Rajarshi Shahu Mahavidyalaya, Latur
(Autonomous)

Department of Computer Science and Information Technology
Syllabus for under graduation course (B.Sc.C.S.)

(With Effect from Academic Year:2014-15)

Name of the Programme: Bachelor of Computer Science (Second Year Semester III + IV)

Semester : III

Code No	Course Name	Maximum Marks		Total Marks	Credits
		Theory / Practical	Internal		
U-COE-301	English	30	20	50	2
U-OOP-382	Object Oriented Programming Using C++	30	20	50	2
U-PCN-383	Programming in C#.Net	30	20	50	2
U-CON-384	Computer Network	30	20	50	2
U-OPS-385	Operating System	30	20	50	2
U-ADC-334-A	Addon Course-I (Android O.S.)	30	20	50	2
U-LAC-386	Object Oriented Programming Using C++	50	-	50	2
U-LAC-387	Programming in C#.Net	50	-	50	2
U-LAC-388	Computer Network	50	-	50	2
U-LAC-389	Open Office	50	-	50	2

Semester : IV

Code No	Course Name	Maximum Marks		Total Marks	Credits
		Theory / Practical	Internal		
U-COE-401	English	30	20	50	2
U-SOE-481	Software Engineering	30	20	50	2
U-ASN-482	ASP.Net	30	20	50	2
U-OSP-483	Oracle 10g SQL & PL/SQL	30	20	50	2
U-MUF-484	Multimedia Using Flash	30	20	50	2
U-ADC-434-M	Addon Course-II (Mobile App. Development)	30	20	50	2
U-LAC-485	Software Engineering	50	-	50	2
U-LAC-486	ASP.Net	50	-	50	2
U-LAC-487	Oracle 10g SQL & PL/SQL	50	-	50	2
U-LAC-488	Multimedia Using Flash	50	-	50	2

Course Title: Object Oriented Programming using C++
Course Code:U-OOP-382

Total Teaching Hours: 50 **Total Marks: 50**
Credit: 2

Learning Objective:

1.Understand object oriented programming and advanced C++ concepts

1.1 Be able to explain the difference between object oriented programming and procedural programming.

1.2 Be able to program using more advanced C++ features such as composition of objects, operator overloads, dynamic memory allocation, inheritance and polymorphism, file I/O, exception handling, etc.

1.3 Be able to build C++ classes using appropriate encapsulation and design principles.

2. Improve your problem solving skills

2.1 Be able to apply object oriented or non-object oriented techniques to solve bigger computing problems .

2.2 Ultimate goal: to make you a good programmer.

Course Outcomes:

- Gain the basic knowledge on Object Oriented concepts.
- Ability to develop applications using Object Oriented Programming Concepts.
- Ability to implement features of object oriented programming to solve real world problems.
- Use the characteristics of an object-oriented programming language in a program.
- Use the basic object-oriented design principles in computer problem solving.
- Use the basic principles of software engineering in managing complex software project.
- Program with advanced features of the C++ programming language.

Syllabus

UNIT- I: Introduction to Object Oriented Programming

1. Principles of Object Oriented Programming (OOP):

Evolution of C++ - Programming Paradigms - Key Concepts of OOP - Advantages of OOP - Usage of OOP and C++ .Input and Output in C++- Streams-Stream classes Unformatted console I/O operations-Member functions of istream class-manipulators-manipulators with parameters

2. Introduction to C++:

Tokens, Keywords, Identifiers, Variables, Operators, Expressions and Control Structures: If,If. Else, Switch – Repetitive Statements- for, while, do..while - Pointers and arrays

UNIT II: Class , Functions and Constructors

3. Structures and Unions:
Declaration of structures, Accessing structure members, Structure Initialization, Arrays of structure, nested structures, structure with pointers, functions & structures, Unions, Structure/Union Versus Class in C++.

4. Class Declaration:
Data Members, Member Functions, Private and Public Members, Data Hiding and Encapsulation, Array within a class.

5. Class Function Definition:
Member Function definition inside the class and outside the class, Friend Function, Inline Function, Static Members & Functions, Scope Resolution Operator, Private and Public Member Functions, Nesting of Member Functions. Creating Objects, Accessing class data members, Accessing member functions, Arrays of Objects, Objects as function arguments: Pass by value, Pass by reference, Pointers to Objects.

6. Constructors and Destructors:
Declaration and Definition, Default Constructors, Parameterized Constructors, Constructor Overloading, Copy Constructors. Destructors: Definition and use.

UNIT III: Inheritance and Overloading

7. Inheritance
Extending Classes Concept of inheritance, Base class, Derived class, Defining derived classes, Visibility modes : Private, public, protected; Single inheritance : Privately derived, Publicly derived; Making a protected member inheritable, Access Control to private and protected members by member functions of a derived class, Multilevel inheritance, Nesting of classes.

8. Function Overloading & Operator Overloading
Binary & Unary.

UNIT IV: Polymorphism and file operations

9. Polymorphism:
Definition, early Binding, Polymorphism with pointers, Virtual Functions, late binding, pure virtual functions.

