

**Paper 1**  
**Data Communication, Networking & Security**

Note:

1. Students can be made familiar with practical aspects by conducting demo session.
2. Demo session can be either Physical/Audio-Video/Online Demo

**Unit I**

Introduction – Data Communication, Networks, Internet, Intranet, Protocols, OSI & TCP/IP Models Addressing.

Physical Layer – Signals, Analog, Digital, Analog VS Digital, Transmission impairment, Data Rate Limits, Performance.

Digital Transmission – Line Coding (Umipolar, Polar, Biphasic), Block Coding (4B/5B Encoding), Analog to digital conversion, PCM, Transmission Modes.

Analog Transmission – Digital to analog conversion (ASK, FSK, PSK, QAM), Analog to Analog conversion.

Multiplexing – FDM, WDM, Synchronous TDM (time slots & frames, interleaving, data rate management).

Spread Spectrum – FHSS, DSSS

Transmission Media – Guided and Unguided.

Switching – Switching, Circuit-Switched Networks, Datagram networks, Concept of Virtual circuit networks, structure of circuit and packet switch. Concepts of DSL and ADSL.

**References:**

1. Data Communication & Networking (Forouzan) – IV Edition → Chapters (1, 2, 3, 4, 5, 6, 7, 8, 9)

**Demo :**

1. Demo of installing NIC cards, Min. LAN Settings such as IP Address
2. Demo of various types of cables [if available], Cross cable and its use, Crimping
3. Study of lab network (type of network topology, bandwidth, switches)

**Unit II (30 lectures)**

Data Link Layer – Error correction & detection. Types of errors. Detection VS Correction, Block Coding, Hamming Distance, Linear Block codes (single parity check, hamming codes), Cyclic codes, CRC Encoder & Decoder, DRC Polynomial and its degree, Checksum.

\$\$\$ Link Control & Protocol – Framing, Flow & Error Control, Simplest. Stop-N- Wait. Stop-N-Wait ARQ, Go Back N ARQ, Selective Repeat ARQ. Piggybacking.

HDLC & PPP – HDLC Modes, HDLC Frames. PPP . PPP Transition states

Multiple Access – Random(CSMA), Controlled(Reservation, Polling, Token Passing), Channelization(FDMA, TDMA, CDMA) Wired LAN – LLC. MAC, Ethernet, Ethernet frame, Addressing, Concept of MBase, V Ethernet, Bridged, Switched, Full Duplex Ethernet, Concept of Fast and Gigabit Ethernet

Wireless LAN- Introduction to WLAN (Architecture Hidden Exposed Station Problem)  
Introduction to Bluetooth & Architecture, Cellular telephony, Concept of LG, 2G, 3G  
cellular telephony.

Connecting Devices Repeaters, Hubs, Bridges, Spanning tree algorithm. Two & Three  
layer Switches Routers, Gateways, Backbone networks, Concept of VLAN

Network Layer Logical addressing IPv4 Addresses & classless address, NAT Addressing.

**References –**

1. Data Communication & Networking (Forouzan ) = IV Edition – Chapter (10, 11, 12, 13, 14, 15, 16, 19)

**Demo :**

1. Study of network connectively devices [switches, modems/routers etc. installed in lab]

**Unit III (30 lectures)**

Network layer protocol – internetworking, IPv4, IPv4 protocol packet format, IPv6  
Protocol & Packet format, IPv4 VS IPv6, Transition from IPv4 to IPv6, Address Resolution  
protocols (ARP, RARP), BOOTP, DHCP,

Routing Protocols – Delivery, forwarding, routing, types of routing, routing tables,  
Unicast Routing, Unicast Routing protocols, RIP, Concepts of OSPF, BGP & Multicast  
Routing

Transport Layer – Process to process delivery, UCP, TCP

Congestion Control & Quality of Service – Data traffic, Congestion, Congestion Control  
(Open Loop, Closed Loop & Congestion control in TCP), QoS and Flow Characteristics

Application Layer – DNS, Remote Logging (Telnet), SMTP, FTP, WWW, HTTP

**References –**

1. Data Communication & Networking (Forouzan) – IV Edition

**Demo:**

1. Study of LAN Settings such as IP Addr, Subnet mask, Gateway Address, DNS addresses, Proxy etc.
2. Accessing machines in networks, sharing files/folders and printers.
3. Study of commands such as ping, netconfig, ipconfig, arp, netsta, rout, traceroute etc. [commands will depend on OS]
4. Useful Browser Settings.

**Unit IV (30 lectures)**

**Introduction :** Introduction to system and network security, security attacks, security services and mechanisms.

**Cryptography:** Traditional and Modern Symmetric-Key Ciphers, DES and AES, Asymmetric-Key Cryptography, RSA and ELGAMAL cryptosystems, Message Digest, Digital Signature, Key Management

**Network Security** : Security at Application Layer (E-MAIL, PGP and S/MIME), Security at Transport Layer (SSL and TLS), Security at Network Layer (IPSec).

**Firewall and Intrusion Detection** : Firewalls and their types, DMZ, Limitations of firewalls, Intruders, Intrusion detection (Host based, Networked, Distributed), IDS.

**Malicious software and Internet Security** : Viruses and related threats, virus countermeasures, denial of service attacks, Hacking, Security policies and plan, Strategies for a secure network.

**References:**

1. BF: "Cryptography & Network Security", Behrouz A. Forouzan, Tata McGraw-Hill. (1.1-1.4, 3.1-3.4, 5.1-5.2, 6.1-6.5, 7.1-7.6, 8.1, 8.2, 10.1-10.4, 11.1, 11.3, 13.1-12.5, 15.1-15.4, 16.1-16.3, 17.1-17.4, 18.1-18.4)
2. WS: "Network security essentials applications and standards", William Stallings, Edition, Pearson Education (1.1-1.6, 2.1-2.6, 3.1-3.5, 5.1-5.2, 3.1-6.4, 7.1-4.3, 8.1-8.2, 9.1-9.3, 10.1-10.3, 11.1-11.3)
3. AC: "Cryptography and Network Security", Atul Kahate, Tata McGrawHill.

