SIES COLLEGE OF COMMERCE & ECONOMICS
AUTONOMOUS
UG DEPARTMENT OF INFORMATION TECHNOLOGY

Date of BOS meeting: 5th April 2025

Name of BOS Chairperson: Mrs. Bhavini Deepak Shah

Sr. No.	Heading	Particulars
1	Title of the course	B. Sc. (Information Technology)
2	Eligibility for admission	HSC or Equivalent with Mathematics as Compulsory Subject
3	Minimum percentage	45 %
4	Semesters	V & VI
5	Level	UG
6	Pattern	03 years & 06 semesters CBGS
7	To be implemented from	From Academic year 2025-26 in a progressive manner

SIES COLLEGE OF COMMERCE & ECONOMICS
(AUTONOMOUS)
(Affiliated to University of Mumbai)
RE-ACCREDITED GRADE "A" BY NAAC

BOARD OF STUDIES
UG DEPARTMENT OF INFORMATION TECHNOLOGY

(WITH EFFECT FROM THE ACADEMIC YEAR 2025-2026)

TYBSc (IT)
Semester V

Semester V			
Course Code	**Course Type**	**Course Title**	**Credits**
BSIT-MJS5-101	Major	Next Generation Technologies	3
BSIT-MJPS5-101	Major Practical	Next Generation Technologies Lab	1
BSIT-MJS5-102	Major	Software Engineering & Project Management	3
BSIT-MJPS5-102	Major Practical	Software Engineering & Project Management Lab	1
BSIT-MJS5-103	Major	Linux System Administration	2
BSIT-MJELS5-104	Major Elective	Advanced Web Programming	3
BSIT-MJELPS5-104	Major Elective Practical	Advanced Web Programming Lab	1
BSIT-MJELS5-105	Major Elective	Enterprise Java	3
BSIT-MJELPS5-105	Major Elective Practical	Enterprise Java Lab	1
BSIT-MNS5-106	Minor	Internet of Things	3
BSIT-MNPS5-106	Minor Practical	Internet of Things Lab	1
BSIT-SECS5-107	Skill Enhancement Course (SEC)	IT Skills Enhancement Lab	2
BSIT-FPS5-108	Field Project	Field Project	2
		Total Credits	22

B. Sc. (Information Technology)		Semester – V	
Course Name: Next Generation Technologies		Course Code: BSIT-MJS5-101	
Periods per week (1 Period is 60 minutes)		3	
Credits		3	
		Hours	Marks
Evaluation System	Theory Examination	2	50
	Internal	--	25

Course Objectives:

- The learner will be able to describe the structure and components of MongoDB database, jQuery and JSON effectively.
- The learner will be able to explain and demonstrate the fundamental concepts required for new age technologies like big data, NoSQL etc.
- The learner will be able to understand fundamental concepts in MongoDB such as architecture, election process, and storage structure.
- The learner will be able to apply different MongoDB techniques such as creating collections, documents, index structures, backup and restore.
- The learner will be able to select the concepts in order to solve real world problems using MongoDB, jQuery and JSON.
- The learner will be able to compare MongoDB with other new age technologies, old age storage with new age SSDs.
- The learner will be able to design different kinds of applications using MongoDB, jQuery and JSON.

Sr. No	Modules/Units	No of Lectures
1.	**Big Data:** Getting Started, Big Data, Facts About Big Data, Big Data Sources, Three Vs of Big Data, Volume, Variety, Velocity, Usage of Big Data, Visibility, Discover and Analyze Information, Segmentation and Customizations, Aiding Decision Making, Innovation, Big Data Challenges, Policies and Procedures, Access to Data, Technology and Techniques, Legacy Systems and Big Data, Structure of Big Data, Data Storage, Data Processing, Big Data Technologies. **NoSQL:** SQL, NoSQL, Definition, A Brief History of NoSQL, ACID vs. BASE, CAP Theorem (Brewer's Theorem), The BASE, NoSQL Advantages and Disadvantages, Advantages of NoSQL, Disadvantages of NoSQL, SQL vs. NoSQL Databases, Categories of NoSQL Databases. **Introducing MongoDB:** History, MongoDB Design Philosophy, Speed, Scalability, and Agility, Non-Relational Approach, JSON-Based Document Store, Performance vs. Features, Running the Database Anywhere, SQL Comparison **The MongoDB Data Model:** The Data Model, JSON and BSON, The Identifier (_id), Capped Collection, Polymorphic Schemas, Object Oriented Programming,	15

	Schema Evolution. **Using MongoDB Shell:** Basic Querying, Create and Insert, Explicitly Creating Collections, Inserting Documents Using Loop, Inserting by Explicitly Specifying _id, Update, Delete, Read, Using Indexes, Stepping Beyond the Basics, Using Conditional Operators, Regular Expressions, MapReduce, aggregate(), Designing an Application's Data Model, Relational Data Modeling and Normalization, MongoDB Document Data Model Approach.	
2.	**MongoDB Architecture:** Core Processes, mongod, mongo, mongos, MongoDB Tools, Standalone Deployment, Replication, Master/Slave Replication, Replica Set, Implementing Advanced Clustering with Replica Sets, Sharding, Sharding Components, Data Distribution Process, Data Balancing Process, Operations, Implementing Sharding, Controlling Collection Distribution (Tag-Based Sharding),Points to Remember When Importing Data in a Sharded Environment, Monitoring for Sharding, Monitoring the Config Servers, Production Cluster Architecture, Scenario 1,Scenario 2,Scenario 3,Scenario 4. **MongoDB Storage Engine:** Data Storage Engine, Data File (Relevant for MMAPv1), Namespace (.ns File), Data File (Relevant for WiredTiger), Reads and Writes, How Data Is Written Using Journaling, GridFS – The MongoDB File System, The Rationale of GridFS, GridFSunder the Hood, Using GridFS, Indexing, Types of Indexes, Behaviors and Limitations. **MongoDB Limitations:** MongoDB Space Is Too Large (Applicable for MMAPv1), Memory Issues (Applicable for Storage Engine MMAPv1), 32-bit vs. 64-bit, BSON Documents, Namespaces Limits, Indexes Limit, Capped Collections Limit - Maximum Number of Documents in a Capped Collection, Sharding Limitations, Shard Early to Avoid Any Issues, Shard Key Can't Be Updated, Shard Collection Limit, Select the Correct Shard Key, Security Limitations, No Authentication by Default, Traffic to and from MongoDB Isn't Encrypted, Write and Read Limitations, CaseSensitive Queries, Type-Sensitive Fields, No JOIN, Transactions, MongoDB Not Applicable Range.	15
3.	**The End of Disk? SSD and In-Memory Databases:** The End of Disk?, Solid State Disk, The Economics of Disk, SSD-Enabled Databases, InMemory Databases, TimesTen, Redis, SAP HANA, VoltDB, Oracle 12c in-Memory Database, Berkeley Analytics Data Stack and Spark, Spark Architecture. **jQuery:** Introduction, Traversing the DOM, DOM Manipulation with jQuery, Events, Ajax with jQuery, jQuery Plug-ins, jQuery Image Slider. **JSON:** Introduction, JSON Grammar, JSON Values, JSON Tokens, Syntax, JSON vs XML, Data Types, Objects, Arrays, Creating JSON, JSON Object, Parsing JSON, Persisting JSON, Data Interchange, JSON PHP, JSON HTML, JSONP.	15

REFERENCE BOOKS:

Sr. No.	Title	Authors	Publisher	Edition	Year
1.	Practical MongoDB	Shakuntala Gupta Edward Navin Sabharwal	Apress	-	-
2.	Beginning jQuery	Jack Franklin Russ Ferguson	Apress	2nd	-
3.	Next Generation Databases	Guy Harrison	Apress	-	-
4.	Beginning JSON	Ben Smith	Apress	-	-

B. Sc. (Information Technology)		Semester – V	
Course Name: Next Generation Technologies Lab		Course Code: BSIT-MJPS5-101	
Periods per week (1 Period is 60 minutes)		2	
Credits		1	
		Hours	Marks
Evaluation System	Practical Examination	--	25

Course Objectives:

- To make students learn NoSQL with document-oriented database, MongoDB.
- To make students understand jQuery and JSON features to efficiently develop web pages and their functionality.

Practical No	Details
1	**MongoDB Basics**
a	Write a MongoDB query to create and drop database.
b	Write a MongoDB query to create, display and drop collection.
c	Write a MongoDB query to insert, query, update and delete a document.
2	**Simple Queries with MongoDB**
3	**Implementing Aggregation**
a	Write a MongoDB query to use sum, avg, min and max expression.
b	Write a MongoDB query to use push and addToSet expression.
c	Write a MongoDB query to use first and last expression.
4	**Replication, Backup and Restore**
a	Write a MongoDB query to create Replica of existing database.
b	Write a MongoDB query to create a backup of existing database.
c	Write a MongoDB query to restore database from the backup.
5	**Java and MongoDB**
a	Connecting Java with MongoDB and inserting, retrieving, updating anddeleting.
6	**PHP and MongoDB**
a	Connecting PHP with MongoDB and inserting, retrieving, updating anddeleting.
7	**Python and MongoDB**
a	Connecting Python with MongoDB and inserting, retrieving, updating anddeleting.
8	**Programs on Basic jQuery**
a	jQuery Basic, jQuery Events
b	jQuery Selectors, jQuery Hide and Show effects
c	jQuery fading effects, jQuery Sliding effects

9	**jQuery Advanced**
a	jQuery Animation effects, jQuery Chaining
b	jQuery Callback, jQuery Get and Set Contents
c	jQuery Insert Content, jQuery Remove Elements and Attribute
10	**JSON**
a	Creating JSON
b	Parsing JSON
c	Persisting JSON
11	**Create a JSON file and import it to MongoDB**
a	Export MongoDB to JSON.
b	Write a MongoDB query to delete JSON object from MongoDB

SCHEME OF EXAMINATION FOR:
Subject : Next Generation Technologies

The scheme of examination shall be divided into three parts:

- Internal Assessment - 25 marks
- Semester End Examination - 50 marks
- Practical Assessment - 25 marks

Internal Assessment 25 marks

Description	Marks
Internal test (online/offline)(Objective/Subjective)	20
Assignments/ group discussions/ debates/ quiz/ open book test/ book review/presentation/ viva/ any other	5
Total	**25**

Semester end Examination 50 marks (paper pattern)

Duration : 2 hour Total Marks: 50	
Q.1 10 marks (from Unit 1)	10
Q.2 10 marks (from Unit 2)	10
Q.3 10 marks (from Unit 3)	10
Q.4 20 marks (from all Units)	20
Note: The 10/20 marks full length question may be sub divided into 2/4 questions of 5 marks each	

Semester end Practical Examination 25 marks

Description	Marks
Practical examination	20
Viva and Journal	05
Total	**25**

Passing criteria: Minimum 40% (10 out of 25) in Internal, 40% (20 out of 50) in semester end and 40% (10 out of 25) in practical examination.

