrams, pseudocode and other documentation.

Basic Modules: The students should follow the divide and conquer theory, so divide the overall problem into more manageable parts and develop each part or module separately. When all modules are ready, the student should integrate all the modules into one system. In this phase, the student should briefly describe all the modules and the functionality of these modules.

Data Design: Data design will consist of how data is organised, managed and manipulated.

• Schema Design: Define the structure and explanation of schemas used in the project.

• Data Integrity and Constraints: Define and explain all the validity checks and constraints provided to maintain data integrity.

Procedural Design: Procedural design is a systematic way for developing algorithms or procedurals.

• Logic Diagrams: Define the systematical flow of procedure that improves its comprehension and helps the programmer during implementation. e.g., Control Flow Chart, Process Diagrams etc.

• Data Structures: Create and define the data structure used in procedures.

• Algorithms Design: With proper explanations of input data, output data, logic of processes, design and explain the working of algorithms.

User Interface Design: Define user, task, environment analysis and how to map those requirements in order to develop a "User Interface". Describe the external and internal components and the architecture of user interface. Show some rough pictorial views of the user interface and its components.

Security Issues: Discuss Real-time considerations and Security issues related to the project and explain how the student intends avoiding those security problems. What are the security policy plans and architecture?

Test Cases Design: Define test cases, which will provide easy detection of errors and mistakes with in a minimum period of time and with the least effort. Explain the different conditions in which the students wish to ensure the correct working of the project.

Chapter 5: Implementation and Testing

Implementation Approaches: Define the plan of implementation, and the standards the students have used in the implementation.

Coding Details and Code Efficiency: Students not need include full source code, instead, include only the important codes (algorithms, applets code, forms code etc). The program code should contain comments needed for explaining the work a piece of code does. Comments may be needed to explain why it does it, or, why it does a particular way.

The student can explain the function of the code with a shot of the output screen of that program code.

• Code Efficiency: The student should explain how the code is efficient and how the students have handled code optimisation.

Testing Approach: Testing should be according to the scheme presented in the system design chapter and should follow some suitable model – e.g., category partition, state machine-based. Both functional testing and user-acceptance testing are appropriate. Explain the approach of testing.

• Unit Testing: Unit testing deals with testing a unit or module as a whole. This would test the interaction of many functions but, do confine the test within one module.

• Integrated Testing: Brings all the modules together into a special testing environment, then checks for errors, bugs and interoperability. It deals with tests for the entire application. Application limits and features are tested here.

Modifications and Improvements: Once the students finish the testing they are bound to be faced with bugs, errors and they will need to modify your source code to improve the system. Define what modification are implemented in the system and how it improved the system.

Chapter 6: Results and Discussion

Test Reports: Explain the test results and reports based on the test cases, which should show that the project is capable of facing any problematic situation and that it works fine in different conditions. Take the different sample inputs and show the outputs.

User Documentation: Define the working of the software; explain its different functions, components with screen shots. The user document should provide all the details of the product in such a way that any user reading the manual, is able to understand the working and functionality of the document.

Chapter 7: Conclusions

Conclusion: The conclusions can be summarised in a fairly short chapter (2 or 3 pages). This chapter brings together many of the points that would have made in the other chapters.

Limitations of the System: Explain the limitations encountered during the testing of the project that the students were not able to modify. List the criticisms accepted during the demonstrations of the project.

Future Scope of the Project describes two things: firstly, new areas of investigation prompted by developments in this project, and secondly, parts of the current work that was not completed due to time constraints and/or problems encountered.

REFERENCES

It is very important that the students acknowledge the work of others that they have used or adapted in their own work, or that provides the essential background or context to the project. The use of references is the standard way to do this. Please follow the given standard for the references for books, journals, and online material. The citation is mandatory in both the reports. E.g:

Linhares, A., & Brum, P. (2007). Understanding our understanding of strategic scenarios: What role do chunks play? *Cognitive Science*, *31*(6), 989-1007. https://doi.org/doi:10.1080/03640210701703725

Lipson, Charles (2011). Cite right : A quick guide to citation styles; MLA, APA, Chicago, the sciences, professions, and more (2nd ed.). Chicago [u.a.]: University of Chicago Press. p. 187.

ISBN 9780226484648.

Elaine Ritchie, J Knite. (2001). *Artificial Intelligence, Chapter 2 , p.p 23 - 44.* Tata McGrawHill.