**Demo:**

1. Antivirus software and its settings.
2. Settings, firewalls, Enabling/Disabling ports.
3. Introduction to cyber crime and cyber law.

**Additional References:**

1. Computer Networks and Internets (5<sup>th</sup> Edition), Douglas Comer
2. Computer Networks (4<sup>th</sup> Edition), Andrew Tanenbaum
3. Networking Complete by Sybex Inc. and Sybex Inc.

## **Paper II**

### **Advanced Java**

#### **Unit I: (30 lectures)**

Introduction to JFCs and Swing, Features of the Java Foundation Classes, Swing API Components, JComponent Class, Windows, Dialog Boxes, and Panels, Labels Buttons, Check Boxes, Menus, Toolbars, Implementing Action interface, Pane, JScrollPane, Desktop pane, Scrollbars, Lists and Combo Boxes, Text-Entry.

Components, Colors and File Choosers, Tables and Trees, Printing with 2D API and Java Print Service API.

JDBC Introduction, JDBC Architecture, Types of JDBC Drivers, The Connectivity Model, The java.sql package, Navigating the ResultSet object's contents, Manipulating records of a ResultSet object through User Interface, The JDBC Exception classes, Database Connectivity, Data Manipulation (using Prepared Statements, Joins, Transactions, Stored Procedures), Data navigation.

#### **References:**

Ch. 5, 7, of Ref. 6.

Ch. 10 of Ref. 5, Ch. 9 of Ref. 6, Ch. 4 of Ref. 2.

#### **Unit II: (30 lectures)**

Threads and Multithreading, The Lifecycle of a thread, Creating and running threads, Creating the Service Threads, Schedules Tasks using JVM, Thread-safe variables, Synchronizing threads, Communication between threads.

Overview of Networking, Working with URL, Connecting to a Server, Implementing Servers, Serving multiple Clients, Sending E-Mail, Socket Programming, Internet Addresses, URL Connections, Accessing Network interface parameters, Posting Form Data, Cookies, Overview of Understanding the Sockets Direct Protocol.

Introduction to distributed object system, Distributed Object Technologies, RMI for distributed computing, RMI Architecture, RMI Registry Service, Parameter Passing in Remote Methods, Creating RMI application, Steps involved in running the RMI application, Using RMI with Applets.

#### **References:**

Ch.3 of ref 2, Ch 11 of Ref 6, Ch 15 of Ref 5, Ref 7

Ch 13 of Ref 5, Ch 5 of Ref 2, Ch 11 of Ref 6

Ch 16 of Ref 5, Ch 25 of Ref 4, Ch 8 of Ref 6, Ch 8 of Ref 2, Ref 7

#### **Unit III: (30 lectures)**

What is a Servlet? The Example Servlets, Servlet Life Cycle, Sharing Information, Initializing a Servlet, Writing Service Methods, Filtering Requests and Responses, Invoking Other Web resources, Accessing the Web Context, Maintaining Client State, Finalizing Servlet.

What Is a JSP Page? The Example JSP Pages, The Life Cycle of a JSP Page, Creating Static Content, Creating Dynamic Content, Unified Expression Language, JavaBeans Components, JavaBeans Concepts, Using NetBeans GUI Builder Writing a Simple Bean,

Properties: Simple Properties, Using Custom tags, Reusing content in JSP Pages, Transferring Control to Another Web Component Including an Applet.

**References:**

Ch 3 of Ref 3, Ch 2, 3, 4 of Ref 1, Ref 7

Ch 4 of Ref 3, Ch 7, 8, 9 of Ref 1, Ref 7

**Unit IV : (30 lectures)**

Introduction to EJB, Benefits of EJB, Types of EJB, Session Bean: State Management Modes: Message-Driven Bean, Differences between Session Beans and Message-Driven Beans, Defining Client Access with Interfaces: Remote Access, Local Access, Local Interfaces and Container-Managed Relationship, Deciding on Remote or Local Access, Web Service Clients, Method Parameters and Access, The Contents of an Enterprise Bean, Naming Conventions for Enterprise Beans, The Life Cycles of Enterprise Beans, The Life Cycle of a Stateful Session Bean, The Life Cycle of a Stateless Session Bean, The Life Cycle of a Message-Driven Bean.

Building Web Services with JAX-WS: Setting the Port, Creating a Simple Web Service and Client with JAX-WS.

**References:**

Ch 20, Ref 3, Ch 14 of Ref 6, Ref 7

**References:**

1. Bryan Basham, Kathy Sierra, Bert Bates, Head First Servlets and JSP, O'reilly (SPD), Second Edition, 2008.
2. Cay S. Horstmann, Gary Cornell, Core Java™ 2: Volume II-Advanced Features. Prentice Hall PTR, 2001 (ISBN: 0-13-092738-4)
3. Eric Jendrock, Jennifer Ball, D Carson and others, The Java EE 5 Tutorial, Pearson Education, Third Edition, 2003.
4. Herbert Schildt, Java 2: The Complete Reference, Tata McGraw-Hill. Fifth Edition, 2002 (ISBN 0-07-049543-2).
5. Ivan Bayross, Web Enabled Commercial Applications Development Using Java 2, BPB Publications, Revised Edition, 2006.
6. Joe Wigglesworth and Paula McMillan, Java Programming: Advanced Topics, Thomson Course.
7. The Java Tutorials of Sun Microsystems Inc.

**Paper III**  
**Operating Systems and Linux**  
**Unit I: (30 lectures)**

**Introduction:** Overview of Operating System, Evolution of Operating System, Different Services of Operating System.

Operating System for Main Frame Computer Systems: Batch Processing, Systems, Micro Programmed Systems, Time-Sharing System, Understanding Multiprogramming, Multiprocessing and Multitasking.

Operating System for Multiprocessor Systems and Distributed Systems, Operating System for Client Server & Peer-to-Peer Systems, Clustered Systems, Real time Operating System.