B. Sc. (Information Technology)		**Semester – V**	
Course Name: Software Engineering & Project Management		**Course Code: BSIT-MJS5-102**	
Periods per week (1 Period is 60 minutes)		**3**	
Credits		**3**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	**2**	**50**
	Internal	**--**	**25**

Course Objectives:

- Students will be provided with the knowledge of basic Software engineering methods and practices, and their appropriate application.
- Students will be provided with demonstrating and decomposing competence in communication, planning, analysis, design, construction and deployment.
- To provide an idea of using various process models in the software industry according to given circumstances.
- To gain the knowledge of how Analysis, Design, Implementation, Testing and Maintenance processes are conducted in a software project.
- Students will be given a general understanding of software process models such as the waterfall and evolutionary models.
- To make the students understand software requirements, SRS, BRS documents.

Sr.No	Modules/Units	No of Lectures
1.	**Introduction:** What is software engineering? Software Development Life Cycle. **Software Requirements:** Functional and Non-functional requirements, User Requirements, System Requirements, Interface Specification, Documentation of the software requirements. **SoftwareProcesses:** Process and Project, Component Software Processes. **Software Development Process Models**. • Waterfall Model. • Prototyping. • Iterative Development. • Rational Unified Process. • The RAD Model • Time boxing Model. **Agile software development:** Agile methods, Plan-driven and agiledevelopment,	15

	Extreme programming, Agile project management, Scaling agile methods. **Socio-technical system:** Essential characteristics of socio technical systems, Emergent System Properties, Systems Engineering, Components of system such as organization, people and computers, Dealing Legacy Systems. **Critical system:** Types of critical system, A simple safety critical system, Dependability of a system, Availability and Reliability, Safety and Security of Software systems. **Requirements Engineering Processes:** Feasibility study, Requirements elicitation and analysis, Requirements Validations, Requirements Management. **System Models:** Models and its types.	
2.	**Architectural Design:** Architectural Design Decisions, System Organisation, Modular Decomposition Styles, Control Styles, Reference Architectures. **User Interface Design:** Need of UI design, Design issues, The UI design Process, User analysis, User Interface Prototyping, InterfaceEvaluation. **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. **Cost Estimation:** Software Productivity, Estimation Techniques, Algorithmic Cost Modelling, Project Duration and Staffing. **The Software Team, Team Structures.** **Process Improvement:** Process and product quality, Process Classification, Process Measurement, Process Analysis and Modelling, Process Change, The CMMI Process Improvement Framework.	15

3.	**Introduction to Software Project Management:** Introduction, Why is Software Project Management Important? What is a Project? Software Projects versus Other Types of Project, 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. **An Overview of Project Planning**: Introduction to Step Wise Project Planning, Step 0: Select Project, Step 1: Identify Project Scope and Objectives, Step 2: Identify Project Infrastructure, Step 3: Analyse Project Characteristics, Step 4: Identify Project Products and Activities, Step 5: Estimate Effort for Each Activity, Step 6: Identify Activity Risks, Step 7: Allocate Resources, Step 8: Review/Publicize Plan, Step 9 and 10: Execute Plan/Lower Levels of Planning. **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. **Managing Contracts:** Introduction, Types of Contract, Stages in Contract Placement, Typical Terms of a Contract, Contract Management, Acceptance. **Project Closeout**: Introduction, Reasons for Project Closure, Project Closure Process, Performing a Financial Closure, Project Closeout Report	15

REFERENCE BOOKS:

Sr. No.	Title	Authors	Publisher	Edition	Year
1.	Software Engineering	Ian Somerville	Pearson Education	9th	-
2.	Software Engineering	Pankaj Jalote	Narosa Publication	-	-
3.	Software engineering, a practitioners approach	Roger Pressman	Tata McGraw Hill	7th	-
4.	Software Project Management	Bob Hughes, Mike Cotterell, Rajib Mall	TMH	6th	2018
5.	Project Management and Tools & Technologies – An overview	Shailesh Mehta	SPD	1st	2017
6.	Software Project Management	Walker Royce	Pearson	-	2005

B. Sc. (Information Technology)		**Semester – IV**	
Course Name: Software Engineering & Project Management Lab		**Course Code: BSIT-MJPS5-102**	
Periods per week (1 Period is 60 minutes)		**2**	
Credits		**1**	
		Hours	**Marks**
Evaluation System	**Practical Examination**	**--**	**25**

Course Objectives:

- Students will be able to understand different UML diagrams.
- To enable students to draw UML diagrams for developing software.

List of Practical (To be executed using Star UML or any similar software)	
0.	**Write down the problem statement for a suggested system of relevance.**
	Perform requirement analysis. Develop Requirement specification for suggested system
1.	Study and implementation of Class Diagrams.
2.	Study and implementation of Use Case Diagrams.
3.	Study and implementation of Entity Relationship 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.	Software Project Management Role play scenarios.

REFERENCE BOOKS:

Sr. No.	Title	Authors	Publisher	Year
1.	Object - Oriented Modelingand Design	Michael Blaha, James Rumbaugh	Pearson	2011
2.	Learning UML 2. 0	Kim Hamilton,RussMiles	O'Reilly Media	2006
3.	The unified modelinglanguage user guide	Grady Booch, James Rumbaugh, Ivar Jacobson	Addison Wesley	2005
4.	UML A Beginners Guide	Jason T. Roff	McGraw Hill Professional	2003

SCHEME OF EXAMINATION FOR:
Subject : Software Engineering & Project Management

The scheme of examination shall be divided into three parts:

- Internal Assessment - 25 marks
- Semester End Examination - 50 marks
- Practical Assessment - 25 marks

Internal Assessment 25 marks

Description	Marks
Internal test (online/offline)(Objective/Subjective)	20
Assignments/ group discussions/ debates/ quiz/ open book test/ book review/presentation/ viva/ any other	5
Total	**25**

Semester end Examination 50 marks (paper pattern)

Duration : 2 hour Total Marks: 50	
Q.1 10 marks (from Unit 1)	10
Q.2 10 marks (from Unit 2)	10
Q.3 10 marks (from Unit 3)	10
Q.4 20 marks (from all Units)	20
Note: The 10/20 marks full length question may be sub divided into 2/4 questions of 5 marks each	

Semester end Practical Examination 25 marks

Description	Marks
Practical examination	20
Viva and Journal	05
Total	**25**

Passing criteria: Minimum 40% (10 out of 25) in Internal, 40% (20 out of 50) in semester end and 40% (10 out of 25) in practical examination.

B. Sc. (Information Technology)		Semester – V	
Course Name: Linux System Administration		Course Code: BSIT-MJS5-103	
Periods per week (1 Period is 60 minutes)		2	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	30
	Internal	--	20

Course Objectives:

- Learners will be able to understand the role and responsibilities of a Linux system administrator.
- Learners will be able to install and configure the Linux operating system on virtual machines.
- Learners will be able to make effective use of Linux utilities, and scripting languages.

Sr. No	Modules/Units	No of Lectures
1	**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 **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	10
2	**Connecting to the Network:** Understanding Network Manager, Working with Services and Run levels, Configuring the Network with Network Manager, Working with system-config-network, Network Manager 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, Configuring Key-Based SSH Authentication, Using SSH Port Forwarding. **Working with Users, Groups, and Permissions:** Managing Users and Groups,	10

	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, Working with Access Control Lists. **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, 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.	
3	**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. **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 **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. **High-Availability Clustering** **Setting Up an Installation Server**	10

REFERENCE BOOKS:

Sr. No.	Title	Authors	Publisher	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	-
3.	Linux Administration: A Beginner's Guide	Wale Soyinka	TMH	-

UG DEPARTMENT OF INFORMATION TECHNOLOGY

SCHEME OF EXAMINATION

SCHEME OF EXAMINATION FOR:
Subject : Linux System Administration

The scheme of examination shall be divided into two parts:

- **Internal assessment - 20 marks**
- **Semester end examination - 30 marks**

Internal Assessment 20 marks

Description	Marks
Objective type test (for 10 marks – online/offline)	10
Any one of the following Presentation / Assignment / Online course / Case Study / Open Book Test	10
Total	20

Semester end Examination 30 marks (paper pattern)

Description	Marks
Q.1 10 marks (Attempt any 2 from 4 questions of 5 marks each)	**10**
Q.2 10 marks (Attempt any 2 from 4 questions of 5 marks each)	**10**
Q.3 10 marks (Attempt any 2 from 4 questions of 5 marks each)	**10**
Total	**30**

Passing criteria: Minimum 40% (8 out of 20) in Internal, 40% (12 out of 30) in semester end examination.

