Rajarshi Shahu Mahavidyalaya, Latur (Autonomous)
Department of Computer Science
Curriculum Structure with effect from June,2019
Syllabus (2022-23)

	Course Code	Core Course	Title of the course with paper number	Hours/ Week	Marks (50)		Credits
					Internal	End Semester	
B. Sc. III (Computer Science) Semester V	U-COS-541	DSEE-I	Relational Database Management System - IX	03	20	30	02
	U-COS-542	DSEE -II	Programming in C#.net – X A or Programming in VB. Net –X B	03	20	30	02
	U-COS-543	DSEEP-I	Laboratory Course –VII (RDBMS)	03	20	30	01
	U-COS-544	DSEEP-II	Laboratory Course -VIII (DSE.)	03	20	30	01
		SECCOS-3	Web Page Designing	03	25	25	2
			Total Credits				**06+02**

Student Stay Hours: 15/Week

Rajarshi Shahu Mahavidyalaya, Latur (Autonomous)
Department of Computer Science
Curriculum Structure with effect from June, 2019

	Course Code	Core Course	Title of the course with paper number	Hours/ Week	Marks (50)		Credits
					Internal	End Semester	
B.Sc. III (Computer Science) Semester - VI	U-COS-641	DSEE-I	Web Programming using PHP and MySQL - XI	03	20	30	02
	U-COS-642	DSEE-II	Introduction to Python Programming – XII A Or Software Engineering - XII B	03	20	30	02
	U-COS-643	DSEEP-I	Laboratory Course-IX (PHP and MySQL)	03	20	30	01
	U-COS-644	DSEEP-II	Laboratory Course-X-A Or Laboratory Course-X-B	03	20	30	01
	SECCOS-4		Introduction to SCILAB	03	25	25	2
			Total Credits				**06+02**

Student Stay Hours: 15/Week

B. Sc. – III [Computer Science] Semester V
Course: Relational Database Management System
Course Code: U-COS-541
DSEE-I
Paper- IX

Teaching Hours: 45   **Marks: 50**

Learning Objectives:
- To design the tables in DBMS
- To write queries to get optimize outputs
- To store, retrieve and view the contents
- To generate report based on customized need

Course Outcomes:
After successful completion of this course, students should be able to –
- Analyze the core terms, concepts, and tools of relational database management systems.
- Identify the information that is needed to design a database and create conceptual and logical database designs for a business information problem.
- Construct a database that satisfies relational theory and provide users with database related queries, business forms, and business reports.

SYLLABUS

UNIT I: Introduction to Database and Elements of DBMS   [10hrs]
Definition of DBMS, File processing Vs DBMS, Advantages and disadvantages of DBMS
Users of DBMS, DBMS Structure,
DBMS Languages: DDL, DML, DCL , **Terms**: Entity, Entity set, attributes. **Keys**: Primary, secondary, foreign, composite

UNIT II: Data Models and Relational Algebra and Calculus   [13hrs]
Introduction, Object based logical model, Record based logical model (RDB, NDB, HDB), E-R model, E-R diagram, Introduction Relation, Schemes, Domain, Tuples, Cardinality degree, Algebraic operation.
Fundamental operation: Select, product, union.
Set difference: Natural join, Cartesian product, rename.
Relational calculus: Tuple and domain relational calculus.

UNIT III: Relational Database Design and SQL   [12 hrs]
Normalization: 1NF, 2NF, 3NF, BCNF, Class diagrams and E-R tables.
Functional dependency, Data types, Table Creation, Modify, Selecting, Deleting records,
Simple queries, Oracle constraints.

UNIT IV: Use of Operators and Advance in SQL   [10 hrs]
Comparison operators: Between, In, Not In, Like, Null
Logical operators: AND, OR, NOT SQL function, Joins, Sub-queries, Views.