10. Working with files:
header files,, redirection, Classes for File Stream Operations - Opening and Closing a File - End-of-File Detection - file input and output. File Pointers - Updating a File - Error Handling during File Operations - Command-line Argumentsbuffers &iostreams,

References / Books:
1. Object Oriented Programming with C++, E. Balagurusami, Fourth Edition, Tata Mc-Graw Hill

2. Object Oriented Programming in Turbo C++, Robert Lafore, Fourth Edition Galgotia Publications.
3. The C++ Programming Language, BjarnaStroustrup, Third Edition, Addison-Wesley Publishing Company.
4. Object Oriented Programming Using C++, Salaria, R. S, Fourth Edition, Khanna Book Publishing
5. Ashok N.Kamthane, Object Oriented Programming with ANSI & Turbo C ++, Pearson Education, 2006

Course Title: Programming in C#.Net
Course code: U-PCN-383

Total Teaching Hours: 50 **Total Marks: 50**

Learning Objectives:

- To understand the DOTNET framework andC# language features
- To develop object oriented programs on C#.
- To develop windows based applications on .NET framework.

Course Outcome:

After the successful completion of the course students will be able to:

- Describe basic concepts and develop programs in C# using object oriented features, delegates, events, errors and exceptions
- Interpret data access and develop windows applications
- Explain Common language runtime (CLR) as a platform for managed code
- Describe the features of Common language runtime (CLR) and develop efficient code with C# on .NET framework
- Develop windows based applications & services on .NET framework
- Describe overview of .NET framework
- Apply an understanding of the .NET technology and C#.net components to develop a windows based application which solves specified problem domain
- Use of ADO.NET technology for developing database oriented applications
- Understand the professional responsibility
- Apply an understanding of the need for high ethical standards in the practice of engineering towards people and the environment

SYLLABUS

UNIT I: Introduction to .net, Arrays and operators

What is .net?,.Net Framework, CLR, Visual Studio.net & .net Languages, Integrated Development Environment, Project types, c#.net History & design Goals, How c# differs from c++, Characterstics of c#.net, I/O Statement with C#.net, Boxing & Unboxing, Short Circuiting Operators, Array & ArrayList class, Jagged Array, String Class

UNIT II: Properties, Events, Delegates and C# namespaces

Properties & its type, Event, Delegate & Multicast Delegate, Creating & Starting thread, Exception handling, using keyword, creating and using namespaces, interface, Method overloading & method overriding, Partial Class

UNIT III: Windows Application

Event Driven Programming, Building windows application with visual studio, TextBox, Label & Button Control, ComboBox, ListBox ,CheckBox & GroupBoxControl, DateTimePicker , Timer

control,Building Menu,. MDI Form, PictureBox , ProgressBarControl, Common Dialog boxes, Introduction to WPF

UNIT IV: Ado.Net and Database Oriented Applications

How Ado.net differs from Ado, Advantages of Ado.net, Connected & Disconnected Architecture, Dataset, DataReader & DataAdapter, Managed Data Providers, DataGridViewControl, Developing Ado.net Based Application, Insert, Update & Delete operation on table, Filling the Dataset

Reference books:

1. Programming in C# A Primer - Second Edition By - E Balagurusamy
2. C#.Net Programming Wrox Publication
3. .Net 4.0 programming black book by KOGENT LEARNINGSOLUTIONS INC.
4. C# 2010 programming black book by KOGENT LEARNINGSOLUTIONS INC.

Course code: U-CON-384
Course Title: COMPUTER NETWORK

Credit:02

TOTAL TEACHING HOUR : 60 **TOTAL MARKS :50**

LEARNING OBJECTIVES:

The basic objective of Computer networking is to share resources, files, make connectivity. Computer networking enables computers to share data, application software and hardware devices. The most basic network consists of two computers connected directly by cable, for example sharing resources, such as printers and modems. Any computer capable of communicating on the network is known as a device or node.

COURSE OUTCOMES:

- To master the terminology and concepts of the OSI reference model and the TCP-IP reference model.
- To master the concepts of protocols, network interfaces, and design/performance issues in local area networks and wide area networks.
- To be familiar with wireless networking concepts.
- To be familiar with contemporary issues in networking technologies.
- To be familiar with network tools and network programming.

Syllabus:

UNIT I: NETWORKS

1.INTRODUCTION TO NETWORK:

1.1 Uses of computer network, network structure, network architecture.
1.2 ARPANET, NFSNET, INTERNET.
1.3 ISO Reference model.
1.4 Services provided by each layer.
1.5 TCP/IP model.
1.6 Services primitives', connection oriented and connectionless services.

UNIT II: PROTOCOLS

2. INTERNET PROTOCOLS:

2.1 Need of IP address.
2.2 Classes of IP Address.
2.3 Unicast, broad cast, multicast IP Addresses.
2.4 Subnet Mask.
2.5 IP datagram format.
2.6 Introduction to IPV6 and its features.