B. Sc. (Information Technology)		Semester – V	
Course Name: Advanced Web Programming		Course Code: BSIT-MJELS5-104	
Periods per week (1 Period is 60 minutes)		3	
Credits		3	
		Hours	Marks
Evaluation System	Theory Examination	2	50
	Internal	--	25

Course Objectives:

- The Learners will be able to describe the working of .NET framework.
- The Learners will be able to explain and demonstrate how to create dynamic Web pages using web forms and code behind file.
- The Learners will be able to use advanced controls such as validation controls, navigation controls, master pages, styles, themes
- The Learners will be able to connect the web applications using SqlDataSource with GridView, DetailsView and FormView controls.
- The Learners will be able to compare different mechanisms and controls and choose a concept that fits the problem description.
- The Learners will be able to develop web applications using a combination of client-side and server-side technologies

Sr. No	Modules/Units	No of Lectures
1.	**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, 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.	15
2.	**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, User Controls. Website Navigation: Site Maps, The SiteMapPath Control, The TreeView Control, The Menu Control. **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,	15

3.	**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, **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 **ASP.NET AJAX:** Understanding Ajax, Using Partial Refreshes, Using Progress Notification, Implementing Timed Refreshes, Working with the ASP.NET AJAX Control Toolkit. **Introduction of MEAN and MERN**	15

REFERENCE BOOKS:

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 McGrawHill 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.
1. https://www.geeksforgeeks.org/

B. Sc. (Information Technology)		**Semester – V**	
Course Name: Advanced Web Programming Lab		**Course Code**: **BSIT-MJELPS5-104**	
Periods per week (1 Period is 60 minutes)		**2**	
Credits		**1**	
		Hours	**Marks**
Evaluation System	**Practical Examination**	**--**	**25**

Course Objectives:

- To assist students to design and deploy a dynamic data driven web applications in ASP.NET.
- To make students understand the debugging techniques using traces for faster application development.

Practical No	**Details**
C# Programs	
1	**a.** Create an application to print on screen the output of adding, subtracting, multiplying, and dividing two numbers entered by the user in C#.
	b. Create a simple application to demonstrate the concepts boxing and unboxing.
	c. Create an application to print Floyd's triangle till n rows in C#.
	d. Create a simple application to perform addition and subtraction using delegate
	e Create a simple application to demonstrate use of the concepts of interfaces.
ASP.NET with C# Programs	
2	Create an application to demonstrate following operations i. Generate Fibonacci series. ii. Test for prime numbers.
3	a.Create a simple web page with various server 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 calendarcontrol c) Selected day in a calendar control using style d) Difference between two calendar dates
	c. Demonstrate the use of Treeview operations on the web form
4	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	a.Create Web Form to demonstrate use of Website Navigation controls.
	b. Create a web application to demonstrate use of Master Page and content page
	c. Create a web application to demonstrate various states of ASP.NET Pages
6	a. Create a web application for inserting and deleting records from a database
	b. Create a web application to display Using Disconnected Data Access and Databinding using GridView

7	Create a web application to demonstrate the use of different types of Cookies.
8	a.Create a web application for inserting and deleting records from a database. (Using ExecuteNon Query).
	b. Create a web application for user defined exception handling.
9	a. Create a web application to demonstrate use of GridView button column and GridView events along with paging and sorting.
	b. Create a web application to demonstrate data binding using DetailsView and FormView Control
	c. Create a web application to demonstrate use of various Ajax controls

UG DEPARTMENT OF INFORMATION TECHNOLOGY

SCHEME OF EXAMINATION

SCHEME OF EXAMINATION FOR:

Subject : Advanced Web Programming

The scheme of examination shall be divided into three parts:

- Internal Assessment - 25 marks
- Semester End Examination - 50 marks
- Practical Assessment - 25 marks

Internal Assessment 25 marks

Description	Marks
Internal test (online/offline)(Objective/Subjective)	20
Assignments/ group discussions/ debates/ quiz/ open book test/ book review/presentation/ viva/ any other	5
Total	**25**

Semester end Examination 50 marks (paper pattern)

Duration : 2 hour Total Marks: 50	
Q.1 10 marks (from Unit 1)	10
Q.2 10 marks (from Unit 2)	10
Q.3 10 marks (from Unit 3)	10
Q.4 20 marks (from all Units)	20
Note: The 10/20 marks full length question may be sub divided into 2/4 questions of 5 marks each	

Semester end Practical Examination 25 marks

Description	Marks
Practical examination	20
Viva and Journal	05
Total	**25**

Passing criteria: Minimum 40% (10 out of 25) in Internal, 40% (20 out of 50) in semester end and 40% (10 out of 25) in practical examination.

B. Sc. (Information Technology)		Semester – V	
Course Name: Enterprise Java		Course Code: BSIT-MJELS5-105	
Periods per week (1 Period is 60 minutes)		3	
Credits		3	
		Hours	Marks
Evaluation System	Theory Examination	2	50
	Internal	--	25

Course Objectives:

- Learners will be able to understand how to create dynamic web pages using enterprise java technologies like servlets, jsp etc.
- Learners will be able to analyze Java EE technologies and will be to create a website using it.

Sr. No	Modules/Units	No. of Lectures
1.	**Understanding Java EE:** What is an Enterprise Application? What is java enterprise edition? Java EE Technologies, Java EE evolution, Glassfishserver **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. **RequestDispatcher:** 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 HttpSession, 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.	15
2.	**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 **Introduction To Java Server Pages:** Why use Java Server Pages? Disadvantages Of JSP, JSP v\s Servlets, Lifecyle 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	15

	Conventions, Unified Expression Language [UnifiedEl], Expression Language. **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.	
3.	**Introduction To Enterprise Javabeans:** Enterprise Bean Architecture, Benefits of Enterprise Bean, Types of Enterprise Bean, Accessing Enterprise Beans, Enterprise Bean Application, Packaging Enterprise Beans **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. **Introduction to Java Persistence API:** The Java Persistence API, JPA, ORM, Database and the Application, Architecture of JPA, How JPA Works? JPA Specifications. **Introduction to Hibernate:** What is Hibernate? Why Hibernate? Hibernate, Database and The Application, Components of Hibernate, Architecture of Hibernate, How Hibernate Works?	15

REFERENCE BOOKS:

1. Java EE 7 For Beginners, Sharman Shah, Vaishali Shah, SPD, First Edition,2017.
2. Java EE 8 Cookbook: Build reliable applications with the most robust and mature technology for enterprise development, Elder Moraes, Packt, First edition ,2018.
3. Advanced Java Programming, Uttam Kumar Roy, Oxford Press,2015

B. Sc. (Information Technology)		Semester – V	
Course Name: Enterprise Java Lab		Course Code: BSIT-MJELPS5-105	
Periods per week (1 Period is 60 minutes)		2	
Credits		1	
		Hours	Marks
Evaluation System	Practical Examination	--	25

Course Objectives:

- Learners will be able to develop an enterprise level application using Java EE technologies.

NOTE: Practical's can be implemented using NetBeans IDE or Eclipse IDE

List of Practical	
1.	**Implement** 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. (Name-txt, 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 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.

UG DEPARTMENT OF INFORMATION TECHNOLOGY

SCHEME OF EXAMINATION

SCHEME OF EXAMINATION FOR:
Subject : Enterprise Java

The scheme of examination shall be divided into three parts:

- Internal Assessment - 25 marks
- Semester End Examination - 50 marks
- Practical Assessment - 25 marks

Internal Assessment 25 marks

Description	Marks
Internal test (online/offline)(Objective/Subjective)	20
Assignments/ group discussions/ debates/ quiz/ open book test/ book review/presentation/ viva/ any other	5
Total	**25**

Semester end Examination 50 marks (paper pattern)

Duration : 2 hour Total Marks: 50	
Q.1 10 marks (from Unit 1)	10
Q.2 10 marks (from Unit 2)	10
Q.3 10 marks (from Unit 3)	10
Q.4 20 marks (from all Units)	20
Note: The 10/20 marks full length question may be sub divided into 2/4 questions of 5 marks each	

Semester end Practical Examination 25 marks

Description	Marks
Practical examination	20
Viva and Journal	05
Total	**25**

Passing criteria: Minimum 40% (10 out of 25) in Internal, 40% (20 out of 50) in semester end and 40% (10 out of 25) in practical examination.

B. Sc. (Information Technology)		**Semester – V**	
Course Name: Internet of Things		**Course Code: BSIT-MNS5-106**	
Periods per week (1 Period is 60 minutes)		**3**	
Credits		**3**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	**2**	**50**
	Internal	**--**	**25**

Course Objectives:

- The learner will be able to describe the architecture of Internet of Things.
- The learner will be able to summarize different protocols used in IoT.
- The learner will be able to apply various programming techniques of IoT.
- The learner will be able to analyse various platforms used in IoT.
- The learner will be able to summarize prototyping models for IoT.
- The learner will be able to integrate ethics of IoT technology with mass manufacturing of the IoT devices.