GLOSSARY

If you the students any acronyms, abbreviations, symbols, or uncommon terms in the project report then their meaning should be explained where they first occur. If they go on to use any of them extensively then it is helpful to list them in this section and define the meaning.

APPENDICES

These may be provided to include further details of results, mathematical derivations, certain illustrative parts of the program code (e.g., class interfaces), user documentation etc.

In particular, if there are technical details of the work done that might be useful to others who wish to build on this work, but that are not sufficiently important to the project as a whole to justify being discussed in the main body of the project, then they should be included as appendices.

VI. SUMMARY

Project development usually involves an engineering approach to the design and development of a software system that fulfils a practical need. Projects also often form an important focus for discussion at interviews with future employers as they provide a detailed example of what the students are capable of achieving. In this course the students can choose your project topic from the lists given in Unit 4: Category-wise Problem Definition.

VII. FURTHER READINGS

1. Modern Systems Analysis and Design; Jeffrey A. Hoffer, Joey F. George, Joseph,S. Valacich; Pearson Education; Third Edition; 2002.

2. ISO/IEC 12207: Software Life Cycle Process
   (http://www.software.org/quagmire/descriptions/iso-iec12207.asp).

3. IEEE 1063: Software User Documentation (http://ieeexplore.ieee.org).

4. ISO/IEC: 18019: Guidelines for the Design and Preparation of User Documentation for Application Software.

5. http://www.sce.carleton.ca/squall.

6. http://en.tldp.org/HOWTO/Software-Release-Practice-HOWTO/documentation.html.   7. http://www.sei.cmu.edu/cmm/

PROFORMA FOR THE APPROVAL PROJECT PROPOSAL

(Note:All entries of the proforma of approval should be filled up with appropriate and complete information. Incomplete proforma of approval in any respect will be summarily rejected.)

PNR **No.:** ……………………   Roll no**: _____**

1. Name of the Student

   _____

2. Title of the Project

   _____

3. Name of the Guide

   _____

4. Teaching experience of the Guide _____

5. Is this your first submission?   ☐   Yes   No
   ☐

Signature of the Student   Signature of the Guide

Date: …………………   Date: ……………………

Signature of the Coordinator

Date: …………………

(All the text in the report should be in times new roman)

TITLE OF THE PROJECT
(NOT EXCEEDING 2 LINES, 24 BOLD, ALL CAPS)

A Project Report (12 Bold)
Submitted in partial fulfillment of the
Requirements for the award of the Degree of (size-12)

BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY)(14 BOLD, CAPS)

By(12 Bold)

Name of The Student (size-15, title case)
Seat Number (size-15)

Under the esteemed guidance of (13 bold)
Mr./Mrs. Name of The Guide (15 bold, title case)
Designation (14 Bold, title case)

COLLEGE LOGO

DEPARTMENT OF INFORMATION TECHNOLOGY(12 BOLD, CAPS)
COLLEGE NAME (14 BOLD, CAPS)
(Affiliated to University of Mumbai) (12, Title case, bold, italic)
CITY, PIN CODE(12 bold, CAPS) MAHARASHTRA (12 bold, CAPS) YEAR (12 bold)
COLLEGE NAME (14 BOLD, CAPS)
(Affiliated to University of Mumbai) (13, bold, italic)
CITY-MAHARASHTRA-PINCODE(13 bold, CAPS)

DEPARTMENT OF INFORMATION TECHNOLOGY (14 BOLD, CAPS)

College Logo

CERTIFICATE (14 BOLD, CAPS, underlined, centered)

This is to certify that the project entitled, **"Title of The Project "**, is bonafied work of **NAME OF THE STUDENT** bearing Seat.No: (**NUMBER**) submitted in partial fulfillment of the requirements for the award of degree of BACHELOR OF SCIENCE in INFORMATION

TECHNOLOGY from University of Mumbai. (12, times new roman, justified)

Internal Guide (12 bold)   **Coordinator**

(Don't write names of lecturers or HOD)

                        External Examiner

Date: **College Seal**

COMPANY CERTIFICATE (if applicable)

(Project Abstract page format)

Abstract (20bold, caps, centered)

Content (12, justified)

Note: Entire document should be with <u>1.5line spacing and all paragraphs should start with 1 tab space.</u>

ACKNOWLEDGEMENT

(20, BOLD, ALL CAPS, CENTERED)

The acknowledgement should be in times new roman,   12 font with 1.5 line spacing, justified.