3. NETWORK LAYER:

3.1 Design issues store and forward packet switching.

3.2 Services provided to the transport layer.
3.3 Connectionless communication and services.
3.4 Connection oriented communication and services.

UNIT III: DATA TRANSMISSION

4. TRANSMISSION CONTROL:
4.1 Reliable services and unreliable services.
4.2 TCP services.
4.3 Process to process communication
4.4 Stream delivery services .
4.5 Full duplex communication.
4.6 TCP Features.
4.7 Numbering system.
4.8 Flow control
4.9 Error Control.
4.10 Conjection control.
4.11 Sliding window.

UNIT IV: NETWORK COMMUNICATION

5. NETWORK COMMUNICATION STANDARDS:
5.1 Ethernet-fast Ethernet, POE, FDDI, Token Ring
5.2 VLAN and its features, frame relay.

6. INTERNET AND WIRELESS TRANSMISSION:
6.1 Electronic mail, FTP, TFTP, SNMP.
6.2 Wireless networks Bluetooth and cellular radio network.
6.3 Satellite network and interactive services.

REFERENCE BOOKS:
1. Computer Networking by Tanum Baum
2. Data and computer communications by William Stallings.
3. Data communication and networking by Behrouz a forouzan.

Course Code: U-OPS-385
Course Title:- OPERATING SYSTEM

Total Teaching Hours: 60 **Theory Total Marks: 50**

CREDIT: 02 **Practical Total Marks: 50**

Learning Objective :

1. To introduce students with basic concepts of Operating System, its functions and services.
2. To familiarize the students with various views and management policies adopted by O.S. as pertaining with processes , Deadlock , memory , File and I/O operations.
3. To brief the students about functionality of various OS like , Linux and Windows XP as pertaining to resource management.
4. To provide the knowledge of basic concepts towards process synchronization and related issues.

Course Outcomes :

1. Master functions, structures and history of operating systems
2. Master understanding of design issues associated with operating systems
3. Master various process management concepts including scheduling, synchronization, deadlocks
4. Be familiar with multithreading
5. Master concepts of memory management including virtual memory
6. Master system resources sharing among the users
7. Master issues related to file system interface and implementation, disk management
8. Be familiar with protection and security mechanisms
9. Be familiar with various types of operating systems including Linux.

SYLLABUS

UNIT- I: Introduction to windows O.S.

CHAPTER 1. Introduction
1.1. Definition of O.S.
1.2. Types of O.S.
1.3. O.S. as resource manager
1.4. O.S. Process view
1.5. Hierarchical view

CHAPTER 2. Introduction to windows O.S.
2.1. Introduction
2.2. History
2.3. Files and Folders
2.4. Architecture of windows

2.5. Basics of Windows: desktop, my computer, etc

CHAPTER 3. Features of MS-Windows
3.1. GUI, Multitasking, Multi-user, network etc.
3.2. Important files of windows

UNIT II: Memory Management

CHAPTER 4. Memory management
4.1. Single continues allocation
4.2. Introduction to multiprogramming
4.3. Partitioned Memory management
4.4. Paged memory management, demand paged memory management
4.5. Segmented Memory management

UNIT III: Processor Management

CHAPTER 5. Processor Management
5.1. State model
5.2. Job Scheduling
5.3. Process Scheduling
5.4. Multiprocessor system
5.5. Process synchronization

UNIT IV: Device and Information Management

CHAPTER 6. Device management
6.1. Techniques for Device management
6.2. Device management characteristics
6.3. Channels and control units
6.4. Device allocation consideration

CHAPTER 7. Information management
7.1. A simple file system
7.2. General model of a file system
7.3. Symbolic File System
7.4. Basic File System.

Reference Books:
1. Silberschatz A., Galvin P., Gagne G. “Operating Systems Principles”, Willey Eight edition
2. Achyut S. Godbole , Atul Kahate “Operating Systems” McGraw Hill Third Edition
3. “Operating SystemInternal & Design Principles”, William Stallings, Pearson
4. Andrew S. Tanenbaum, “Modern Operating System”, Prentice Hall.

Class :- B.Sc.(C.S.)- S.Y [III-Sem]

Course Title :- Android Operating System **Course Code:-U-ADC-334-A**

Credit :- 02

Learning Objective:- Learn Basic of Android operating system, Learn basic of XML , and learn basic concepts of java programming like Classes, Packages, Thread, Exception Handling etc.

Course Outcome:-Student should write xml code to design android controls , students also able to write java programs with object oriented features, they should create their own packages and able to access created packages.