Sr. No	Modules/Units	No of Lectures
1.	**The Internet of Things:** An Overview` **Design Principles for Connected Devices:** Calm and Ambient Technology, Magic as Metaphor, Privacy, Keeping Secrets, Whose Data Is It Anyway? Small Pieces, Loosely Joined, Graceful Degradation, Affordances. **IoT Network Architecture, The "Things" in IoT:** IoT World Forum(IoTWF) standardized Architecture, Sensors, Actuators, MEMS, Smart Objects, Sensor Networks **Internet Principles:** IP, TCP, UDP, IP Addresses, DNS, IPv6, MAC Addresses, TCP and UDP Ports, An Example: HTTP Ports, HTTP, HTTPS: Encrypted HTTP, CoAP, MQTT.	15
2.	**Thinking About Prototyping:** Sketching, Familiarity, Costs versus Ease of Prototyping, Prototypes and Production, Changing Embedded Platform, Physical Prototypes and Mass Personalization, climbing into the Cloud, Open Source versus Closed Source, Mixing Open and ClosedSource. **Prototyping Embedded Devices:** ScalingUp the Electronics, Embedded Computing Basics,Microcontrollers, System-on-Chips, Choosing Your Platform, Introduction to AVR microcontroller, Arduino, developing on the Arduino, Openness, Raspberry Pi,Developing on the Raspberry Pi, Openness. Getting Started with Arduino: Introduction, Arduino Variants, Install the Drivers, Arduino IDE Basic Functions: Overview, Structure, Digital I/O Functions, Analog I/O Functions, Advanced I/O Functions, Timer Functions, Communication Functions, Interrupt Functions, Math Functions, Programming Language Reference Using Sensors with the Arduino: Light Sensitive Sensors, Temperature Sensors, Temperature and Humidity Sensor, Ultrasonic Sensors, Sensor, Joystick Module, Gas Sensor,	15

	Prototyping the Physical Design: Preparation, Sketch, Iterate, and Explore, 3D Printing, Types of 3D Printing, Software, CNC Milling, Repurposing/Recycling.	
3.	**Prototyping Online Components:** Getting Started with an API, Mashing Up APIs, Scraping, Legalities, writing a New API, Clockodillo, Security, implementing the API. **Techniques for Writing Embedded Code:** Memory Management, Types of Memory, Making the Most of Your RAM, Performance and Battery Life, Libraries, Debugging. **Business Models:** The Business Model Canvas, Funding an Internet of Things Startup, Hobby Projects and Open Source, Venture Capital, Government Funding, Crowdfunding, Lean Startups. **Moving to Manufacture:** What Are You Producing? Designing Kits, Designing Printed circuit boards, Software Choices, The Design Process, Etching Boards, Milling Boards. Assembly, Testing, 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, Cautious Optimism, The Open Internet of Things Definition.	15

REFERENCE BOOKS:

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
5. IoT Fundamentals: Networking Technologies, Protocols, and use cases for Internet of Things, Pearson , David Hanes, Gonzalo Salgueiro, Patrick rossetete, Rob Barton, Jarome Henry.
6. Arduino_ A Technical Reference_ A Handbook for Technicians, Engineers, and Makers,J. M. Hughes, O'Reilly Media, 2016.

B. Sc. (Information Technology)		**Semester – V**	
Course Name: Internet of Things Lab		**Course Code: BSIT-MNPS5-106**	
Periods per week (1 Period is 60 minutes)		**2**	
Credits Internet of Things		**1**	
		Hours	**Marks**
Evaluation System	**Practical Examination**	**--**	**25**

Course Objectives:

- Students will be able to build IoT applications in Raspberry Pi kit.
- Students will learn the use of Internet in Mobile Devices, Cloud & SensorNetworks.
- Students will be able to study and understand the application of different sensors using Arduino(TinkerCAD).

Practicals to be done online using TinkerCAD and Raspberry Pi kit.

Practical No	Details
0	Introduction to Arduino Introduction to Arduino circuits and breadboarding Blinking of LEDs. Introduction to Raspberry Pi.
1	Program using Light Sensitive Sensors
2	Program using temperature sensors
3	Programs using humidity sensors
4	Programs using Ultrasonic Sensors
5	Programs using gas sensors
6	Programs making Joystick with Arduino
7	Displaying different LED patterns with Raspberry Pi.
8	Displaying Time over 4-Digit 7-Segment Display using Raspberry Pi
9	Visitor Monitoring with Raspberry Pi and Pi Camera
10	Installing Windows 10 IoT Core on Raspberry Pi

UG DEPARTMENT OF INFORMATION TECHNOLOGY

SCHEME OF EXAMINATION

SCHEME OF EXAMINATION FOR:

Subject : Internet of Things

The scheme of examination shall be divided into three parts:

- Internal Assessment - 25 marks
- Semester End Examination - 50 marks
- Practical Assessment - 25 marks

Internal Assessment 25 marks

Description	Marks
Internal test (online/offline)(Objective/Subjective)	20
Assignments/ group discussions/ debates/ quiz/ open book test/ book review/presentation/ viva/ any other	5
Total	**25**

Semester end Examination 50 marks (paper pattern)

Duration : 2 hour Total Marks: 50	
Q.1 10 marks (from Unit 1)	10
Q.2 10 marks (from Unit 2)	10
Q.3 10 marks (from Unit 3)	10
Q.4 20 marks (from all Units)	20
Note: The 10/20 marks full length question may be sub divided into 2/4 questions of 5 marks each	

Semester end Practical Examination 25 marks

Description	Marks
Practical examination	20
Viva and Journal	05
Total	**25**

Passing criteria: Minimum 40% (10 out of 25) in Internal, 40% (20 out of 50) in semester end and 40% (10 out of 25) in practical examination.

B. Sc. (Information Technology)		Semester – V	
Course Name: IT Skills Enhancement Lab		Course Code: BSIT-SECS5-107	
Periods per week (1 Period is 60 minutes)		4	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	--	50

Course Objectives:

- Learners will be able to develop mobile applications.
- Learners will be able to learn the basics of Android platform and get to understand the application lifecycle.
- Learners will be able to install the Linux operating system and configure peripherals.
- Learners will be able to configure and maintain basic networking services.
- Learners will be able to create shell scripts.

SECTION I: Mobile App Development

Practical No	Details
1.	Introduction to Android, Introduction to Android Studio IDE, Application Fundamentals: Creating a Project, Android Components, Activities, Services, Content Providers, Broadcast Receivers, Interface overview, Creating Android Virtual device, USB debugging mode, Android Application Overview. Simple "Hello World" program.
2.	Programming Resources Android Resources: (Color, Theme, String, Drawable, Dimension, Image),
3.	Programming Activities and fragments Activity Life Cycle, Activity methods, Multiple Activities, Life Cycle of fragments and multiple fragments
4.	Programs related to different Layouts Coordinate, Linear, Relative, Table, Absolute, Frame, List View, Grid View.
5.	Programming UI elements AppBar, Fragments, UI Components
6.	Programming menus, dialog, dialog fragments
7.	Programs on Intents, Events, Listeners and Adapters The Android Intent Class, Using Events and Event Listeners
8.	Programs on Services, notification and broadcast receivers
9.	Database Programming with SQLite
10.	Programming threads, handles and asynchronized programs

11.	Programming Media API and Telephone API
12.	Programming Security and permissions
13.	Programming Network Communications and Services (JSON)

REFERENCE BOOKS:

1. Flutter for Beginners Alessandro Biessek Packt Publishing 2019.

2. PhoneGap By Example Andrey Kovalenko PACKT Publishing 1 st 2015.

SECTION II: Linux System Administration

List of Practical	
1.	**Installation of RHEL 6.X, Configuring Booting with GRUB.**
2.	**Graphical User Interface and Command Line Interface and Processes**
a.	Exploring the Graphical Desktop
b.	The Command Line Interface
c.	Managing Processes
3.	**Storage Devices and Links, Backup and Repository**
a.	Working with Storage Devices and Links
b.	Making a Backup
c.	Creating a Repository
4.	**Working with RPMs Storage and Networking**
a.	Using Query Options
b.	Extracting Files from RPMs
c.	Configuring and Managing Storage
d.	Connecting to the Network
5.	**Working with Users, Groups, and Permissions**
6.	**Firewall and Cryptographic services**
a.	Securing Server with iptables
b.	Setting Up Cryptographic Services
7.	**Configuring Server for File Sharing**
a.	Configuring NFS Server and Client
b.	Configuring Samba
c.	Configuring FTP
8.	**DNS, DHCP and Mail Server**

a.	Configuring DNS
b.	Configuring DHCP
c.	Setting Up a Mail Server
9.	**Web Server**
a.	Configuring Apache on Red Hat Enterprise Linux
b.	Writing a Script to Monitor Activity on the Apache Web Server
10.	**Shell Scripts**
a.	Use of echo command, Use of comment, Using While loop, Using For loop, Get User Input, using if statement, Using Case Statement.
b.	Get Arguments from Command Line, Combine String variables, Get substring of String, create function, Create function with Parameters.
c.	Read a File, delete a File, Append to File, Send Email, Wait Command, Sleep Command

UG DEPARTMENT OF INFORMATION TECHNOLOGY

SCHEME OF EXAMINATION

SCHEME OF EXAMINATION FOR:
Subject : IT Skills Enhancement Lab

The scheme of examination shall be:

- Practical Assessment 50 marks
  Semester end Practical Examination 50 marks

Description	Marks
Practical examination	40
Viva	05
Journal	05
Total	**50**

Passing criteria: Minimum 40% (20 out of 50) in practical examination.

B. Sc. (Information Technology)		Semester – V	
Course Name: Field Project		Course Code: BSIT-FPS5-108	
Periods per week (1 Period is 60 minutes)		-	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	--	50

Course Objectives:

- The learner will be able to recall and describe key concepts, tools, and technologies used in project development and fieldwork.
- The learner will be able to explain the fundamental principles and methodologies of project management, software development, and problem-solving within the context of real-world projects.
- The learner will be able to apply theoretical knowledge to practical, real-life problems.
- The learner will be able to design and develop live or application-based projects using appropriate tools and technologies.
- The learner will be able to analyze project requirements, identify challenges, and evaluate the effectiveness of different technologies and approaches for solving complex problems.
- The learner will be able to document technical processes, present project outcomes effectively, and collaborate with faculty and/or industry mentors for feedback and improvement.