DECLARATION (20 bold, centered, allcaps)

Content (12, justified)

I here by declare that the project entitled, "**Title of the Project**" done at **place where the project is done**, has not been in any case duplicated to submit to any other university for the award of any degree. To the best of my knowledge other than me, no one has submitted to any other university.

The project is done in partial fulfillment of the requirements for the award of degree of

BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY) tobe submitted as final semester project as part of our curriculum.

Name and Signature of the Student

TABLE OF CONTENTS (20bold, caps, centered)

Should be generated automatically using word processing software.

List of Tables (20 bold, centered, Title Case)

Should be generated automatically using word processing software.

List of Figures (20 bold, centered, Title Case)

Should be generated automatically using word processing software.

Chapter 1

Introduction (20 Bold, centered)

Content or text (12, justified)

Note: Introduction has to cover brief description of the project with minimum 4 pages.

Chapter 2

System Analysis (20 bold, Centered)

Subheadings are as shown below with following format (16 bold, CAPS)

2.1 Existing System (16 Bold)
2.1.1 ------------ (14 bold, title case)
2.1.1.1 ------------ (12 bold, title case)

2.2 Proposed System 2.3 Requirement Analysis 2.4 Hardware Requirements 2.5 Software Requirements
2.6 Justification of Platform – (how h/w & s/w satisfying the project)

Table 2.1: Caption

Chapter 3

System Design (20 bold, centered)

Subheadings are as shown below with following format (16 bold, CAPS)
Specify figures as Fig 11.1 – caption

3.1Module Division 3.2Data Dictionary 3.3E-R Diagrams
3.4Data Flow Diagrams / UML

Note: write brief description at the bottom of all diagrams

Sample Figure

Fig. 3.1: Caption

Chapter 4

Implementation and Testing (20 bold, centered)

4.1Code (Place Core segments)

Content includes description about coding phase in your project (Font-12)

(* don't include complete code-----just description)

4.2Testing Approach Subheadings are as shown below with following format (16 bold, CAPS)

4.2.1 Unit Testing
4.2.2 Integration Testing

Note:

○ Explain about above testing methods
○ Explain how the above techniques are applied in your project

Provide Test plans, test cases, etc relevant to your project

Chapter 5 Results and Discussions(20 bold, centered)

Note: Place Screen Shots and write the functionality of each screen at the bottom

Conclusion and Future Work (20 bold, centered)

The conclusions can be summarized in a fairly short chapter around 300 words. Also include limitations of your system and future scope (12, justified)

Chapter 7
References (20 bold, centered)

Content (12, LEFT)

[1] Title of the book, Author

[2] Full URL of online references

[3] ------------

*NOTE ABOUT PROJECT VIVA VOCE:

Student may be asked to write code for problem during VIVA to demonstrate his coding capabilities and he/she may be asked to write any segment of coding used in the in the project. The project can be done in group of at most four students. However, the length and depth of the project should be justified for the projects done in group. A big project can be modularised and different modules can be assigned as separate project to different students.

Marks Distribution:

Semester VI: 150 Marks

Documentationand Viva Voce: 50 Marks:

Implementation: 100 Marks

The plagiarism should be maintained as per the UGC guidelines.

Practical Exam: 50 marks

Note: A Certified copy of the journal is essential to appear for the practical examination.

1.	**Practical Question 1**	**20**
2.	**Practical Question 2**	**20**
3.	**Journal**	**5**
4.	**Viva Voce**	**5**

O
R

1.	**Practical Question**	**40**
2.	**Journal**	**5**
3.	**Viva Voce**	**5**

SOPHIA COLLEGE AUTONOMOUS

MUMBAI – 400 026

SEMESTER (V/VI) (2018 - 19) INTERNAL ASSESSM Date:
EXAM

SUBJECT NAME

SUBJECT CODE

Class: T.Y. B.Sc. – I.T   **Marks:**

20

Instructions:

1. All questions are compulsory.   **(20)**

2. Figures to the right indicate full marks.

3. Draw neat diagrams wherever necessary.

Q.1 Answer the following questions (any four out of six)

1.

2.

3.

4.

5.

6.

Note: 5 marks for the attendance.

Class: T.Y.B.Sc. – I.T
Marks: 75

15

Instructions:
    1. All questions are compulsory.
    2. Figures to the right indicate full marks.
    3. Draw neat diagrams wherever necessary.

15

Q. 1 Attempt any three out of six

15

Q. 2 Attempt any three out of six

15

Q. 3 Attempt any three out of six

15

Q. 4 Attempt any three out of six

Q. 5 Attempt any three out of six