Reference Books:
1. Database System Concepts by Abraham Silberschatz, Henry Korth, and S. Sudarshan

2. Database Management Systems by Raghu Ramakrishnan

B. Sc. – III [Computer Science] Semester V
Course: Relational Database Management System
[Laboratory Course]
Course Code: U-COS-543
DSEEP-I
Paper-VII

Learning Objectives:
- To design the tables in DBMS
- To write queries to get optimize outputs
- To store, retrieve and view the contents
- To generate report based on customized need

Course Outcomes:
After successfully completion of this course, students should be able to –
- Identify the information that is needed to design a database management system and Create conceptual and logical database designs for a business information problem.
- Construct a database management system that satisfies relational theory and provides users with business queries, business forms, and business reports.
- Analyze the core terms, concepts, and tools of relational database management systems.

Proposed Practical List:
Create following tables and put proper constraints whenever required:
Employee (eid, Name, Qualification, Dsg, Dept, DOJ, Experience)
Accounts (eid, salary, DA, gross)
Training (eid, domain, domain_id, course, duration)

Insert data in the above tables.
Solve the following queries using SQL.
1. Select name of all the employees in Administration Department.
2. Select Employee No. and Qualification of all Employees in Sales Department.
3. Select Employee No. of all Employees whose Salary is in range 5000 to 10000.
4. Select Employee Name of all Employees whose Designation is "Manager".
5. Select Employees name whose experience is greater than 10 years.
6. Select eid from Accounts table whose salary is greater than 5000 or gross is greater than 7000
7. Count the number of Employee Name from Employee table
8. Find the sum of all salary of all Employee from Accounts table.
9. Select the domain from Training table whose domain_id =5 and course = dbms
10. Find the min and max number of Salary from Accounts table.
11. Add column named "city" in the Employee table with datatype as varchar
12. Delete column named "duration" from Training table.
13. Select distinct Dept from Employee table.
14. Truncate the table Training.
15. Insert into Accounts table eid=44, salary as 40000 DA as 5000 and gross 45000

B. Sc. – III [Computer Science] Semester V
Course: Programming in C#.net
Course Code: U-COS-542
DSEE-II
Paper-X A

Teaching Hours: 45   **Mark: 50**

Learning Objectives:
- Working with Visual Studio
- Designing Forms and writing code
- To Create Database Connectivity

Course Outcomes:
After successful completion of this course, students should be able to –
- Handle Visual Studio.
- Design form with menus, controls and write code.
- Work with Advance Controls
- Connect Front End with Back End
- Perform DML Operation

UNIT I: Introduction to .net, Arrays & Operators   [10 hrs]
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 java, I/O Statement with C#.net, Boxing & Unboxing, Short Circuiting Operator, Array & ArrayList class, Jagged Array, Hash Table, String Class.

UNIT II: Properties, Error Handling & Namespaces   [10 hrs]
Properties & its type, Event, Delegate & Multicast Delegate, Thread, Exception handling, using keyword, creating and using namespaces, interface, Method overloading & method overriding, Partial Class.

UNIT III: GUI Programming (Windows Application)   [13 hrs]
 Event Driven Programming, Building windows application with visual studio
TextBox, Label & Button Control, Combo Box, List Box, Check Box & Group Box Control
DateTimePicker, Timer Control, Tree View, Building Menu, MDI Form, Picture Box, Progress Bar Control, Common Dialog boxes, Introduction to WPF.

UNIT IV: Database Programming   [12 hrs]
How Ado.net differs from Ado, Advantages of Ado.net, Connected & Disconnected Architecture, Dataset, Data Reader & Data Adapter
Managed Data Providers, Data Grid View Control, 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. .net 4.0 programming black book by Kogent Learning Solutions Inc.
3. C# 2010 programming black book by Kogent Learning Solutions Inc.