B.Sc. (IT) for Software development, group project by two students per group

Credits and duration:

1. FP will carry weightage of two credits.
2. Each student is required to complete minimum of 2-3 field visits/ project submission and presentation.
3. The FP program is to be completed during respective Semester, before the prescribed date of submission

Project (Dissertation) Report: (as per the format prescribed by Faculty Mentor)

1. To be submitted in the prescribed format (Hard Copy)
2. Obtain the signature of faculty mentor
3. Upload the Project Report on drive (Link provided by Faculty Mentor)

Students are required to submit a report of the field project at the end of the semester in following suggested format.

All projects should be typed on ***A4 sheets, Font Size 12, Times New Roman, one and a half spacing.*** The project report shall have appropriate chapter scheme and be presented in minimum of 20 pages.

Report should be arranged in the following manner.

Title Page

- Title of the Report (Font size 14)
- Name of the Student
- Roll number/Seat number
- Program Title
- Name of the Mentor
- Date of Submission

Table of contents

- Include headings and subheadings with page numbers.

List of Figures and Tables

- List all figures and tables included in the report with corresponding page numbers.

NOTE: Refer ANNEXURE I for report format

UG DEPARTMENT OF INFORMATION TECHNOLOGY
SCHEME OF EXAMINATION

SCHEME OF EXAMINATION FOR:
Field Project with 2 credits

The scheme of examination shall be:

- Field Project 50 marks

  Semester end Examination 50 marks

Description	Marks
Project Documentation	30
Presentation and Viva	20
Total	**50**

Students should submit the certificate from the company.

Passing criteria: Minimum 40% (20 out of 50) in practical examination.

SEMESTER VI

Semester VI			
Course Code	**Course Type**	**Course Title**	**Credits**
BSIT-MJS6-101	Major	Business Intelligence	3
BSIT-MJPS6-101	Major Practical	Business Intelligence Lab	1
BSIT-MJS6-102	Major	Software Testing and Quality Assurance	3
BSIT-MJPS6-102	Major Practical	Software Testing & Quality Assurance Lab	1
BSIT-MJS6-103	Major	Geographic Information Systems	2
BSIT-MJELS6-104	Major Elective	Security in Computing	3
BSIT-MJELPS6-104	Major Elective Practical	Security in Computing Lab	1
BSIT-MJELS6-105	Major Elective	Ethical Hacking	3
BSIT-MJELPS6-105	Major Elective Practical	Ethical Hacking Lab	1
BSIT-MNS6-106	Minor	Artificial Intelligence	3
BSIT-MNPS6-106	Minor Practical	Artificial Intelligence Lab	1
BSIT-OJTS6-107	OJT	On Job Training	4
		Total Credits	22

B. Sc. (Information Technology)		**Semester – VI**	
Course Name: Business Intelligence		**Course Code: BSIT-MJS6-101**	
Periods per week (1 Period is 60 minutes)		**3**	
Credits		**3**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	**2**	**50**
	Internal	**--**	**25**

Course Objectives:

- Students will get a solid understanding of BI concepts, including data warehousing, ETL (Extract, Transform, Load) processes, and the role of BI in supporting business decision-making.
- Students will learn to identify the major frameworks decision support systems (DSS) and business intelligence (BI).
- Students will learn the foundations, definitions, architecture and capabilities of DSS and BI.
- Students will learn to apply BI knowledge to real-world business problems, improving their ability to make data-driven decisions and contribute to strategic business initiatives.

Sr.No	Modules/Units	No of Lectures
1.	**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 **Mathematical models for decision making:** Structure of mathematical models, Development of a model, Classes of models	15
2.	**Data mining:** Definition of data mining, Representation of input data ,Data mining process, Analysis methodologies **Data preparation**: Data validation, Data transformation, Data reduction **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	15

3.	**Business intelligence applications:** **Marketing models:** Relational marketing, Sales force management **Knowledge Management:** Introduction to Knowledge Management,Organizational Learning and Transformation, Knowledge Management Activities, Approaches to Knowledge Management, InformationTechnology (IT) In Knowledge Management, Knowledge ManagementSystems 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	15

REFERENCE BOOKS:

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: Business Intelligence Lab		Course Code: BSIT-MJPS6-101	
Periods per week (1 Period is 60 minutes)		2	
Credits		1	
		Hours	Marks
Evaluation System	Practical Examination	--	25

Course Objectives:

- Enable students to clean, prepare, and transform raw data for analysis using tools like Power BI, Excel, and R.
- Develop students' ability to create insightful visualizations and interactive dashboards to represent business data effectively.
- Equip students with the skills to design and generate reports and dashboards that support data-driven decision-making.
- Teach students how to perform comprehensive data analysis using Excel and R, including summarizing and manipulating data.
- Help students understand and apply time-based data analysis to track and forecast business performance using Power BI and R.

No	Details
1	Power BI Practical: Create Interactive Dashboards with Power BI 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.) Create a set of visualizations: bar charts, pie charts, and line charts to represent sales performance.
2	Perform the Extraction Transformation and Loading (ETL) process to construct the database in the Power BI.
3	Perform Data Cleaning and Transformation in Excel
4	Perform Text Analytics and Sentiment Analysis with Python
5	Create Pivot Tables for Data Analysis in Excel.
6	Perform Sales Forecasting Using Time Series Analysis in R
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.

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

UG DEPARTMENT OF INFORMATION TECHNOLOGY

SCHEME OF EXAMINATION

SCHEME OF EXAMINATION FOR:
Subject : Business Intelligence

The scheme of examination shall be divided into three parts:

- Internal Assessment - 25 marks
- Semester End Examination - 50 marks
- Practical Assessment - 25 marks

Internal Assessment 25 marks

Description	Marks
Internal test (online/offline)(Objective/Subjective)	20
Assignments/ group discussions/ debates/ quiz/ open book test/ book review/presentation/ viva/ any other	5
Total	**25**

Semester end Examination 50 marks (paper pattern)

Duration : 2 hour Total Marks: 50	
Q.1 10 marks (from Unit 1)	10
Q.2 10 marks (from Unit 2)	10
Q.3 10 marks (from Unit 3)	10
Q.4 20 marks (from all Units)	20
Note: The 10/20 marks full length question may be sub divided into 2/4 questions of 5 marks each	

Semester end Practical Examination 25 marks

Description	Marks
Practical examination	20
Viva and Journal	05
Total	**25**

Passing criteria: Minimum 40% (10 out of 25) in Internal, 40% (20 out of 50) in semester end and 40% (10 out of 25) in practical examination.

B. Sc. (Information Technology)		Semester – VI	
Course Name: Software Testing and Quality Assurance		Course Code: BSIT-MJS6-102	
Periods per week (1 Period is 60 minutes)		3	
Credits		3	
		Hours	Marks
Evaluation System	Theory Examination	2	50
	Internal	--	25

Course Objectives:

- Understand the significance of software testing in ensuring software quality and reliability.
- Familiarize with the fundamental concepts and principles of software quality assurance.
- Learn different software testing techniques and methodologies for effective test case design.
- Explore the role of verification and validation in software development and testing processes.
- Gain practical experience in using software testing tools and frameworks for automated testing

Sr. No	Modules/Units	No of Lectures
1.	**Introduction to Software Testing and Quality Assurance** **Introduction to Software Testing:** Nature of errors and the need for testing **Definition of Quality and Quality Assurance:** Understanding quality in software development, Pillars of Quality Management System, Quality Control (QC), Quality Management (QM), and Software Quality Assurance (SQA), Total Quality Management (TQM) **Software Development Life Cycle (SDLC):** Overview of SDLC phases and their relationship to testing, Role of testing in each phase, Software quality factors and their impact on testing **Verification and Validation (V&V):** Definition of V&V and its significance in software development, Different types of V&V mechanisms, Concepts of Software Reviews, Inspection, and Walkthrough, Verification Workbench	15
2.	**Software Testing Techniques and Strategies** **Testing Fundamentals:** Basics of software testing process, Test case design principles and techniques, Test execution, reporting, and documentation **White Box Testing and Black Box Testing:** Functional/Specification based Testing as Black Box, Black box: Equivalence Partitioning, Boundary Value Analysis, Decision Table Testing, State Transition Testing. Structural Testing as White Box, White Box: Statement testing, Branch testing. Experience-based: Error guessing, Exploratory testing, Checklist-based testing. **Software Testing Strategies:** Strategic approach to software testing Unit Testing: purpose, techniques, and best practices, Integration Testing: approaches and challenges, Validation Testing: ensuring adherence to user requirements, System Testing: comprehensive end-to-end testing **Special Tests:** GUI testing, Regression Testing, Manual Support Testing, Smoke Testing, Stress Testing, Recovery Testing, Installation Testing **Software Metrics:** Concept of software metrics and their importance, Developing and utilizing different types of metrics, Complexity metrics and their significance in testing	15
3.	**Defect Management and Software Quality Assurance** **Defect Management:** Definition of defects and their lifecycle, Defect management process, including defect reporting and tracking, Metrics related to defects and their	15

	utilization for process improvement **Software Quality Assurance:** Understanding quality concepts and the Quality Movement: Background issues and challenges in SQA, Activities and approaches in Software Quality Assurance, Software Reviews: Formal Technical Reviews and their benefits, Statistical Quality Assurance and Software Reliability , Requirement Traceability Matrix, PDCA, Continual (Continuous) Improvement Cycle **Statistical process control techniques for quality assurance:** Software reliability measurement and improvement, The ISO 9000 standards, ISO 9126 Standards. **Quality Improvement Techniques:** Introduction to quality improvement methodologies, Utilizing quality costs for decision-making, Introduction to quality improvement tools: Pareto Diagrams, Cause-effect Diagrams, Scatter Diagrams, Run charts, Gantt Chart	