Syllabus

Unit –I :- Android History and Scope

Chp 1. Introduction to Android

1.1 Need of Mobile Application

1.2 Introduction to Android

1.3 Types of Mobile Applications

1.4 Android Versions

Chp 2. Android Architecture

2.1 Android Architecture

2.2 Linux Kernal

2.3 Dalvik Virtual Machine

UNIT –II:-IDE's and Java Basic

Chp 3. Android IDE's & Components

3.1 Various IDE For Android

3.2 Installtion of Android

3.3 Android Virtual Device

3.4 Android Components

Chp 4. Introduction To Java

4.1 Introduction & History of Java

4.2 Java Applications

4.3 Java Architecture

Unit – III:- Java Programming

Chp 5. Programming Basics

5.1 Variable, Constants

5.2 Hello World Program

5.3 Classes&Inheritance

5.4 Interface

Unit –IV:- Packages & UI Designing

Chp 6. Java Packages & Threads

6.1 Packages

6.2 Thread

6.3 Exception Handling

6.4 Method Overloading

Chp 7. XML &Json

7.1 Tag ,Attribute

7.2 XML

7.3 Json

Reference Books:-

1. The Complete Reference Java2 By Herbert Schildt
2. Java CookBook By Ian Darwin,PublisherO'Reilly

U-LAC-386

Practical Total Marks: 50 **CREDIT: [PR-2]**

Learning Objective:

1.Understand object oriented programming and advanced C++ concepts

1.1 Be able to explain the difference between object oriented programming and procedural programming.

1.2 Be able to program using more advanced C++ features such as composition of objects, operator overloads, dynamic memory allocation, inheritance and polymorphism, file I/O, exception handling, etc.

1.3 Be able to build C++ classes using appropriate encapsulation and design principles.

2. Improve your problem solving skills

2.1 Be able to apply object oriented or non-object oriented techniques to solve bigger computing problems .

2.2 Ultimate goal: to make you a good programmer.

Course Outcomes:

- Gain the basic knowledge on Object Oriented concepts.
- Ability to develop applications using Object Oriented Programming Concepts.
- Ability to implement features of object oriented programming to solve real world problems.
- Use the characteristics of an object-oriented programming language in a program.
- Use the basic object-oriented design principles in computer problem solving.
- Use the basic principles of software engineering in managing complex software project.
- Program with advanced features of the C++ programming language.

Proposed Practical List:

1. Program to demonstrate encapsulation using of class.
2 Program to demonstrate use of array of objects
3 Program to demonstrate use of pointers
4 Program to demonstrate use of pointer to members of class
5 Program to demonstrate use of function overloading
6. Program to demonstrate inline function.
7 Program to demonstrate use of friend function
8. Program to demonstrate static data members & member functions of class.
9 Program to demonstrate use of different manipulators
10. Program to demonstrate objects as function arguments.
11. Program to demonstrate use of constructor, constructor overloading & destructor
12. Program to demonstrate use of all types of Inheritance.
13. Program to demonstrate use of unary & binary operator overloading
14. Program to demonstrate use of polymorphism (virtual function)
15. Program for reading and writing operations on text file.
16. Program to demonstrate command line arguments.

U-LAC-387

Practical Total Marks: 50 **CREDIT: [PR-2]**

Learning Objectives:

- To understand the DOTNET framework andC# language features
- To develop object oriented programs on C#.
- To develop windows based applications on .NET framework.

Course Outcome:

After the successful completion of the course students will be able to:

- Describe basic concepts and develop programs in C# using object oriented features, delegates, events, errors and exceptions
- Interpret data access and develop windows applications
- Explain Common language runtime (CLR) as a platform for managed code
- Describe the features of Common language runtime (CLR) and develop efficient code with C# on .NET framework
- Develop windows based applications & services on .NET framework
- Describe overview of .NET framework
- Apply an understanding of the .NET technology and C#.net components to develop a windows based application which solves specified problem domain
- Use of ADO.NET technology for developing database oriented applications
- Understand the professional responsibility
- Apply an understanding of the need for high ethical standards in the practice of engineering towards people and the environment

Proposed Practical List:

1. Program to demonstrate jagged array
2. Program to demonstrate String class and its methods
3. Program to demonstrate properties concept
4. Program to demonstrate delegate concept
5. Program to demonstrate creation of C# namespaces
6. Program to demonstrate interface concept
7. Program to demonstrate common windows controls
8. Program to demonstrate advance windows controls
9. Program to demonstrate simple database connectivity application
10. Program to perform insert, update and delete operation on database

U-LAC-388

Practical Total Marks: 50 **CREDIT: [PR-2]**

LEARNING OBJECTIVES:

The basic objective of Computer networking is to share resources, files, make connectivity. Computer networking enables computers to share data, application software and hardware devices. The most basic network consists of two computers connected directly by cable, for example sharing resources, such as printers and modems. Any computer capable of communicating on the network is known as a device or node.

COURSE OUTCOMES:

- To master the terminology and concepts of the OSI reference model and the TCP-IP reference model.
- To master the concepts of protocols, network interfaces, and design/performance issues in local area networks and wide area networks.
- To be familiar with wireless networking concepts.
- To be familiar with contemporary issues in networking technologies.
- To be familiar with network tools and network programming.