Components of Operating System: Process Management, Main Memory Management, Secondary Storage Management, File Management, I/O Management, Operating System Services, Command Interpreter, Interface between user and Operating System, Introduction to System calls: Types of System calls.

**System programs and Operating System Structure:** Layered approach, Kernel based approach, Operating System design and Implementation.

**Process Management:** Introduction to Process, Process states: two state and five state model, processes and resources, concurrent processes, process description, process control block and its role operation on processes, Cooperating processes.

**Interprocess Communication and light weight process:** Direct and Indirect communication, message passing, synchronization, buffering. Threads, single and multithreaded processes, user and kernel threads multithreaded models, Threading issues, Creation of threads.

**CPU Scheduling and Process synchronization:** Need for Process scheduling, queuing diagram, scheduler and its types, Scheduling queues. Need for Process switching, context switching, process synchronization, CPU scheduling algorithms.

General structure of a typical process, Critical Section Problem and its solutions, Two and multiple process solutions. Need for Mutual Exclusion, Classifying process Interactions and Achieving mutual exclusion: Dekker's Algorithm, Peterson's Algorithm and their final correct solution for two processes.

**Tools for process synchronization:** Semaphores, Binary semaphores, monitors, message passing: their use and implementation for mutual exclusion.

[SG: Chapter 1, 3, 4, 5 6 and WS: Chapter 3, 4, 5]

## Unit II : (30 lectures)

### **Classical Problems of Process synchronization:**

Producer-Consumer Problem for infinite and bounded buffers and its bounded buffer solution using Semaphore monitor and messages Reader-writer problem and its solutions with readers' priority and writers' priority, Dining-Philosophers Problem and its solutions.

**Concurrency and Deadlock:** Deadlocks and their Characteristics, Resource Allocation Graph, methods of handling deadlocks, Deadlock prevention technique. Deadlock detection and avoidance safe and unsafe state, resource allocation algorithm, Banker's algorithm, Recovery from deadlock.

**Memory management:** Memory Management and its need, swapping technique. Contiguous memory allocation, Paging and Segmentation, Segmentation with paging, Introduction to Virtual Memory, Demand paging technique, Need for page replacement, Basic scheme, replacement algorithms, Thrashing and its cause.

**File System:** File Concept: attributes, operation, types, structure. File access methods, Different directory structure, File System structures, File system implementations, Directory implementations, Allocation methods, Free space management.

**I/O System:** Principles of I/O hardware and Software: typical bus structure, polling, interrupts, direct memory access, application I/O Interface.

**Disk Scheduling:** FCFS, SSTF, SCAN, C-SCAN. Examples related to disk arm movement.

**Operating System Security:** Introduction to security problem Program and system threats, Intrusion Detection, Computer-Security Classifications.

[SG: Chapters 6, 7, 8, 9, 10, 11, 12, 13, 14, 15 and 19]

Ws: Chapters 4, 5]

### **References:**

1. SG: "*Operating System Concepts*:- Abraham Silberschatz, Peter Galvin, Greg Gagne, John Wiley and Sons.
2. WS: "*Operating Systems*" – William Stallings, Pearson-Prentice Hall, Fifth Edition.

### **Additional References:**

1. TW: "*Operating Systems*" – Andrew Tanenbaum and Albert WoodHull, Second Edition, PHI.
2. AG: "*Operating Systems*" – Achyut S. Godbole, Tata Mc-Graw Hill,

## Unit III : (30 lectures)

**Linux System:** History, Design Principles, Kernel Modules, Process Management, Scheduling, Memory Management, File System, I/O. (SGG: 20.1 to 20.8)

**Linux Basics:** Looking into the Linux Kernel, GNU Utilities, Desktop environments, (RB: Chapter-1), The Linux console (RB:Chapter-2) The Unix/Linux architecture, Features of Unix/Linux. (SD:2.1, 2.2).

**Basic bash shell commands:** Starting the shell, Shell prompt, Filesystem Navigation, File directory listing, File handling, Directory handling, Viewing file contents. (RB: Chapter-3).

**More bash shell commands:** Monitoring programs, Monitoring disk space, Working with data files: Sorting, Searching, Compressing, Archiving. (RB: Chapter -4)

**The Linux environment variable:** Environment variables, setting environment variables, Removing environment variables, Default shell environment variables, setting the PATH environment variables, Locating system environment variables, Variables arrays, Using command aliases. (RB Chapter-5).

**Understanding Linux file permission:** Linux security, Using Linux groups, Decoding file permissions, Changing Security settings, Sharing files, (RB: Chapter-6).

**Basic script building:** Using multiple commands, Creating a script file, Displaying messages, Using variables, Redirecting Input and Output, Pipes, Performing math, Exiting the script, (RB: Chapter-8).

**Using structured commands:** Working with the if-the, if-then-else and nesting if statements, test command, compound condition testing, advanced if then features, the case command. (RB: Chapter-9).

**More Structured commands:** For command, C-style for command, whole command, until command, nesting loops, Looping on file data, controlling the loop, processing the o/p of a loop. (RB: Chapter-10).

**Handling user input:** Command line parameters, Special parameter variables, shift command, working with options, Standardizing options, Getting user I/P. (RB: Chapter-11).

### **Unit III : (30 lectures)**

**Presenting Data:** Understanding I/O, Redirecting O/p in scripts, Redirecting I/p in scripts, Creating your own redirection, Listing open file descriptors, Suppressing command o/p, Using temporary files, Logging Messages. (RB: Chapter-12).

**Script control:** Handling signals, Running scripts in background mode, Running scripts and without a console, Job control Job Scheduling Commnads: nice, renice, at, batch, cron table, Running the script at boot. (RB: Chapter-13).

**Essential System Administration:** root: The system administrator's login, The Administrator's privileges, Startup and Shutdown. (SD: 15.1, 15.2, 15.5).

**TCP/IP networking:** TCP/IP Basics, TCP/IP Model, Resolving IP addresses, Applications, telnet, ftp, Berkeley commands. (SD: Chapter-17).