REFERENCE BOOKS:

Sr No.	Title	Author	Publisher
1.	Software Engineering for Students, A Programming Approach	Douglas Bell	4th Edition, Pearson Education, 2005
2.	Software Engineering – A Practitioners Approach	Roger S. Pressman	7th Edition, Tata McGraw Hill
3.	Quality Management	Donna C. S. Summers	5th Edition, Prentice-Hall
4.	Software Testing and Quality Assurance Theory and Practice	Kshirsagar Naik, Priyadarshi Tripathy	John Wiley & Sons, Inc. , Publication.
5.	Software Testing and Continuous Quality Improvement	William E. Lewis	CRC Press, Third edition, 2016

B. Sc. (Information Technology)		Semester – VI	
Course Name: Software Testing & Quality Assurance Lab		Course Code: BSIT-MJPS6-102	
Periods per week (1 Period is 60 minutes)		2	
Credits		1	
		Hours	Marks
Evaluation System	Practical Examination	--	25

Course Objectives:

- Gain practical experience in using software testing tools and frameworks for automated testing

Practical No	Details
1.	Install Selenium IDE and create a test suite containing a minimum of 4 test cases for different web page formats (e.g., HTML, XML, JSON, etc.).
2.	Conduct a test suite for two different websites using Selenium IDE. Perform various actions like clicking links, filling forms, and verifying content.
3.	Install Selenium Server (Selenium RC) and demonstrate its usage by executing a script in Java or PHP to automate browser actions.
4.	Write a program using Selenium WebDriver to automate the login process on a specific web page. Verify successful login with appropriate assertions.
5.	Write a program using Selenium WebDriver to update 10 student records in an Excel file. Perform data manipulation and verification.
6.	Write a program using Selenium WebDriver to select the number of students who have scored more than 60 in any one subject (or all subjects). Perform data extraction and analysis.
7.	Write a program using Selenium WebDriver to provide the total number of objects present or available on a web page. Perform object identification and counting.
8.	Write a program using Selenium WebDriver to get the number of items in a list or combo box on a web page. Perform element identification and counting.
9.	Write a program using Selenium WebDriver to count the number of checkboxes on a web page, including checked and unchecked counts. Perform checkbox identification and counting.
10.	Perform load testing on a web application using JMeter. Generate and analyze load scenarios. Additionally, explore bug tracking using Bugzilla as a tool for logging and tracking software defects.
11.	**Case Studies**

UG DEPARTMENT OF INFORMATION TECHNOLOGY
SCHEME OF EXAMINATION

SCHEME OF EXAMINATION FOR:
Subject : Software Testing & Quality Assurance

The scheme of examination shall be divided into three parts:

- Internal Assessment - 25 marks
- Semester End Examination - 50 marks
- Practical Assessment - 25 marks

Internal Assessment 25 marks

Description	Marks
Internal test (online/offline)(Objective/Subjective)	20
Assignments/ group discussions/ debates/ quiz/ open book test/ book review/presentation/ viva/ any other	5
Total	**25**

Semester end Examination 50 marks (paper pattern)

Duration : 2 hour Total Marks: 50	
Q.1 10 marks (from Unit 1)	10
Q.2 10 marks (from Unit 2)	10
Q.3 10 marks (from Unit 3)	10
Q.4 20 marks (from all Units)	20
Note: The 10/20 marks full length question may be sub divided into 2/4 questions of 5 marks each	

Semester end Practical Examination 25 marks

Description	Marks
Practical examination	20
Viva and Journal	05
Total	**25**

Passing criteria: Minimum 40% (10 out of 25) in Internal, 40% (20 out of 50) in semester end and 40% (10 out of 25) in practical examination.

B. Sc. (Information Technology)		**Semester – VI**	
Course Name: Geographic Information Systems		**Course Code: BSIT-MJS6-103**	
Periods per week (1 Period is 60 minutes)		**2**	
Credits		**2**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	**--**	**30**
	Internal	**--**	**20**

Course Objectives:

- Learner will be able to understand the fundamental concepts and principles of Geographic Information Systems (GIS).
- Learner will be able to develop skills in spatial data management, analysis, and visualization.
- Learner will be able to apply GIS techniques to solve real-world spatial problems.

Sr. No	Modules/Units	No of Lectures
1.	**Introduction to GIS** : Gentle Introduction to GIS, Computer Representations of Geographic Information, Components of GIS, Spatial and Attribute Data, Role of Metadata in GIS, GIS Data Models: Vector and Raster Models	10
2.	**Data Management and Processing :** Data Collection, Data Input Methods, Data Quality and Accuracy in GIS, Data Transformation Techniques, Data Conversion techniques, Georeferencing	10
3.	**Spatial Data Analysis and Visualization** : Querying GIS Data, Vector Data Analysis, Raster Data Analysis, Cartographic Principles: Symbols, Colors, and Design for Maps, Map Cosmetics, GIS Applications and future trends	10

REFERENCE BOOKS:

1. Introduction to Geographic Information Systems, Chang Kang-tsung (Karl), McGrawHill 3rd Edition 2013 7th Edition.
2. Principles of Geographic Information Systems An Introductory Text Book Editors: Otto Huisman and Rolf A. The International Institute of Geoinformation Science and Earth Observation Fourth 2009.
3. Principles of Geographic Information Systems, P.A Burrough and R.A.McDonnell, Oxford University Press, Third 1999.
4. Fundamentals of Spatial Information Systems, R.Laurini and D. Thompson, Academic Press,1994.
5. Fundamentals of Geographic Information Systems, Michael N.Demers, Wiley Publications Fourth, 2009.
6. GIS Fundamentals: A First Text on Geographic Information Systems, Paul Bolsatd XanEdu Publishing Inc 5th Edition.

UG DEPARTMENT OF INFORMATION TECHNOLOGY

SCHEME OF EXAMINATION

SCHEME OF EXAMINATION FOR:
Subject : Geographic Information Systems
The scheme of examination shall be divided into two parts:

- **Internal assessment - 20 marks**
- **Semester end examination - 30 marks**

Internal Assessment 20 marks

Description	Marks
Objective type test (for 10 marks – online/offline)	10
Any one of the following Presentation / Assignment / Online course / Case Study / Open Book Test	10
Total	20

Semester end Examination 30 marks (paper pattern)

Description	Marks
Q.1 10 marks (Attempt any 2 from 4 questions of 5 marks each)	**10**
Q.2 10 marks (Attempt any 2 from 4 questions of 5 marks each)	**10**
Q.3 10 marks (Attempt any 2 from 4 questions of 5 marks each)	**10**
Total	**30**

Passing criteria: Minimum 40% (8 out of 20) in Internal, 40% (12 out of 30) in semester end examination.

B. Sc. (Information Technology)		**Semester – VI**	
Course Name: Security in Computing		**Course Code: BSIT-MJELS6-104**	
Periods per week (1 Period is 60 minutes)		**3**	
Credits		**3**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	**2**	**50**
	Internal	**--**	**25**

Course Objectives:

- Learners will be able to understand the basic concepts related to security in field of computers and networking.
- Learners will be able to analyze packets in a network to detect various security related attacks.
- Learners will be equipped with the knowledge to secure IoT systems and leverage AI/ML technologies in cybersecurity.

Sr.No	Modules/Units	No of Lectures
1	**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. **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. **Firewalls:** Overview, The Evolution of Firewalls, Core Firewall Functions, Additional Firewall Capabilities, Firewall Design.	15
2	**Database Security:** General Database Security Concepts, Understanding Database Security Layers, Understanding Database- Level Security, Using Application Security, Database Backup and Recovery, Keeping Your Servers Up to Date, Database Auditing and Monitoring. **Secure Network Design:** Introduction to Secure Network Design, Performance, Availability, Security. **Network Device Security:** Switch and Router Basics, Network Hardening. **Wireless Network Security:** Radio Frequency Security Basics, Data-Link 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. **Intrusion Detection and Prevention Systems:** IDS Concepts, IDS Types and Detection Models, IDS Features, IDS Deployment Considerations, Security Information and Event Management (SIEM).	15

3	**Voice over IP (VoIP) and PBX Security:** Background, VoIP Components, VoIP Vulnerabilities and Countermeasures, PBX, TEM: Telecom Expense Management. **Virtual Machines and Cloud Computing:** Virtual Machines, Cloud Computing. **Secure Application Design:** Secure Development Lifecycle, Application Security Practices, Web Application Security, Client Application Security. **Physical Security:** Classification of Assets, Physical Vulnerability Assessment, Choosing Site Location for Security, Securing Assets: Locks and Entry Controls, Physical Intrusion Detection. **IoT Security:** - IoT Security Challenges and Risks, Secure IoT Architectures ,Common IoT Attacks (Botnets, DDoS, Firmware Exploits), Secure Communication in IoT Devices. **AI & ML in Security:** - Role of AI & ML in Cybersecurity, AI-driven Threat Detection, Deep Learning for Malware Detection, Adversarial Attacks on AI Models. **Digital Forensics**.	15

REFERENCE BOOKS:

Sr. No.	Title	Authors	Publisher	Edition	Year
1.	The Complete Reference: Information Security	Mark Rhodes Ousley	McGraw-Hill	2nd	2013
2.	Essential Cybersecurity Science	Josiah Dykstra	O'Reilly	5th	2017
3.	Principles of computer Security: CompTIA Security and Beyond	Wm Arthur Conklin, Greg White	McGraw-Hill	2nd	2010

B. Sc. (Information Technology)		Semester – VI	
Course Name: Security in Computing Lab		Course Code: BSIT-MJELPS6-104	
Periods per week (1 Period is 60 minutes)		2	
Credits		1	
		Hours	Marks
Evaluation System	Practical Examination	--	25

Course Objectives:

- Learners will be able to simulate and apply the working of various security protocols on a given topology.