Proposed Practical List:

1. Perform the practical on my network places.
2. Study the OSI reference model.
3. Stduy the TCP/IP model.
4. Perform the practical on creating user .
5. Perform the practical on creating groups.
6. Perform the practical on sharing files and folders.
7. Perform the practical on IP Config command.
8. Perform the practical on net stat command.
7. Perform the practical on net view command.
8. Perform the practical on net user command.
9. Study the topologies.
10. Study the types of networks.
11. Perform the practical on creating an e-mail id and send the messages and receives the messages.

U-LAC-389

Practical Total Marks: 50 **CREDIT: [PR-2]**

Learning Objective :

1. To introduce students with basic concepts of Operating System, its functions and services.
2. To familiarize the students with various views and management policies adopted by O.S. as pertaining with processes , Deadlock , memory , File and I/O operations.
3. To brief the students about functionality of various OS like , Linux and Windows XP as pertaining to resource management.
4. To provide the knowledge of basic concepts towards process synchronization and related issues.

Course Outcomes :

1. Master functions, structures and history of operating systems
2. Master understanding of design issues associated with operating systems
3. Master various process management concepts including scheduling, synchronization, deadlocks
4. Be familiar with multithreading
5. Master concepts of memory management including virtual memory
6. Master system resources sharing among the users
7. Master issues related to file system interface and implementation, disk management
8. Be familiar with protection and security mechanisms
9. Be familiar with various types of operating systems including Linux.

Proposed Practical List:

1. Study booting process of Windows XP, Linux.
2. To describe the basic function & services of operating systems
3. To explain the block level description of functional units of Operating system
4. To Differentiate Between working of various types of operating systems
5. To identify different types of services and system calls.
6. To describe the fundamentals and technological aspects of File Management.
7. To list and explain different types of disk scheduling algorithms.
8. To describethe File structure.
9. Comparevarious file allocation techniques.
10. Summarize the features of LINUX. (U) 2.Analyze directory system in LINUX.
11. Summarize the features of WINDOWS 7.
12.Analyze directory system, Process management in windows 7.

Course code: U-SOE-481
Course Title: SOFTWARE ENGINEERING

Credit:02

TOTAL TEACHING HOUR : 60 **TOTAL MARKS :50**

LEARNING OBJECTIVE:

The basic objective of software engineering is to develop methods and procedures for software development that can scale up for large systems and that can be used consistently to produce high-quality software at low cost and with a small cycle of time. In software engineering you develop your skills for developing new and useful software's. Main objectives are:
Understanding user conceptual manual and develop better specifications. Improvement in design languages. Reusable codes. Interactive debugging. Mockup to conform specifications.
A. Be employed in industry, government, or entrepreneurial endeavors to demonstrate professional advancement through significant technical achievements and expanded leadership responsibility;
B. Demonstrate the ability to work effectively as a team member and/or leader in an ever-changing professional environment.
C. Progress through advanced degree or certificate programs in computing, science, engineering, business, and other professionally related fields.

COURSE OUTCOME:

An ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability. An ability to function on multi-disciplinary teams. An ability to identify, formulate, and solve engineering problems. An understanding of professional and ethical responsibility. An ability to communicate effectively.

SYLLABUS

UNIT I: ROLE OF SOFTWARE

1.INTRODUCTION TO SOFTWARE ENGINEERING:

The evolving role of software, changing nature of software, software myths. The software problem: Cost, Schedule and quality, scale and change.

2. SOFTWARE PROCESS:

Process and project, component software process.
Software development process models- waterfall model, prototyping , iterative development , relational unified process, time boxing model, Extreme programming and agile process, using process models in a project. project management process.

UNIT II: REQUIREMENT ANALYSIS

3.SOFTWARE REQUIREMENT ANALYSIS AND SPECIFICATION:
Value of good SRS, requirement process, requirement specification, functional specifications with use-cases, other approaches for analysis, validation.

4. PLANNING A SOFTWARE PROJECT:
Effort estimation, project schedule and staffing, quality planning, risk management plans, project monitoring plan, detailed scheduling.

UNIT III: ARCHITECTURE

5. SOFTWARE ARCHITECTURE:
Role of software architecture, architecture view, components and connector view, Architecture styles for C and C view , documenting architecture design, evaluating architectures.

6.DESIGN:
Design concepts, function-oriented design, object oriented design, detailed design, verification, and metrics.

UNIT IV: TESTINGS

7. CODING AND UNIT TESTING:
Programming principles and guidelines, incrementally developing code, managing evolving code, unit testing ,code inspection, and metrics.
Testing: Testing concepts, testing process, black-box testing, white-box testing and metrics.

REFERENCE BOOKS:
1. R.PRESSMAN: Software Engineering- Mc Graw -Hill
2. R.K. Agrawal and Y.Sing: Software Engineering- New Age International.
3. P. Jalote : Software Project Management in practice- Pearson.

Course Title: ASP.NET
Course Code: U-ASN-482

Total Credit :-02

Learning Objective:

- Understand the difference between desktop and dynamic web applications.
- Understand the ASP.NET web application execution model.
- Create simple websites.