B. Sc. – III [Computer Science] Semester V
[Laboratory Course]
Course: Programming in C#.net
Course Code: U-COS-544
DSEEP-VIII

Learning Objectives:
- Working with Visual Studio
- Designing Forms and writing code
- To Create Database Connectivity

Course Outcomes:
After successful completion of this course, students should be able to –
- Handle Visual Studio
- Design form with menus, controls and write code
- Work with Advance Controls
- Connect Front End with Back End
- Perform DML Operation

Practical List: -

1. Introduction to VisualStudio.net
2. Console application for I/O statement
3. Console application for each loop with Array class
4. Console application for Hash Table
5. Console application for Read Write properties
6. Console application for Exception Handling and Thread
7. Windows application for MDIform and Create Menus
8. Windows application to demonstrate Message Box and TextBox
9. Windows application to work with ComboBox and ListBox
10. Windows application to work with Progress Bar and Timer
11. Windows application to demonstrate Dialog Box
12. Windows application to work with Tree View
13. Windows application to connect with MS-Access and Oracle
14. Windows application to Perform DML operation on Table
15. Windows application to show database record in Data Grid View

Teaching Hours:45   Marks: 50

Learning Objectives:
- The student will use Visual Basic.Net to build Windows applications using structured and object-based programming techniques.
- Students will be exposed to the following concepts and skills
- Analyze program requirements
- Design/develop programs with GUI interfaces
- Code programs and develop interface using Visual Basic.Net
- Perform tests, resolve defects and revise existing code

Course Outcomes:
- After successful completion of this course, students should be able to –
- Understand .NET Framework and describe some of the major enhancements to the new version of Visual Basic.
- Describe the basic structure of a Visual Basic.NET project and use main features of the integrated development environment (IDE)
- Create applications using Microsoft Windows Forms
- Create applications that use ADO.NET

UNIT I: Introduction to .Net Technology and Visual Basic.Net IDE   [13 hrs]
Introduction to .Net, Features of .Net, Advantages of .Net, Net Framework, CLR, CTS, CLS, Creating a project, Types of project in .Net, Exploring and coding a project, **Solution explorer** - toolbox, properties window, Output window, Object Browser.

UNIT II: Programming with VB.Net and Understanding Console Application
                                                                                                        [10 hrs]
Variables, constants, operators, data types, working with string, Methods.
Control statements: Making decisions, if statement, Select case, Loops MsgBox and Input Box, Classes and Objects, **Access Specifiers**: Private, Public and Protected, Building Classes, Constructors, Inheritance types, Overloading and Overriding.

UNIT III: GUI Programming   [13 hrs]
Introduction to Window Applications, Using Form – Properties, Methods and Events,
Interacting with controls - Textbox, Label, Button, Listbox, Combobox, Checkbox, Picture Box, Radio Button, GroupBox, Timer, toolbar, Progress Bar,
Common Dialog Controls (Save, Open, Font, Color).

UNIT IV: Introduction to ADO.Net   [9 hrs]
Connected and disconnected Object Model
Creating Connection, Command, Data Adapter, Data Reader and Data Set with OLEDB, Insertion and Updating with table.

Reference Books:

1. VB.Net programming Black Book, by Kogent Learning Solutions.
2. VB.Net Step By Step by Michael Halvorson, Wiley India, PHI.
3. Mastering VB.Net by Evangelos Petroutsos,
4. Beginning VB.Net by Wiley Publications (Wrox)

B. Sc. – III [Computer Science] Semester V
[Laboratory Course]
Course: Programming in VB.NET
Course Code: U-COS-544
DSEEP-VIII

Learning Objectives:
- The student will use Visual Basic.Net to build Windows applications using structured and object-based programming techniques.
- Design/develop programs with GUI
- Code programs and develop interface using Visual Basic.Net

Course Outcomes:
After successful completion of this course, students should be able to –
- Understand .NET Framework and describe some of the major enhancements to the new version of Visual Basic.
- Understand the basic structure of a VisualBasic.NET project and use main features of the integrated development environment (IDE)
- Create applications using Microsoft Windows Forms.
- Create an application that contains use of ADO.NET