**Advanced System Administration:** Partitions and file systems, /etc/fstab, fsck, System startup and init, Shutdown and sync operation. (SD: 25.2, 25.6.3, 25.7, 25.8, 25.9).

**Editors:** Sed and awk (RB: Chapter-16).

**Linux Firewall:** Introduction to firewall, Displaying status of firewall, Turning an iptables Firewall on/off, Testing firewall, Configuring the firewall for Remote SSH Administration. (CS: 3.0, 3.7-3.11).

**OpenSSH:** Introduction, Starting and Stopping OpenSSH, Creating strong Passphrases, Setting up host keys for simplest authentication, Generation and copying SSH Keys, Public key authentication to protect system password, Hardening OpenSSH, Changing Passphrase, Retrieving a key, Managing SSH Configuration file, X-Windows and SSH, Mounting remote file system with sshfs. (CS: 7.0 to 7.13, 7.17).

**Main References:**

1. SGG: Operating System Concepts, 6e, Silberschatz, Galvin and Gagne, Wiley.
2. SD: Unix Concepts and Applications, 4e, Sumitabha Das, TMH.
3. RB: Linux Command line and Shell Scripting: Bible, Richard Blum, Wiley-India.
4. CS: Linux Networking Cookbook, Carla, Schroder, O'reilly.

**Additional References:**

1. Unix Complete Reference, TMH.
2. Linux Complete Reference, TMH.
3. Linux Command Reference, Shroff.

**Paper IV**  
**DBMS-II & Software Engineering**

**UNIT I: [30 Lectures]**

**Decomposition:** Functional dependency, Closure of a set of functional dependency, Lossless-Join decomposition, Multi valued dependency and fourth normal form, Join dependency, Fifth normal form.

**Concurrency Control:** Concept of a transaction ACID properties, Serial and serializable schedules, Conflict and View serializability, Precedence graphs and test for conflict serializability.

**Enforcing serializability by locks:** Concept of locks, the locking scheduler, Two phase Locking, upgrading and down grading locks, Concept of dead locks, Concurrency and control by time stamps, The Thomas Write rule.

**Crash Recovery:** ARIES algorithm, The log based recovery, recovery related structures like transaction and dirty page table, Write-ahead log protocol, check points, recovery from a system crash, Redo and Undo phases.

**UNIT II: [30 Lectures]**

**Sequences:** creating sequences, referencing, altering and dropping a sequence.

**Fundamentals of PL/SQL:** Defining variables and constants, PL/SQL expressions and comparisons: Logical Operators, Boolean Expressions, CASE Expression Handling, Null Values in Comparisons and Conditional Statements, PL/SQL Datatypes: Number Types, Character Types, Boolean Type, Datetime and Interval Types.

**Overview of PL/SQL Control Structures:** Conditional control: IF and CASE Statements, IF-THEN Statement, IF-THEN-ELSE Statement, IF-THEN-ELSE Statement, CASE Statement, Iterative, Control: LOOP and EXIT Statement, WHILE-LOOP, FOR-LOOP, Sequential Control: GOTO and NULL, Statements, Concept of nested tables.

**Query evaluation:** System Catalog, Evaluation of relational operators like selection, projection, join and set, introduction to query optimization.

**Cursors:** Concept of a cursor, types of cursors: implicit cursors: explicit cursor, Cursor for loops, Cursor variables, parameterized cursors.

**Transactions in SQL:** Defining a transaction, Making Changes Permanent with COMMIT, Undoing Changes with ROLLBACK, Undoing Partial Changes with SAVEPOINT and ROLLBACK, Defining read only transactions, explicit locks: transaction and system level. Choosing a Locking Strategy: ROWSHARE AND ROW EXCLUSIVE Mode.

**References:**

- (a) Ramakrishnam, Gehrke. "Database Management Systems:. McGraw – Hill.

(Chapter 12.1-12.4, 16.1-16.4, 17.1=17.4, 17.6.2, 18.1-18.6, 19.1-19.3, 19.5, 19.8).

- (b) Ivan Bayross, "SQL.PL/SQL – The Programming language of Oracle". B.P.B. \$\$\$\$ 3<sup>rd</sup> Revised Edition: [Chap 11, 15-17].
- (c) Michael Abbey, Michael Corey, Ian Abramson, Oracle 8i A Beginner's Guide,\$\$ McGraw- Hill. (Chapter 8).

### **Additional References:**

- (a) Elsmasri and Navathe. "Fundamentals of Database Systems". Pearson Education.
- (b) Peter Rob and Coronel, "Database Systems, Design, Implementation and Management", Thomson Learning.
- (c) C.J. Date, Longman, "Introduction to database Systems", Pearson Education.
- (d) Jeffrey-D, Ullman, Jennifer Widom, "A First Course in Database Systems", Pearson Education.
- (e) Martin Gruber, "Understanding SQL", B.P.B. Publications.
- (f) George Koch and Kevin Loney, ORACLE "The complete Reference", Tata McGraw Hill, New Delhi.

### **UNIT III: [30 Lectures]**

**Project Management:** Revision of Project Management Process [Re13-2.4.1]. Role of Project Manager, Project Management Knowledge Areas [Ref 1-Chap 3(Pg. 99-100, 104)]. Managing changes in requirements [Ref3-2.4.4].

Role of software Metrics [Ref 3-1.3.2, Ref 2-23.1.1].

**Size & Effort Estimation:** Concepts of LOC & Estimation [Ref 2-26.6.2, 26.6.3], Function Point [Ref 3-3.6.1, Ref 2-25.2.3], COCOMO Model [Ref 3-5.2.4], Concept of Effort Estimation & Uncertainty [Ref 3-5.2, 5.2.1], [Ref 1- Chap 3 (Pg 121)].

Project Scheduling [Ref 2-27.2, 27.2.1 & Ref 3-5.3.2], Building WBS, Use of Gantt & PERT/CPM chart

[Ref 1-Chap 3(Pg112-118)][3 lectures] Staffing [Ref 1-Chap 3(Pg 100). Ref 3-5.3.4].