Course Outcome:
Develop simple dynamic web applications

Develop simple dynamic web applications

Students can get job of .net web developer and create simple websites.

SYLLABUS

UNIT- I: Introduction to ASP.NeT

1. **Introduction to .NET**
1.1. The .NET Framework.
1.2. .NET languages.
1.3. About ASP.NET

2. Introduction ASP.NET 4.0
2.1. Features of ASP.NET 4.0.
2.2. Stages in Web Forms Processing.
2.3. ASP.Net Web page Life Cycle.
2.4. Conditional Structures, Loop Structures
2.5. Introduction to Server Controls.
2.6. HTML Controls.

UNIT – II: Working in ASP.NET session ,view state and web Server

3. Working in ASP.NET
3.1. Introduction to Session and Viewstate.
3.2. Query String
3.3. Client Server model
3.4. Http Handlers.

4. Web Server and User
4.1. IIS Manager- Creating a virtual directory.
4.2. Virtual directories and Applications.
4.3. Folder Settings.

UNIT – III:

5. Creating Web Forms Applications
5.1. Creating an ASP.NET Web Application Project.
5.2. Responding to Events.
5.3. Where Does Processing Occur?

6. Creating a User Interface
6.1. Validating Data – Using server side Controls.
6.2. Building and using Menus.
6.3. Navigation Between Pages using menu items.

UNIT – IV: Asp.net Application and ADO.NET

7. ASP. NET Applications
7.1. ASP.NET Master Page.
7.2. The Global.asax.
7.3. ASP.NET configuration.

8. Overview of ADO.NET
8.1. ADO.NET architecture.
8.2. Accessing Data using data adapters and datasets.
8.3. Viewing Data Using Grid View

Reference Books
1. Mastering ASP.Net BPB Publication
2. ASP.net – The Complete Reference Tata McGraw Hill
3. The complete Reference ASP.NET by Matthew MacDonald- Tata McGraw-Hill.
4. Professional ASP.NET – Wrox Publication
5. ASP.Net Black Book – Wrox Publication

Website Reference

- **http://tutorialpoint.com**
- **https://www.w3schools.in/java-tutorial**

Course Title: Oracle 10g SQL & PL/SQL
Course Code: U-OSP-483

Total Teaching Hours: 60 **Total Marks: 50**
Total Credit :-02

Learning Objective:

- Execute PL/SQL data type conversion functions
- Display output through PL/SQL programs
- Manipulate character strings in PL/SQL programs

Course Outcome :

- After completing this course, you should be able to:
- Describe the fundamentals of the PL/SQL programming language
- Write and execute PL/SQL programs in SQL*Plus
- Debug PL/SQL programs

SYLLABUS

UNIT - I

1. SQL Statements & Working with tables
1.1. DDL, DML, DQL, DCL
1.2. Data types in SQL
1.3. Creating & Managing Tables
1.4. Manipulating Data
1.5. Retrieving data using SELEC T Command WHERE, Order by, Distinct clause
1.6. Using Column Aliases
1.7. Oracle view
2. Grouping Data in SQL
2.1. Using Group By & Having clause
2.2. Substitution Variables
2.3. Using &, &&, Define, Verify

UNIT – II

3. SQL Functions
3.1. Single Row Functions
3.2. Character Functions, Case Manipulation, Character Manipulation
3.3. Number Functions
3.4. Date Functions
3.5. Conversion Functions

3.6. General Functions

4. Joining Tables &Working with Sub queries

4.1. What is Join?
4.2. Natural Join/Inner Join/Equijoin/self join
4.3. Joining With ‘USING’ Clause
4.4. Joining With ‘ON’ Clause
4.5. Outer Join
4.6. Subqueries: Single Row Sub query, Multiple Row Sub query

UNIT - III

5. Security

5.1. Creating User
5.2. Privileges: System Level Privileges, Object Level Privileges
5.3. Granting Privileges
5.4. Revoking Privileges
5.5. Roles: Study of default roles, Creating roles, Granting and Revoking roles

6. PL/SQL

6.1. An Introduction to PL/SQL
6.2. PL/SQL Overview
6.3. Declaration section
6.4. Executable Commands section
6.5. Condition logic
6.6. Loops

UNIT - IV

7. Advance in PL/SQL

7.1. Exception Handlings
7.2. Triggers: Triggers Syntax, Types of triggers, Enabling and Disabling Triggers, Replacing and Dropping Triggers
7.3. Oracle Cursor concept: Implicit & Explicit cursor
7.4. PL/SQL Procedure & Functions

Reference Books -

1. Oracle Database 10g SQL (Osborne ORACLE Press Series)by Jason price, McGrawHill, 0-07-222981-0.
2. Oracle Database 10g PL/SQL Programming by Scott Urman , Ron HARDMAN, MichaleMc Laughlin, Oracle Press, TMH, ISBN-0-07-059779-0.
3. Oracle Database 10g The Complete Reference By Kevin Loney, Bob Bryla Oracle Press (TATA McGraw Hill Edition) ISBN-13:978-0-07-059425-8, ISBN-10: 0-07-059425-2

Course Title: Multimedia Using Flash
Course Code: U-MUF-484

Total Teaching Hours: 60 **Total Marks: 50**
Total Credit :-02

LEARNING OBJECTIVES:

- To learn the basic concepts of animation as an art.
- To produce exercises as well as a final project in animation using Flash.
- To understand the basic animation techniques and concepts covered in the films and projects viewed in class.