Practical List
1. Study of Integrated development environment in vb.net.
2. Study of project types in vb.net
3. Creating a project in VB.NET
4. Study of form with its all properties and methods.
5. Programs to demonstrate textbox, label and command buttons
6. Programs to demonstrate InputBox and MsgBox.
7. Programs to demonstrate Radiobutton , Checkbox and List
8. Programs to demonstrate Operators in vb.net using different controls.
9. Programs to demonstrate if-else, elseif and switch statements in vb.net.
10. Programs to demonstrate lopping statements in vb.net.
11. Programs to demonstrate listbox.
12. Programs to demonstrate ComboBox.
13. Program to demonstrate use of arrays.
14. Programs to demonstrate string handling functions.
15. Programs to demonstrate database connectivity.

B. Sc. – III [Computer Science] Semester V
Skill Enhancement Course
(Web Page Designing)
SECCOS – 3

Course Objectives:
- Create static web sites using HTML5, CSS3, BOOTSTRAP4, JS
- Create Responsive web pages with form validation
- Effective Use of Visual Studio Code editor

Course Outcomes:
After successful completion of this course, students should be able to –
- Design basic programming structures to implement functionality for web page designing.
- Develop static websites.
- Develop responsive websites with Bootstrap.
- Develop websites independently.

UNIT I: Introduction to HTML5
Formatting text by using tags, using lists and backgrounds.
Creating hyperlinks and anchors, Creating tables, creating simple table, specifying the size of the table, specifying the width of the column, merging table cells, using tables for page layout, Formatting tables, applying table borders, applying background and foreground fills, changing cell padding, spacing and alignment, creating user forms creating basic form, using check boxes and radio buttons creating lists, additional input types in HTML5, Incorporating sound and video, Audio and video in HTML5, HTML multimedia basics, embedding video clips, incorporating audio on web page, Image Mapping.

UNIT II: CSS3
Introduction to CSS, how does CSS work? syntax, identification and grouping of elements, selectors, colors, background, fonts, text, links, lists, tables. CSS Box model, Margin, Padding, Border, height and width, floating elements, positioning of elements, align, dropdowns, navigation bar, counters, Image gallery.

UNIT III: Java Script, Bootstrap
Introduction to Client Side Scripting, Introduction to Java Script, JavaScript Types, Variables in JS, Operators in JS, Conditional statements, Java Script Loops, JS Popup Boxes, JS Events, JS Arrays, Working with Arrays, JS Objects, JS Functions, Document and its associated objects, Document, Link, Area, Anchor, Image, Applet, Layer Events and Event Handlers, Using Java Script in Real-time, Validation of Forms.
What is Bootstrap?
Basic Bootstrap Pages, Bootstrap Grid System, Grid Classes, and Basic Structure of a bootstrap Grid, Equal Columns and Unequal Columns.
Typography, Table, Images, Wells, Alerts Button, Button groups, Badges/Labels Progress Bars, Pagination, pager, List groups, Panels, Drop Down, Collapse, Tabs/Pills, Navbars

UNIT IV: Development of Website

Website should contain minimum five web pages.
Website may contain Java script for validation, Bootstrap for responsiveness.

Reference Books:

1) Start Here Learn HTML5, Faithe Wempen, Microsoft Publication
2) HTML and CSS Design and Build Websites, John Duckett, Willy Publications
3) JavaScript 2.0: The Complete Reference, Second Edition by Thomas Powell and Fritz Schneider
4) Internet & Web Development, Soma Das Gupta, Khanna Publishing House

Online Reference:

www. W3schools.co

Teaching Hours:45   **Marks:50**

Learning Objectives: -
- Designing of Web Pages.
- Writing PHP code
- Join HTML with PHP,
- Testing application on Browsers, Object Oriented Programming with php, database Connectivity etc.