Configuration Management Process & Functionally & Mechanism [Ref 3-2.4.3].

Process Management, CMM & its levels [Ref 3-2.4.5].

Risk Management & Activities [Ref 3-5.6.1-5.6.3].

Management of OO software Projects- Object oriented metrics, Use-base Estimation [Ref 2-23.3.3, 25.2.5, 26.6.7-26.6.8.27.5.3].

Selecting development tools, Introduction to CASE [Ref 1- Chap 2 (Pg 85-86, Pg 341)]/

### **References:**

1. System Analysis & Design – Satzinger, Jackson, Burd, Cengage Learning, India Ed.
2. Software Engineering – A Practitioner's Approach, 7<sup>th</sup> Edition, McGraw Hill int.
3. Integrated Approach to Software Engineering – Pankaj Jalote (Narosa), 3<sup>rd</sup> Edition. By – Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides.

#### **UNIT IV: [30 Lectures]**

Software Testing; Introduction to Quality Assurance, Six Sigma [Ref 1-14.4.4, 16.2, 16.5.2].

Testing Fundamentals, Common Terms (like Error, Fault, Failure, Bug, Crash) Objectives of testing, Challenges in testing, Principles of Testing [Ref 2.1.2, 1.2.1, 1.2.3.1.2.4, 1.5].

Static Testing: Levels of testing such as Unit testing, Integration testing, System testing, Validation Testing, Acceptance testing.

Types of testing such as Black Box, White box, Functional, Performance, Regression, Acceptance, Volume, Stress, Alpha, Beta testing [Ref 2-1.1].

Black box, Testing: Introduction, Equivalence partitioning, Boundary-value analysis, Robustness testing, Cause Effect Graph, [Ref 2-4.1, 1-4.5. Ref 3-10.2.3].

White Box Testing: Statement Coverage, Branch/Decision Coverage, Condition Coverage, Graph Matrix, Cyclomatic complexity, Mutation Testing [Ref 2-5.1-5.6].

Object Oriented Testing & Web Site testing: Object Oriented Testing Strategies, Testing methods [Ref 1-19.3, 19.4.3, 19.4.5, Ref 2-11.2] Overview of web site testing [Ref 1-20.1.1].

Planning Software Testing – Test plan, Test plan specification, Test Case Execution and Analysis, Defect logging and tracking [Ref 3-10, 4.2-10.4.5].

#### **References:**

1. Software Engineering - a Practitioner's Approach, 7<sup>th</sup> Edition. McGraw Hill Int.
2. Software Testing – Concepts & Practices, Narosa.
3. Integrated Approach to Software Engineering – Pankaj Jalote (Narosa).

#### **Additional References:**

1. Software Engineering : Waman Jawadekar, TMH
2. Software Engineering : Sommerville, VIIIth Edition, Pearson Education.

### **Group I : Linux & Database**

#### **List of Practicals**

1. Study of Basic and advanced Linux commands.
2. Study of filter commands
3. Basic Shell scripting
4. Advance Shell scripting
5. Writing PL/SQL Blocks with basic programming constructs.
6. Procedures and Functions in PL/SQL Block
7. Implementing cursors & sequences
8. Study of transactions and locks

**Note: Demo of installation of Linux should be given to students.**

### **Group II : Advanced Java**

#### **List of Practicals**

1. Developing GUI applications using Swing.
2. Database programming with Java
3. Study of Multithreading
4. Study of Socket programming and Distributed computing
5. Server-Side Programming using Servlets
6. Server-side programming using JSP
7. Developing Enterprise Java Beans.
8. Developing Web services in Java

### **Group III : Elements of Software Engineering and Project Management**

#### **List of Practicals**

1. Preliminary Design of a software system in context with its components, descriptions, limitations, advantages and disadvantages for a given Case Study.
2. To draw Event Table & Use Case for a given Case Study.
3. To draw Activity Diagram, Class Diagram & Object Diagram for a given Case Study.
4. To draw Sequence diagram/Collaboration diagram for a given Case Study.
5. To derive tables from entities & relationships with integrity constraints for a given Case Study.
6. To draw Component, Package & Deployment diagram for a given Case Study.
7. To design Form Layouts/ Web Pages/Report layouts for a given Case Study.
8. To list test cases and validations for a given Case Study.

### **Group IV : OOAD Project**

#### **General Guidelines for Project Development**

1. Faculties should arrange project demos for SY students at the end of the year or just at the beginning of TY. The demos can be of some good students of previous TY batches or it can be a project developed by faculties themselves.
2. SY students should be encouraged to start finding projects in the summer vacation. It will be advantageous if students finish majority of the preliminary

investigation phase during summer vacation. Faculties may take one or two introductory sessions for SY students before the vacation which will help students to work on preliminary investigation phase during summer vacation.

3. It can be Stand Alone, Multi-user or Web Based. Projects must be done using Java or Net Technologies & RDBMS.
4. Each student shall do the project individually, though a project with the same topic name could be done by more than one student.
5. A project guide should be assigned to students (Maximum 24 students for 1 guide). He/she who will assign a schedule for each phase of the project and hand it over to students. The guides should oversee the project progress on a weekly/fortnightly basis. The guides should control iteration if any non-linear technique is used for project development. It is advisable that the design phase gets over in the first term. Sample schedule can be as follows-

Phase	Time of completion
Preliminary investigation	30 <sup>th</sup> June
System Analysis	14 <sup>th</sup> August
System Design	30 <sup>th</sup> September
Coding	15 <sup>th</sup> December
Implementation	5 <sup>th</sup> January
Project Report Submission	15 <sup>th</sup> January

6. College can arrange few sessions by experienced industry people on project management/best practices/technologies etc.
7. The student will maintain a Gantt chart from the given schedule and completion of each phase should be noted on the chart.
8. Like Gantt Chart, a separate table should be maintained to check timely completion of the project. The table should contain the names of phases & its sub-phases, Expected Date of completion and Actual Date of completion. The guide should sign and mention the actual date of completion in the specific column of this table. This table should be produced at the time of final project demonstration and should be used for assigning marks. The dates of this table should match with the dates of Gantt Chart. For the students who fail to do projects as per the schedule, project guide can sign the projects with late remarks.