List of Practical	
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

UG DEPARTMENT OF INFORMATION TECHNOLOGY
SCHEME OF EXAMINATION

SCHEME OF EXAMINATION FOR:
Subject : Security in Computing

The scheme of examination shall be divided into three parts:

- Internal Assessment - 25 marks
- Semester End Examination - 50 marks
- Practical Assessment - 25 marks

Internal Assessment 25 marks

Description	Marks
Internal test (online/offline)(Objective/Subjective)	20
Assignments/ group discussions/ debates/ quiz/ open book test/ book review/presentation/ viva/ any other	5
Total	**25**

Semester end Examination 50 marks (paper pattern)

Duration : 2 hour Total Marks: 50	
Q.1 10 marks (from Unit 1)	10
Q.2 10 marks (from Unit 2)	10
Q.3 10 marks (from Unit 3)	10
Q.4 20 marks (from all Units)	20
Note: The 10/20 marks full length question may be sub divided into 2/4 questions of 5 marks each	

Semester end Practical Examination 25 marks

Description	Marks
Practical examination	20
Viva and Journal	05
Total	**25**

Passing criteria: Minimum 40% (10 out of 25) in Internal, 40% (20 out of 50) in semester end and 40% (10 out of 25) in practical examination.

B. Sc. (Information Technology)		**Semester – VI**	
Course Name: Ethical Hacking		**Course Code: BSIT-MJELS6-105**	
Periods per week (1 Period is 60 minutes)		**3**	
Credits		**3**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	**2**	**50**
	Internal	**--**	**25**

Course Objectives:

- Learners will understand the terminology and concepts related to ethical hacking and penetration testing.
- Learners will explore various hacking technologies and the skills required to become an ethical hacker.
- Learners will learn the different phases involved in ethical hacking and the methodologies used in penetration testing.
- Learners will gain knowledge of common hacking techniques, such as footprinting, scanning, enumeration, and session hijacking.
- Learners will develop proficiency in identifying and exploiting vulnerabilities in web servers, web applications, and wireless networks.

Sr.No	Modules/Units	No of Lectures
1	**Introduction:** Terminology, Types of Hacking Technology, Five Stages of Ethical Hacking, Hacktivism, Hacker Classes, Skills Required for an Ethical Hacker, Vulnerability Research, Ways to Conduct Ethical Hacking. **Footprinting:** Definition, Information Gathering Methodology, Competitive Intelligence, DNS Enumeration, Whois and ARIN Lookups, Types of DNS Records, Use of Traceroute in Footprinting, E-Mail Tracking. **Social Engineering:** Common Types of Attacks, Countermeasures. **Scanning and Enumeration:** Port Scanning, Network Scanning, Vulnerability Scanning, CEH Scanning Methodology, Ping Sweep Techniques, Nmap Command Switches, SYN, Stealth, XMAS, NULL, IDLE, FIN Scans, Banner Grabbing and OS Fingerprinting, Anonymizers, HTTP Tunneling Techniques, IP Spoofing Techniques, Steps Involved in Enumeration.	15
2	**System Hacking:** Password-Cracking Techniques, Types of Passwords, Keyloggers and Other Spyware Technologies, Escalating Privileges, Rootkits, Steganography. **Sniffers:** Protocols Susceptible to Sniffing, Active and Passive Sniffing, ARP Poisoning, MAC Flooding, DNS Spoofing Techniques, Sniffing Countermeasures **Denial of Service:** Types of DoS Attacks, Working of DoS Attacks, BOTs/BOTNETs, "Smurf" Attack, "SYN" Flooding, DoS/DDoS Countermeasures **Session Hijacking:** Spoofing vs. Hijacking, Types, Sequence Prediction, Steps, Prevention **Hacking Web Servers:** Web Server Vulnerabilities, Attacks against Web Servers, Patch Management Techniques, Web Server Hardening	15
3	**Web Application Vulnerabilities:** Web Applications Working, Web Application Hacking, Web Application Threats, Google Hacking, Countermeasures **Web-Based Password Cracking Techniques:** Authentication Types, Password Crackers, Countermeasures **SQL Injection:** Steps, SQL Server Vulnerabilities, Countermeasures	15

	Buffer Overflows with types **Wireless Hacking:** WEP, WPA Authentication Mechanisms, and Cracking Techniques, Wireless Sniffers, Rogue Access Points, Wireless Hacking Techniques, Securing Wireless Networks **Penetration Testing Methodologies:** Methodologies, Steps, Automated Tools, Pen-Test Deliverables	

REFERENCE BOOKS:

Sr. No.	Title	Authors	Publisher
1.	CEH official Certfied Ethical Hacking Review Guide	Kimberly Graves	Wiley India Edition
2.	Certified Ethical Hacker	Michael Gregg,	Pearson Education
3.	Certified Ethical Hacker	Matt Walker	TMH

B. Sc. (Information Technology)		**Semester – VI**	
Course Name: Ethical Hacking Lab		**Course Code: BSIT-MJELPS6-105**	
Periods per week (1 Period is 60 minutes)		**2**	
Credits		**1**	
		Hours	**Marks**
Evaluation System	**Practical Examination**	**--**	**25**

Course Objectives:

- Learners will understand Information Gathering and Enumeration Techniques:
- Learners will be able to analyze and Perform Network Scanning
- Learners will be able to analyze Backdoors and Trojans
- Learners will be able to analyze and Exploit System and Network Vulnerabilities
- Learners will be able to understand and Mitigate Various Cyber Attacks

List of Practical	
1.	**Using the tools for whois, traceroute, email tracking, google hacking**
2.	**Using the tools for scanning network, IP fragmentation, war dialing countermeasures, SSL Proxy, Censorship circumvention**
3.	**Using NETBIOS Enumeration tool, SNMP Enumeration tool, LINUX/ UNIX. Enumeration tools, NTP Enumeration tool, DNS analyzing and enumeration tool**
4.	**Using System Hacking Tools**
5.	**Study of Backdoors and Trojan Tools**
6.	**Study of Sniffing Tools**
7.	**Study of Denial of Service attack tools**
8.	**Study of Hijacking Tools**
9.	**Study of Web Server Attack Tools**
10.	**Study of SQL Injection and Web Server Tools**
11.	**Study of Wireless Hacking Tools**
12.	**Using Cryptanalysis Tools**
13.	**Study of Different Security Tool**

UG DEPARTMENT OF INFORMATION TECHNOLOGY
SCHEME OF EXAMINATION

SCHEME OF EXAMINATION FOR:
Subject : Ethical Hacking

The scheme of examination shall be divided into three parts:

- Internal Assessment - 25 marks
- Semester End Examination - 50 marks
- Practical Assessment - 25 marks

Internal Assessment 25 marks

Description	Marks
Internal test (online/offline)(Objective/Subjective)	20
Assignments/ group discussions/ debates/ quiz/ open book test/ book review/presentation/ viva/ any other	5
Total	**25**

Semester end Examination 50 marks (paper pattern)

Duration : 2 hour Total Marks: 50	
Q.1 10 marks (from Unit 1)	10
Q.2 10 marks (from Unit 2)	10
Q.3 10 marks (from Unit 3)	10
Q.4 20 marks (from all Units)	20
Note: The 10/20 marks full length question may be sub divided into 2/4 questions of 5 marks each	

Semester end Practical Examination 25 marks

Description	Marks
Practical examination	20
Viva and Journal	05
Total	**25**

Passing criteria: Minimum 40% (10 out of 25) in Internal, 40% (20 out of 50) in semester end and 40% (10 out of 25) in practical examination.

B. Sc. (Information Technology)		**Semester – VI**	
Course Name: Artificial Intelligence		**Course Code: BSIT-MNS6-106**	
Periods per week (1 Period is 60 minutes)		**3**	
Credits		**3**	
		Hours	**Marks**
Evaluation System	**Theory Examination**	**2**	**50**
	Internal	**--**	**25**

Course Objectives:

- Learner will be able to understand the foundational principles of Artificial Intelligence.
- Learner will be able to gain proficiency in designing and implementing intelligent agents for problem-solving using search algorithms.
- Learner will be able to develop skills in logical reasoning and knowledge representation, including propositional and first-order logic.
- Learner will be able to learn and apply machine learning algorithms for classification, regression, and unsupervised learning tasks.
- Learner will be able to explore reinforcement learning techniques and probabilistic models to solve complex decision-making problems.