Course Outcomes: -
After successfully completion of this course, students should be able to –
- Student should design Web Pages.
- Writing HTML page with PHP Code.
- Run webpage on various browsers.
- Create php classes.
- Works with different php classes.
- Perform database connectivity with Mysql
- Student can work as web developer or web designer.

UNIT- I: PHP Basic and Programming Constructs.   [10 hrs]
HTTP basics, introduction to Web server and Web browser,
Introduction to PHP, advantages of PHP.
PHP syntax, Variables, Constants, print/Echo Data types, Strings, Constants, while,
For, Operators and Expressions, if-else, else if, Switch statement, Break and Continue.

UNIT- II: Function & Arrays   [13 hrs]
Defining and calling a function, Default parameters, Variable parameters, Variable function, Types of strings in PHP, Printing functions, Encoding and escaping, Comparing strings, Manipulating and searching strings, Regular expressions, Indexed Vs Associative arrays, Multidimensional arrays, Converting between arrays and variables, Traversing arrays, Using arrays, Functions: count, list, in array, current, next, previous, end, each, sort, rsort, assort, array_merge, array_reverse.

UNIT- III: OOPs and Forms   [12 hrs]
Classes, Objects, Serialization, Inheritance, Interfaces, Encapsulation.
FORM element, INPUT elements, Validating user input, Passing variables between pages, Passing variables through a GET, Passing variables through a POST, Passing variables through a REQUEST, MVC Architecture.

UNIT- IV: My SQL   [10 hrs]

Introduction to MySql, Benefits of MySql, reserve words, key words, variables, data types,

Types of commands: Data Definition Commands, Data Manipulation Commands, Data Control

Commands Clauses: where, order by, group by, having, like, between, Connectivity with PHP.

Reference Books:

1. Programming PHP Rasmus Lerdorf and Kevin Tatroe O'Reilly publication

2. Beginning PHP 5 Wrox publication

3. PHP and MYSQL O'Reilly publication

Web References

1. www.php.net.in

2. www.W3schools.com

3. www.wrox.com

B. Sc. – III [Computer Science] Semester VI
Course: Web Programming using PHP and MySQL
[Laboratory Course – IX]
Course Code: U-COS-643
DSEEP-I
Paper-IX

Learning Objectives:
- Learn Designing of Web Pages.
- writing PHP code, Join HTML with PHP,
- Testing application on Browsers, Object Oriented Programming with php, MySql Connectivity etc.

Course Outcomes: -
After successful completion of this course, students should be able to –
- Student should design Web Pages.
- Writing HTML page with PHP Code.
- Run web page on various browsers.
- Create php classes.
- Works with different php classes.
- Perform database connectivity with Mysql
- Student can work as web developer or web designer.

Practical List:
1. Installing xampp/wamp.
2. Write a program for
    a. Print Hello world
    b. Comments
    c. Printing statement using echo and print
3. Write a program for
    a. Declaring data types
    b. Printing data types of variable
4. Write a program for conditional statements.
5. Write a program for looping statements.
6. Write a program for
    a. Operators
    b. String functions
7. Write a program for defining and calling function.
8. Write a program for
    a. Default parameter function
    b. Variable function
9. Write a program for
    a. Variable parameter
    b. Encoding and decoding
    c. Printing function
10. Write a program for
    a. Indexed array
    b. Associative array
    c. Multidimensional array

11. Write a program for classes and objects.
12. Write a program for passing variables between pages using

      a. GET method

      b. POST method

      c. REQUEST method

13. Study of DDL, DML command.
14. Study of MYSQL clauses- WHERE, ORDER BY, GROUP BY, HAVING, LIKE, BETWEEN.
15. Database connectivity.