Sample Phase completion table-

Phase title	Expected date of Completion	Actual time of Completion with Guide's Signature	Remarks
I. Phase Name	30 <sup>th</sup> June		
a) Sub Phase I	20 <sup>th</sup> June		
b) Sub Phase N	30 <sup>th</sup> June		
II. Phase Name	14 <sup>th</sup> August		
a) Sub Phase I	5 <sup>th</sup> August		
b) Sub Phase N	14 <sup>th</sup> August		

9. After the Completion of phase/projects, demos can be planned in front of faculties/clients/students.
10. Projects should have at least following:
  - a) Good content management, presentation & meaningful images.
  - b) Data entry with Validations.
  - c) Suitable navigation schemes (menus/toolbars/tabs/links etc).
  - d) Record Manipulation (ad, update, delete, display, search, sort).
  - e) Transactions/Sessions/Reports/Feedback/Registration whichever applicable.
  - f) Login accounts (Admin & User) with separate functionalities for administrators and users.
11. A certificate should be added in the project report which should contain the following information-
  - a) The fact that the student has successfully completed the project as per the syllabus and that it forms a part of the requirements for completing the BSc degree in Computer Science of University of Mumbai.
  - b) The name of the student and the project guide.
  - c) The academic year in which the project is done.
  - d) Date of submission.
  - e) Signature of the project guide and the head of the department with date along with the department stamp.
  - f) Space for signature of the university examiner and date on which the project is evaluated.
12. Project should be evaluated by External Examiner as follows-
  - i. Project Quality → 10 marks
  - ii. Adherence to schedule → 10 marks
  - iii. Working of Project → 20 marks
  - iv. Student's Presentation → 10 marks

Note:

  - i. Evaluating "Adherence to schedule" : A project can approximately have 5 phases. Completion of these phases on time will carry 2 marks. Sub-phases need not go as per the schedule but main phase should be completed as per the plan. However, a grace period of 7 days should be given for completing each phase. If the project gets delayed after 7 days of actual time of completion, then the marks can be cut. Concession of some more days can be given for students having genuine reasons (medical cases etc) but it should be mentioned in remarks.
  - ii. Evaluating "Project Quality": It involves overall modules included in the project, whether it was sufficiently large enough so that it can be done in a year, whether validations were done for data entry, variety of reports etc.
  - iii. Evaluating "Working of the Project": It involves error-free execution of the project.
  - iv. Evaluating Student's Presentation: Marks can be given based on the presentation skills of a student. A student can prepare a power point presentation for the project.

## Index for Project Documentation

Acknowledgement
I. Preliminary Investigation
(i) Organizational Overview
(ii) Description of System
(iii) Limitations of present system
(iv) Proposed system and its adv. [for web project, URL can be mentioned].
(v) Feasibility Study
(vi) Stakeholders
(vii) Gantt Chart
II. System Analysis
(i) Fact Finding Techniques (Questionnaire, Sample Reports, Forms.)
(ii) Prototypes (if any)
(iii) Event Table
(iv) Use Case Diagram Scenarios & Use Case Description
(v) ERO
(vi) Activity Diagram
(vii) Class diagram
(viii) Object diagram
(ix) Sequence diagram/Collaboration Diagram
(x) State diagram
III. System Design
(i) Converting ERD to Tables
(ii) Design class diagram [with UI classes, Persistent classes etc...]
(iii) Component Diagram
(iv) Package Diagram
(v) Deployment Diagram
(vi) Prg Flow charts & System flow chart
(vii) Structure Chart (Prg level and System level)
IV. System Coding
(i) Menu Tree/Sitemap
(ii) List of tables with attributes and constraints
(iii) Design Patterns used (if any)
(iv) Program Description [Programs/Classes and their responsibilities in brief] with Naming Conventions.
(v) Validations
(vi) Test Cases, Test Data and Test Results [Write test cases for all imp. Programs]
(vii) Screen Layouts & Reports Layouts
(viii) Program Listing/for dummy proj]
(ix) System implementation/Uploading
(x) Future Enhancements
(xi) References and Bibliography

**Applied Component**  
**Applied Component I**  
**Principles of Web Design & Web Technologies**  
**Unit I [15 lectures]**

**Web Site Design Principles** – Design for the Medium, Design for the Whole Site, Design for the User, Design for the Screen.

**Planning the Site** – Create a Site Specification Identify the Content Goal, Analyze your Audience, Build Planning Site Navigation – Creating Navigation, Using Text-based Navigation, Using Graphics-Based Navigation.

**Creating Page Templates** – Understanding Table Basics, Table Pointers, Creating a Page Template.

Unit Reference

Code : PWD (Ch 2, 3, 4, 5)

HTML – HTML 4.0 Tag Reference, Global Attributes, Event Handlers, Document Structure Tags, Formatting Tags, List Tags, Hyperlinks, Image & Image map, Table Tags, Form Tags, Frame Tags, Executable Content Tags and Style Sheets, CSS

Unit References:

Code : ELJO (Ch 3-9)

**Unit II [15 lectures]**

Introduction to Java Script (Functions, objects)

Client-Side Java Script – Java script in web browser, windows and frames, the document object model, events and even handling, forms and form elements, dynamic html and saving state with cookies.

References

Code : JSDG (Ch 7, 8, 11-18)

Code : Dummies (Ch 2)

XML – Introduction to XML, Problems with HTML & SGML, Types of XML Markup, Document Type Definitions, Using Style Sheets with XML, creating XML well formed, valid Documents.

Code : ELJO (Ch 12, 13)

Code : Unleashed (Ch 1 to 3, 5, 6 8, 9)

**Unit III [15 lectures]**

**Introduction to Ajax** - Working of Ajax from a user's perspective and developer's perspective, Applications of Ajax : Searching in real time with live searches, Getting the answer with autocomplete Chatting with friends, Dragging and dropping with Ajax,

Getting instant login feedback. Ajax-enabled popup menus, Modifying Web pages on the fly, Google Maps and Ajax.