COURSE OUTCOMES:

- To develop an understanding of the tools used for creating two dimensional (2D) Graphics and animation.
- To facilitate the students to design 2D Graphics, 2D character modelling and Animation.
- To make the students create a CBT presentation.

UNIT – I

1. **Exploring The Flash Interface**
1.1. The Flash stage
1.2. Stage Settings
1.3. Creating a new Flash file
1.4. The various import formats
1.5. Timeline- Play head/Frames/Key Frames/ Blank frames
1.6. Menus, Toolbox and Properties
1.7. Color Swatches and Color Mixer
1.8. Rulers, Guides, Grids and Snappings

UNIT - II

2. **Introduction**
2.1. CDROM and Multimedia Highway
2.2. Applications of Multimedia
2.3. Stages of Multimedia Project
3. **Macintosh and Windows Productions Platforms**
3.1. Macintosh Platform
3.2. Windows Platform
3.3. Connections- SCSI and IDE
3.4. Memory and Storage devices
3.5. Input and Output Devices

UNIT - III

4. **Basic Software Tools**
4.1. Text editing and word Processing tools

4.2. Painting and drawing tools
4.3. Image Editing Tools
4.4. Sound Editing Tools
4.5. Font Editing and designing tools
4.6. Hypermedia and Hypertext
4.7. Making Still Images : BITMAPS , Vector Drawing
4.8. Colors, Image file formats

Unit - IV

5. **Animation and Video**
5.1. Principal of Animation
5.2. Making animation that work: Rolling Ball, Bouncing ball
5.3. Using Video
5.4. Broadcast Video Standards
5.5. Recording Formats

References Books:

1. Macromedia Flash MX 2004: The Complete Reference by Brian Underdahl
2. Action Script for Flash MX: The Definitive Guide, 2nd Edition By Colin Moock
3. Macromedia Flash MX 2004 Bible by Robert Reinhardt and Snow Dowd
4. Multimedia : Making it work (5th Editions) By Tay Vaughan (Tataamc)
5. Multimedia : Computing Communications and Applications By Ralf Steinmetz ,Klara Nahrstedt

Recommended Web sites:

1. http://www.webdevelopersnotes.com/tutorials/flash/
2. http://www.adobe.com/devnet/flash/
3. http://www.adobe.com/support/flash/tutorial_index.html
4. http://www.thefreecountry.com/webmaster/flash.shtml

Class B.Sc.C.S. S.Y. (IV Sem)
Course Title:- Mobile Application Development Using Android
Course Code:- U-ADC-434-M

Total Credit :-02 **Marks: 50**

Learning Objective :-Learn Designing of android application, writing java code, joining xml with java, testing application on real mobile device or virtual device, database connectivity etc.

Course Outcome:- Student should design xml controls, join xml controls with java object, run app on real mobile device, create services, works with different java android classes like LocationManager, SensorManager, SQLiteOpenHelper etc.

Students can get job of Android developer or xml-android app UI designer.

SYLLABUS

Unit-I :- Android Basic

Chp1. Android Basic
- 1.1 What is Android?
- 1.2 Activity Life Cycle
- 1.3 Call Back Methods
- 1.4 Logcat usage

Chp2. Android Application Structure
- 2.1 Android Application Structure
- 2.2 AndroidManifest.xml, <Uses-permission>, < uses-sdk>
- 2.3 First Sample Android Application
- 2.4 Activity Registration
- 2.5 Activity & Intent

Unit-II :- Android Widgets

Chp3.UI Widget – I
- 3.1 LinearLayout, RelativeLayout
- 3.2 Button, EditText,TextView
- 3.3 Event Handling
- 3.4 RadioButton , CheckBox
- 3.5 ImageView

Chp 4.UI Widget – II
- 4.1 SeekBar, ProgressBar
- 4.2 Switch
- 4.3 Analog Clock , Digital Clock
- 4.4 ListView

4.5 WebView

Unit-III :- Android Menus &Database Connectivity

Chp 5.Menus & Notifications

5.1 Alert Dialog
5.2 Prompt Dialog
5.3 Android Menus
5.4 Toast Notification
5.5 Custom Toast

Chp6. Services & SQLite

6.1 Android Services
6.2 SQL Commands
6.3 SQLiteOpenHelper Class
6.4 SQLite Based Application

Unit- IV :- Telephony & Mini Projects

Chp7. Telephony & Sensor API

7.1 TelephonyManager
7.2 PhoneCall
7.3 Send SMS
7.4 SensorManager

Chp8. Mini Projects

8.1 Torch App
8.2 Media Player
8.3 Voice To Text Conversion

Reference Books:-

1. Learning Android , OREILLY By:- Marko Gargenta
2. Android Black Book

U-LAC-485

Practical Total Marks: 50 **CREDIT: [PR-2]**

LEARNING OBJECTIVE:

The basic objective of software engineering is to develop methods and procedures for software development that can scale up for large systems and that can be used consistently to produce high-quality software at low cost and with a small cycle of time. In software engineering you develop your skills for developing new and useful software's. Main objectives are:
Understanding user conceptual manual and develop better specifications. Improvement in design languages. Reusable codes. Interactive debugging. Mockup to conform specifications.
A. Be employed in industry, government, or entrepreneurial endeavors to demonstrate professional advancement through significant technical achievements and expanded leadership responsibility;
B. Demonstrate the ability to work effectively as a team member and/or leader in an ever-changing professional environment.
C. Progress through advanced degree or certificate programs in computing, science, engineering, business, and other professionally related fields.

COURSE OUTCOME:

An ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability. An ability to function on multi-disciplinary teams. An ability to identify, formulate, and solve engineering problems. An understanding of professional and ethical responsibility. An ability to communicate effectively.

Proposed Practical List:

1. Perform the practical on Requirement gathering.
2. Perform the practical on Requirement Specification.
3. Perform the practical on Requirement Analysis.
4. Practical on Designing the software.
5. Creating the Login form.
6. Testing the all forms.
7. Practical on designing the Tables.
8. Testing the all the tables.
9. Perform the practical on Database connectivity.
10. Perform the practical on Testing's
11. Perform the practical on Error testing.

U-LAC-486

Practical Total Marks: 50 **CREDIT: [PR-2]**

Learning Objective:

- Understand the difference between desktop and dynamic web applications.
- Understand the ASP.NET web application execution model.
- Create simple websites.

Course Outcome:
Develop simple dynamic web applications

Develop simple dynamic web applications

Students can get job of .net web developer and create simple websites.

Proposed Practical List:

1. **Program to demonstrate how to create html home page**
2. **Program to demonstrate implementing CSS**
3. **Program to demonstrate simple how to create Asp.net home page**
4. **Program to demonstrate Html controls(label, button textbox)**
5. **Program to demonstrate server side controls(label, button textbox)**
6. **Program to demonstrate Calendar control.**
7. **Program to demonstrate Master page**
8. **Program to demonstrate different validation controls**
9. **Program to demonstrate session and Query string**
10. **Program to demonstrate building simple web application.**

U-LAC-487(Oracle 10g SQL and PL/SQL)

Practical Total Marks: 50 **CREDIT: [PR-2]**

Learning Objective:

- Execute PL/SQL data type conversion functions
- Display output through PL/SQL programs
- Manipulate character strings in PL/SQL programs

Course Outcome :

- After completing this course, you should be able to:
- Describe the fundamentals of the PL/SQL programming language
- Write and execute PL/SQL programs in SQL*Plus
- Debug PL/SQL programs

Proposed Practical List:

1. STUDY OF DDL STATEMENTS
2. STUDY OF DML STATEMENTS
3. STUDY OF DCL STATEMENTS
4. STUDY OF SELECT STATEMENTS WITH DIFFERENT OPERATORS.
   (Arithmetic Operators ,Logical Operators, Range Searching, Pattern Matching, Column Alias)
5. STUDY OF Data Constraints.
6. STUDY OF ORACLE FUNCTIONS.
7. STUDY OF ORACLE JOIN& SUBQUERIES.
8. STUDY OF Grouping Data from tables.
9. STUDY OF ORACLE VIEWS.
10. INTRODUCTION OF PL/SQL.
11. LOOPING & CONDITIONAL STRUCTURES.
12. ORACLE CURSORS
13. ORACLE STORED PROCEDURES & FUNCTIONS.
14. ORACLE TRIGGERS
15. ORACLE EXCEPTIONAL HANDLAERS.

U-LAC-488

Practical Total Marks: 50 **CREDIT: [PR-2]**

LEARNING OBJECTIVES:

- To learn the basic concepts of animation as an art.
- To produce exercises as well as a final project in animation using Flash.
- To understand the basic animation techniques and concepts covered in the films and projects viewed in class.

COURSE OUTCOMES:

- To develop an understanding of the tools used for creating two dimensional (2D) Graphics and animation.
- To facilitate the students to design 2D Graphics, 2D character modelling and Animation.
- To make the students create a CBT presentation.

Proposed Practical List:

1. Study of Flash environment.
2. Working with shape tween by designing A to Z characters animation.
3. Working with shape motion by designing A to Z characters animation.
4. Designing Bouncing Ball animation.
5. Creating a commercial Advertisement.
6. Creating a Jumping Man animation.
7. Creating a Flying Bird animation.
8. Creating a Walking Man animation.
9. Creating a Riding Bike animation.
10. Creating a Flying Honey Bee animation.
11. Creating a growing Plant animation.
12. Working with shape tween with blending options.