Teaching Hours: 45   **Marks:50**

Learning objectives:
- To acquire programming skills in core Python.
- To be aware about the data types, looping structure
- To be able to know the use of string, lists, dictionary and tuples
- To learn the concepts of Exception handling and file handling
- To acquire Object Oriented Skills in Python

Course Outcomes:
After successful completion of this course, students should be able to –
- Write python programs that use strings, lists, tuples and dictionaries
- Demonstrate the concepts of Object-Oriented Programming using python programs
- Write python programs that stores and manipulates data using file
- Implement python program that uses regular expressions and exception handling

UNIT I: Getting Started [15 hrs]
Introduction, History, important features, overview of python and installation, Lexical Matters: Lines, Comments, Names and Tokens, Doc Strings, Simple Program, Identifiers, Reserved Words, Multi-Line Statements, Operators, variables, assignment, Numbers (int, long, float and complex), Strings. Decision and Looping Statements, Introduction to decision statement, If Statement, if—else statement, if-elif-else statement. Introduction to Looping statement, while loop, for loop, nesting of loop, break, continue and pass statement.

UNIT II: Sequence: String, List, Tuples and Error, exceptions [8 hrs]
Strings, Strings and Operators, String Built-in methods, Lists, List type built-in method, Tuples, Special features of Tuples, Dictionary, what are exceptions? exceptions in Python, Detecting and handling exceptions, raising exception, Assertions, Standard exceptions, creating exceptions.

UNIT III: Functions, Class and OOPs [13 hrs]
What are functions, calling functions, creating functions, passing functions, formal arguments, positional arguments, default arguments, variable length argument, recursion, Introduction to OOP, Classes, Class attributes, Instances, Instance attribute, building and Method of invocation, Sub classing and derivation, Inheritance, Built-in functions for classes, instances and other objects privacy.

UNIT IV: Graphical Interfaces [9 hrs]
Graphical user interfaces, event-driven programming paradigm, tkinter module, creating simple GUI, button, labels, entry, dialogs, widget attribute – sizes, fonts, color layouts, nested frames.

Reference Book:
1. Core Python Programming – Wesley J. Chun, Printice Hall PTR, First edition.
2. Learning To Program with Python - Richard L. Halterman.

B. Sc. – III [Computer Science] Semester VI
Course: Introduction to Python Programming
[Laboratory Course]
Course Code: U-COS-644
DSEEP-II
Paper-XA

Learning objectives
- To acquire programming skills in core Python.
- To acquire Object Oriented Skills in Python
- To develop the skill of designing Graphical user Interfaces in Python
- To develop the ability to write database applications in Python

Course Outcomes:
After successfully completion of this course, students should be able to –
- Explain basic principles of Python programming language.
- Implement Object Oriented concepts.
- Implement database and GUI applications.

PRACTICAL LIST:

1) Program to demonstrate Constant Variable.

2) Program to demonstrate scope of Variable

3) Program to demonstrate branching statement

4) Program to demonstrate Looping statement

5) Program to demonstrate simple class

6) Program to demonstrate String class and its method.

7) Program to demonstrate String Buffer and its method.

8) Program to demonstrate inheritance and its Types

9) Program to demonstrate recursion

10) Program to demonstrate function

11) Program to demonstrate class and object

12) Program to demonstrate Inheritance

13) Program to demonstrate GUI

14) Program to demonstrate tkinter module

15) Program to demonstrate widget attribute

Teaching Hours: 45   **Marks: 50**

Learning Objectives:
- Knowledge of basic software engineering methods and practices, and their appropriate application.
- A general understanding of software process models such as the waterfall and evolutionary models.
- Understanding of the role of project management including planning, scheduling, risk management etc.
- Understanding of approaches to verification and validation including static analysis, and reviews.
- Understanding of software testing approaches such as unit testing and integration testing.

Course Outcomes:
   After successful completion of this course, students should be able to –
- Basic knowledge and understanding of the analysis and design of complex systems.
- Apply software engineering principles and techniques.
- Develop, maintain and evaluate software systems.
- Work as an effective member or leader of software engineering teams.
- Understand and meet ethical standard s and legal responsibilities

UNIT I: Software Engineering Fundamentals and Software Process   [13 hrs]
Definition of Software, Software characteristics, Software Applications Software Process Models, Waterfall model, Prototyping model, Spiral model, Incremental model, Concurrent development model
Project management Concepts: The Management Spectrum - The People, The Product, The Process, and The Project.