**Ajax and PHP** – Starting with PHP, Getting a Handle on variables Handling Your Data with Operators, Making Choices with the if Statement, Round and Round with Loops, Handling HTML Controls, Getting data from text field, Checking out data from check boxes, Tuning in data from radio buttons, Sending Data to the Server.

**XML and Ajax** – Creating and opening the XMLHttpRequest object, Handling asynchronous downloads, relative versus absolute URLs, Interactive Mouseovers Using Ajax, Server-Side Scripting, Choosing a server-side scripting language, connecting to a script on a server, setting up a Web page to read XML, Handling the XML you read from the server, Extracting data from XML, Passing Data to the Server with GET, passing data to the Server with POST.

Ajax in Depth

Returning JavaScript from the Server, Calling a Different Domain, Reversing the roles: performing validation on the server, getting data with HEAD Requests., Returning all the header data you can get finding the last-modified date, Debugging Ajax, Using GET to get text, Using GET to get XML, using POST to post data and get text, Using POST to post data and get XML.

Unit References

Code :

Dummies (Ch 1, 3, 4, 5, 10)

#### **Unit IV [15 lectures]**

\$\$\$\$XML in Ajax Applications

Requesting XML Data in Ajax, Extracting XML data using properties right on the node, introducing the JavaScript properties, Navigating an XML document using JavaScript properties, Extracting with node Value, Accessing XML Elements by Name, Accessing Attribute Values in XML Elements, Validating XML, Documents in Ajax Applications.

#### **Working with Cascading Style Sheets in Ajax Applications**

An Ajax-Driven Menu System, Setting up the styles, Handling mouse events, Displaying menu, Hiding a menu, Handling the menu items, Displaying Text that Gets Noticed, styling text, handling colors and backgrounds, positioning using styles.

#### **Ajax Design Issues**

Breaking the Back Button and Bookmarks, giving visual, cues, leaving the user in control, remembering all the different browsers, showing users when text changes, avoiding a sluggish browser, handling sensitive data, creating a Backup Plan, showing up in Search Engines, Sidestopping a Browser's Cache.

Code :

Dummies (Ch 8, 9 11)

**References**

1. ELJO : Using HTML 4, XML & JAVA by Eric Ladd & Jim O'Donnell. (Platinum Edition) (PH)
2. PWD : Principles of Web Design by Joel Sklar.
3. JSDG : Java Script the definitive guide by David Flanagan
4. Dummies: Ajax for Dummies Steve Holzner, PhD. Wiley Publishing Inc.
5. Unleashed: XML Unleashed Techmedia SAMS, Michael Morrison

**Additional Ref:**

1. Ajax in Action Dave Crane, Eric Pascarello, Darren James
2. Beginning javascript, Wilton, Wrox Publication
3. Head first HTML which CSS and XHTML, Elisabeth freeman and Erice freeman, SPD O, reilly
4. AJAX For Beginners, Ivan Bayross and Sharanam Shah, SPD The team.

## **Applied Component II – Dot Net Technologies**

### **Unit I [15 lectures]**

Introduction to NET Framework 3.5 (or above) : Overview of NET Framework, Objectives, Main Components of NET Framework and their overview, Types of Applications.

NET Framework Architecture – CLR (Goal of CLR, Services/Features, Benefits, managed execution process, Automatic memory management). CTS(CTS overview, type definitions, type members, Different types if data such as delegates, pointers, arrays, interfaces), Meta Data, structure of metadata and self describing components, cross language interoperability & CLS. Assemblies(Assembly overview, Benefits, Contents, Types)

Ref. 1. MSDN

Introduction to NET IDE : [2 Sessions] [Ref 1 : &]

Introduction to VB 2008 : Adding forms, controls, setting properties, adding code, handling events, displaying simple messages, data types, declaring variables, strings and constants, operators, expressions, declaring methods, subroutines, procedures, passing & returning arguments, decision statements (if select) Loops Adding Comments, converting data types [ 4 Sessions] [Ref 1 : Chap 1, 2, 3, 100-112, 116-123, 142-149, 152-157, 168-175, Ref 2 : Pg 1-32, 35-75, 85-115]

Ref:

1. Murach Visual Basic 2008, Anne Boelun, Murach (Chapter 1, 2, 3, 4)
2. Mascring Visual Basic 2008, Evangelos Petrooutsos, Wiley-India Edition (Chapter 1, 2, 3)

**Working with Strings, Dates and Time:** Char, String class and functions, converting strings to numbers, Date Time Class, formatting Dates [Ref 1 : Pg 264-279, Ref 2: Pg 461-473, 481-492]

**Window controls and Events:** Using basic window controls such as forms, labels, buttons, text boxes, check boxes, radio buttons, list boxes, combo boxes, adding properties, processing events, using various dialog boxes such as MessageBox, Open dialog, save dialog, Stream/Reader and Stream Writer, using menus, adding toolbars, status bars MDI form, processing events, adding event handlers for one/many events.

[Ref 1: Pg 176-185, 294-303, 306-317, 666-669, 730-731, 736-751, Ref 2 :Pg 173-190, 195-209, 217-246, 253-263, 267-279, 572-576]

**Structured Exception Handling :** try, catch, finally blocks, throwing exceptions, Err object, using masked textboxes [Ref 1 : Pg 190-203, 218-219, Ref 2 : Pg 147-149]

**Collections :** Creating and manipulating arrays, using system array, ArrayLists Collection, [Ref 1 : Pg 224-235, 240-243, 246, 251, 258-259, Ref 2 : Pg 499-516]

Ref :

1. Murach Visual Basic 2008, Anne Boehm, Murach (Chapter 1, 2, 3, 4, 5, 6, 7, 9, 10, 24)

2. Mastering Visual Basic 2008, Evangelos Petroutsos, Wiley-India Edition (Chapter 1, 2, 3, 4, 6, 7, 8, 13, 14, 15)

Additional Ref:

1. MurachVB. NET Programming with ADO. NET

### **Unit II [15 lectures]**

OO Programming : Understanding objects, creating classes, adding constructors, properties, methods and variables, access specifiers, shared members, defining & using events, implementing inheritance, partial classes, creating, referencing, importing namespaces [Ref 1 :Pg 332-351, 550-551, 554-555, 580-582, 640-643, Ref 2 : Pg 349-363, 375-383, 395-426]

**Building class libraries:** Understanding class libraries, using strong names, registering assemblies, designing class libraries [Ref 3 : Chapter 13]

**Creating Custom controls:** Enhancing Existing controls, building compound controls, adding custom events [Ref 2 : Pg 429-442, 450-452]

ADO.NET : ADO.NET and Date management, advantages of ADO.NET, ADO.NET Object model, Data objects, Data Source Interaction objects, NET Data provides [Ref 1 : Pg 414-425, Ref 4: Pg 486-492]

Using Connection, Command, DataReader classes, Queries returning result sets, scalars, passing parameters in queries, [Ref 1: Pg 506-519, Ref 2: Pg 806-822]

Disconnected Data, Data Adapter, Using Data Set (Typed), Data Table, Data Row and Data Column, Command Builder, Modifying and updating disconnected data [Ref 2: Pg 822-843, 84-854]

Creating and using Data sources, Binding (Simple and Complex binding) controls with data sources, DataGridView control, role of binding source, TableAdapter, AdapterManager and BindingNavigator classes [Ref 1: Pg 425-451, 458-459, 470-479, 488-493, Ref 2: Pg 855-876]

LINQ : Introduction to LINQ, advantages of LINQ, using from, where, order by, select [Ref 1: Pg 704-715]

Ref:

1. Murach' Visual Basic 2008, Anne Boehm, Murach (Chapter 8, 11, 13, 14, 15, 16, 17, 18, 19, 20, 23, 25)
2. Mastering Visual Basic 2008, Evangelos Petroutsos, Wiley-India Edition
3. Beginning Microsoft Visual Basic 2008 (Wrox) by Thearon Willis & Bryan Newsome (Chapter 13)

Additional Ref:

1. Murach's VB.NET Programming with ADO.NET

### **Unit III [15 lectures]**

ASP.NET Applications- Evolution of Web Development [Ref 1: Pg 1-10]. Creating ASP.NET Applications, designing Web pages, Anatomy of web, form [Ref 1: Pg 81-99], Writing Code-behind and event handlers, [Ref 1: Pg 105-118], ASP.NET files, Application directories, server controls, HTML control classes, Page Class, Global asax files and application events, ASP.NET configuration files [Ref 1: Pg 121-159]

Web Controls – Advantages of web controls, Web control classes, AutoPostBack and Web control events, [Ref 1 : Pg 163-199]

State Management – Maintaining state using ViewState [Ref 1: Pg 201-206], Query String, Cross Page posting, URL Encoding, Cookies, Sessions State [Ref 1: Pg 209-230], Application State [Ref 1: Pg 235-236]

Validation and rich controls – validations and Validators controls, undertaking regular expressions [Ref 1:Pg 321-340], Calendar Control, AdRotator, Advertisement file, AdRotator class [Ref 1:Pg 343-355]

#### References:

1. Beginning ASP.NET 3.5 in VB 2008, From Novoice to professional, second edition, Mathew Mcdonald (Chapters 1, 4, 5, 6, 7, 9, 10, 11]

### **Unit IV [15 lectures]**

ADO.NET & Data Binding – ADO.NET fundamentals [Ref 1: 486-522], Single-value, repeated-value Data Binding, Data binding with ADO.NET. Data source controls [Ref 1: 525-558], GridView, defining columns, selecting, sorting and paging GridView [Ref 1:Pg 565-579, 584-589]

Using XML – XML Role in NET.XML Classes [Ref 1: Pg 635-636, 642-658]

Protection and performance: Determining security requirements. ASP.NET Security model [Ref 1:Pg 679-686], caching, types of caching, example of output caching, data caching [Ref 1:Pg 809-813, 819-822]

AJAX in ASP.NET – Using ScriptManager, Partial refreshes, UpdatePanel, Triggers, Timed Refreshes [Ref 1 –843-849, 852-857, 861-863]

Web service – what is web service. ASP.NET Web services, Creating a simple web service, Consuming Web service [Ref 2:Pg 981-982, 984-990] (Ref – Mastering VB 2008 Chapter 27)

#### References:

1. Beginning ASP.NET 3.5 in VB 2008, from Novoice to professional, second edition, Mathew Mcdonald (Chapter 13, 14, 15, 16, 17, 19, 20, 24, 25)
2. Mastering Visual Basic 2008, Evangelos Petroustos, Wiley-India edition (Chapter 27)

Additional References:

1. Beginning ASP.NET 3.5 in C# VB (Programmer to Programmer) By Imar Spaanjaars
2. Murach's ASP.NET 3.5 with VB 2008.
3. ASP.NET 3.5 Application Architecture & Design- Vivek Thakur (SPD-PACK 1)

List of Practicals

1. Study of basic and Advanced HTML Tags.
2. Applying CSS, Representing Data using XML.
3. Executing Client Side Scripts using JavaScript.
4. Using AJAX for enhancing web-sites.

**Group II AC: DotNet Technologies**

Note: Demo on using NET IDE for building NET projects, IntelliSense, Running & Debugging Application should be given.

List of Practicals

1. Study of basic and advanced VB.NET Programming alongwith controls, Data Conversion and Message Box, MDI, Menus, Toolbars, Dialog boxes and Collections.
2. Implementing OOP concepts in VB.NET, Creating Custom controls, ADO.NET Programming and Data binding.
3. Fundamentals of ASP.NET such as using HTML/Web Controls, Autopostback, Application events, configuration files, State management, Validation & Rich controls.
4. ADO.NET in ASP.NET with Data Binding, Interacting with XML documents, Performance improvement using Caching, Creating & Consuming Web services.