Sr. No	Modules/Units	No of Lectures
1.	**Introduction**: What is Artificial Intelligence? Foundations of AI, history, the state of art AI today. **Intelligent Agents**: agents and environment, good behavior, nature of environment, the structure of agents. **Solving Problems by Searching**: Problem solving agents, examples problems, searching for solutions, uninformed search, informed search strategies, heuristic functions.	15
2.	**Logical Agents**: Knowledge base agents, The Wumpus world, logic, propositional logic, propositional theorem proving, effective propositional model checking, agents based on propositional logic, First Order Logic **Knowledge Representation and Reasoning**: Knowledge Representation and different forms, Reasoning, Planning, Uncertainty in Knowledge Fuzzy Logic & Fuzzification **Machine Learning**: Forms of Learning, Parametric & Non-Parametric Models, Classification, Regression, Regularization, Decision Trees, SVM, Artificial Neural Networks, Ensemble Learning, Boosting, K-NN, Gradient Descent	15

3.	**Probabilistic models:** Statistical Learning, Learning with Complete Data, Naive Bayes Classifier, Learning with Hidden Variables: The EM Algorithm **Unsupervised Learning**: Concept of Unsupervised learning,, Association Rule Mining **Reinforcement learning**: Concept of Reinforcement learning, Q-Learning, Hidden Markov Model **Current Trends in AI**	15

REFERENCE BOOKS:

1. Artificial Intelligence: A Modern Approach, Stuart Russel and Peter Norvig, Pearson 3rd 2015
2. A First Course in Artificial Intelligence, Deepak Khemani, TMH, First 2017
3. Artificial Intelligence: A Rational Approach, Rahul Deva, Shroff publishers, 1st 2018
4. Artificial Intelligence Elaine Rich, Kevin Knight and Shivashankar Nair TMH 3rd 2009
5. Artificial Intelligence & Soft Computing for Beginners, Anandita Das, Bhattacharjee, SPD 1st 2013

B. Sc. (Information Technology)		**Semester – VI**	
Course Name: Artificial Intelligence Lab		**Course Code: BSIT-MNPS6-106**	
Periods per week (1 Period is 60 minutes)		**2**	
Credits		**1**	
		Hours	**Marks**
Evaluation System	**Practical Examination**		**25**

Course Objectives:

- Learner will be able to understand the foundational principles of Artificial Intelligence.
- Learner will be able to gain proficiency in designing and implementing intelligent agents for problem-solving using search algorithms.
- Learner will be able to develop skills in logical reasoning and knowledge representation, including propositional and first-order logic.
- Learner will be able to learn and apply machine learning algorithms for classification, regression, and unsupervised learning tasks.
- Learner will be able to explore reinforcement learning techniques and probabilistic models to solve complex decision-making problems.

Course Code	Course Title
1	Implement Breadth First Search and Depth First Search algorithms
2	Implement A* Search and AO* algorithms
	Write a program to solve a. 4-Queen / N-Queen problem b. tower of Hanoi problem. c. Hill climbing problem d. water jug problem. e. Missionaries and Cannibals problem. f. traveling salesman problem
3	Implement the Decision Tree Learning algorithm to build a decision tree for a given dataset.
4	Implement the Feed Forward Backpropagation algorithm to train a neural network.
5	Implement the SVM algorithm for binary classification.
6	Implement the Adaboost algorithm to create an ensemble of weak classifiers.
7	Implement the Naive Bayes' algorithm for classification.
8	Implement the K-NN algorithm for classification or regression.
9	Implement the Association Rule Mining algorithm (e.g., Apriori) to find frequent itemsets.
10	Demonstration and experimentation of OpenAI/TensorFlow/PyTorch Tools and libraries
11	Mini Project

REFERENCE BOOKS:

1. Artificial Intelligence: A Modern Approach, Stuart Russel and Peter Norvig, Pearson 3rd 2015
2. A First Course in Artificial Intelligence, Deepak Khemani, TMH, First 2017
3. Artificial Intelligence: A Rational Approach, Rahul Deva, Shroff publishers, 1st 2018
4. Artificial Intelligence Elaine Rich, Kevin Knight and Shivashankar Nair TMH 3rd 2009
5. Artificial Intelligence & Soft Computing for Beginners, Anandita Das, Bhattacharjee, SPD 1st 2013

UG DEPARTMENT OF INFORMATION TECHNOLOGY
SCHEME OF EXAMINATION

SCHEME OF EXAMINATION FOR:
Subject : Artificial Intelligence

The scheme of examination shall be divided into three parts:

- Internal Assessment - 25 marks
- Semester End Examination - 50 marks
- Practical Assessment - 25 marks

Internal Assessment 25 marks

Description	Marks
Internal test (online/offline)(Objective/Subjective)	20
Assignments/ group discussions/ debates/ quiz/ open book test/ book review/presentation/ viva/ any other	5
Total	**25**

Semester end Examination 50 marks (paper pattern)

Duration : 2 hour Total Marks: 50	
Q.1 10 marks (from Unit 1)	10
Q.2 10 marks (from Unit 2)	10
Q.3 10 marks (from Unit 3)	10
Q.4 20 marks (from all Units)	20
Note: The 10/20 marks full length question may be sub divided into 2/4 questions of 5 marks each	

Semester end Practical Examination 25 marks

Description	Marks
Practical examination	20
Viva and Journal	05
Total	**25**

Passing criteria: Minimum 40% (10 out of 25) in Internal, 40% (20 out of 50) in semester end and 40% (10 out of 25) in practical examination.

B. Sc. (Information Technology)		**Semester – VI**	
Course Name: On Job Training		**Course Code: BSIT-OJTS6-107**	
Periods per week (1 Period is 60 minutes)		**-**	
Credits		**4**	
		Hours	**Marks**
Evaluation System	**Practical Examination**		**100**

Course Objectives

1) To provide practical, hands-on experience that strengthens Student's skills and knowledge.
2) To build confidence and independence by applying learning in real work situations.
3) To develop professionalism, including proper work habits, communication, and responsibility.
4) To prepare for future employment through real workplace exposure and improved competency.

1. Introduction

On-the-Job Training (OJT) / Internship is a credit-bearing component designed to give students hands-on professional exposure and practical skill enhancement aligned with NEP 2020 requirements. Students must undertake a structured internship in a recognized organization to gain real-world experience and industry readiness.

2. Duration & Credit Requirements

- Students must complete a minimum of 120 hours of Internship / OJT / Articleship / Part-Time Job / Full-Time Job.
- Successfully completing 120 hours shall earn the student 4 academic credits.
- Internship in OWN family business and Self Employment is strictly NOT allowed. (Students must gain external, unbiased exposure and evaluation.)

3. Evaluation Procedure

After completing the internship:

1. Students must prepare a comprehensive Internship Report describing:
    - Learning outcomes
    - Tasks performed
    - Skills gained
    - Challenges and reflections
2. The report must be:
    - Signed by the Internship Supervisor (from the organization)
    - Countersigned by the Faculty Mentor
3. The College will evaluate internship performance based on:
    - Supervisor appraisal
    - Quality of report
    - Attendance log
    - Oral presentation / Viva (if applicable)

Passing Standards

- Minimum Grade D required in the internship component.
- A student is declared PASS only if:
    - All required documents are submitted, AND
    - The student completes evaluation / viva successfully.
    -

4. Mandatory Documents

The following documents must be submitted in the prescribed format:

1. Experience Certificate (minimum 120 hours)
2. Acknowledgement Letter confirming internship selection
3. Student Diary (Logbook) with daily tasks & attendance
4. Company Supervisor Evaluation form
5. College Supervisor Evaluation form
6. Student Feedback Form post-internship

Note: Missing ANY document will result in *incomplete internship* status.

5. Guidelines for the Internship Report

Each student must prepare a formal Project Report based on their internship experience.

A. Structure of the Report

The report must include:

1. **Executive Summary**
    - A concise overview of the entire internship experience.
2. **Company Introduction**
    - Background, products/services, structure, SWOT analysis.
3. **Statement & Objectives**
    - Mission, vision, and strategic direction of the organization.
4. **Role & Responsibilities During Internship**
    - Department assigned
    - Key tasks performed
    - Tools/software used
    - Supervisor-verified summary
5. **Challenges Faced**
    - Practical difficulties
    - How theoretical knowledge was applied
    - How challenges were resolved
6. **Conclusion & Recommendations**
    - Key learning and takeaways
    - Suggestions to improve alignment between theory & practice
7. **Appendices**
    - All mandatory documents listed earlier

6. Report Formatting Requirements

- **Font:** Times New Roman
- **Font Size:** 12 (content), 14 (titles)
- **Line Spacing:** 1.5 for content, 1.0 for tables
- **Paper:** A4
- **Margins:** Left 1.5 inch; Right/Top/Bottom 1 inch
- **Binding:** Spiral
- **Length: 10–15 pages**

8. Submission Requirements

1. **Hard Copy** of the report in prescribed format
2. **Signature of Faculty Mentor** is mandatory
3. **Soft Copy Upload** on Google Drive (link provided by mentor)
4. Late submissions may not be accepted unless approved by the department

NOTE: The formats for all required documents shall be provided by the teacher-in-charge and will also be available on the college website.

****

UG DEPARTMENT OF INFORMATION TECHNOLOGY

SCHEME OF EXAMINATION

SCHEME OF EXAMINATION FOR:

Project Implementation with 4 credits

The scheme of examination shall be:

- Project Implementation 100 marks

  Semester end Examination 100 marks

Description	Marks
Project Documentation	50
Project Implementation and Viva	50
Total	**100**

Please refer Appendix – I for further details.

Students should submit the certificate from the company, if applicable.

Passing criteria: Minimum 40% (40 out of 100).

APPENDIX – I

Field Project Semester V

Format:

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

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

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.

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 sub-problem 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 theproject 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.

I. 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.

II. 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/

TABLE OF CONTENTS (20bold, caps, centered)

Should be generated automatically using word processing software.

Chapter 4: Implementation and Testing

- 4.1 Code (Place Core segments)
- 4.2 Testing Approach
- 4.2.1 Unit Testing (Test cases and Test Results)
- 4.2.2 Integration System (Test cases and Test Results)

Chapter 5: Results and Discussions (Output Screens)

Chapter 6: Conclusion and Future Work

Chapter 7: References

List of Tables (20 bold, centered, Title Cases

Should be generated automatically using word processing software.

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

Should be generated automatically using word processing software.

(Project Introduction page format)

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

3.1 Module Division

3.2 Data Dictionary

3.3 E-R Diagrams

3.4 Data Flow Diagrams / UML

Note: write brief description at the bottom of all diagrams

Fig. 3.1: Caption

Chapter 4

Implementation and Testing (20 bold, centered)

4.1 Code (Place Core segments)

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

don‟t include complete code just description)

4.2 Testing 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

Chapter 6

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] _____