UNIT II: Software Process and Project Metrics: [10 hrs]
Measures, Metrics and Indicators,
Software measurement: Size - Oriented Metrics, Function - Oriented Metrics, Extended Function point metrics, Software Project Planning, Project Planning Objectives, Software Project Estimation, Decomposition Techniques - Problem Based Estimation, Process Based Estimation, Empirical Estimation Models- The COCOMO Model

UNIT III: Risk Analysis and Management and Software Quality Assurance
[12 hrs]

Software risks, Risk identification, Risk Projection, Risk Refinement Risk Mitigation, Monitoring and Management.

Basic concepts- Quality, Quality Control, Quality Assurance, Cost of Quality Software Quality Assurance (SQA), Formal Technical Review.

UNIT IV: Coding and Testing   [10 hrs]

Programming principles and guidelines, incrementally developing code, Managing, evolving code, Unit testing, Code inspection, Testing concepts, Testing process Black-box testing, White-box testing.

Reference Books:

1. R. Pressman: Software Engineering, McGraw-Hill.
2. K.K. Agrawal and Y. Sing: Software Engineering, New Age International.
3. P. Jalote: Software Project Management

B. Sc. – III [Computer Science] Semester VI
Course: Software Engineering –XII B
[Laboratory Course]
Course Code: U-COS-644
DSEEP-II
Paper-XII B

Learning Objectives:
- Knowledge of basic software engineering methods and practices, and their appropriate application.
- A general understanding of software process models such as the waterfall and evolutionary models.
- Understanding of the role of project management including planning, scheduling, risk management, etc.
- Understanding of approaches to verification and validation including static analysis, and reviews.
- Understanding of software testing approaches such as unit testing and integration testing.

Course Outcomes:
After successfully completion of this course, students should be able to –
- Basic knowledge and understanding of the analysis and design of complex systems.
- Apply software engineering principles and techniques.
- Develop, maintain and evaluate software systems.
- Work as an effective member or leader of software engineering teams.
- Understand and meet ethical standards and legal responsibilities

Practical List:
1. Study of Problem Statement
2. Study of Process Model
3. Creating a Data Flow Diagram
4. Study of Data Dictionary
5. Computing FP
6. Calculate Efforts
7. Decide Schedule
8. Creating Risk Table
9. Creating Timeline chart
10. Creating Architectural Design
11. Creating Data Design
12. Creating Data Flow Design
13. Study of COCOMO Model
14. Study of White Box Testing
15. Study of Black Box Testing

Learning Objectives:

1) To learn Scilab installation on different O.S.
2) To learn Batch processing
3) To learn datatypes and variables of Scilab
4) To learn Matrix, control and looping statements

Course Outcomes:

After successfully completion of this course, students should be able to –

- Install the Scilab on different operating system.
- Perform Batch processing operations
- Use scilab variables in problem domain.
- Perform Matrix manipulations, control and looping statements.

1. Overview

Introduction to Scilab, Installation on Windows & Linux Getting Help from Scilab Exercise

2. Getting Started Scilab

The console, The editor, Docking, Using exec Batch processing

3. Basic elements of the language

Creating real variables, Variable names, Comments and continuation lines Elementary mathematical functions, pre-defined mathematical variables, Booleans, Complex numbers, Integers, Floating point integers, The ans variable, Strings, Dynamic type of variables.

4. Matrices

Working with Matrix, Multiplication of two vectors, Comparing two real matrices

5. Control & Looping Statement

if statement, select statement, for statement, while statement, break and continue statements

Reference:

1. The Scilab Consortium.Scilab.http://www.scilab.org
2. SCILAB (A Free Software To MATLAB) KindleEdition by ACHUTHSANKAR S. NAIR(Author)