<table>
<thead>
<tr>
<th>S. No.</th>
<th>Course No.</th>
<th>Subject</th>
<th>L:T:P</th>
<th>Hours / Week</th>
<th>Examination Schedule (Marks)</th>
<th>Duration of Exam (Hrs)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Major Test</td>
<td>Minor Test</td>
</tr>
<tr>
<td>1</td>
<td>CSE 302N</td>
<td>Compiler Design</td>
<td>3:1:0</td>
<td>4</td>
<td>75</td>
<td>25</td>
</tr>
<tr>
<td>2</td>
<td>CSE 304N</td>
<td>Essential of Information</td>
<td>3:1:0</td>
<td>4</td>
<td>75</td>
<td>25</td>
</tr>
<tr>
<td>3</td>
<td>CSE 306N</td>
<td>Mobile Computing</td>
<td>3:1:0</td>
<td>4</td>
<td>75</td>
<td>25</td>
</tr>
<tr>
<td>4</td>
<td>CSE 308N</td>
<td>Web Technology</td>
<td>3:1:0</td>
<td>4</td>
<td>75</td>
<td>25</td>
</tr>
<tr>
<td>5</td>
<td>CSE 310N</td>
<td>Software Engineering</td>
<td>3:1:0</td>
<td>4</td>
<td>75</td>
<td>25</td>
</tr>
<tr>
<td>6</td>
<td>HS 303N</td>
<td>Business Intelligence and</td>
<td>4:0:0</td>
<td>4</td>
<td>75</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Entrepreneurship</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>CSE 312N</td>
<td>Web Technology Lab</td>
<td>0:0:3</td>
<td>3</td>
<td>0</td>
<td>40</td>
</tr>
<tr>
<td>8</td>
<td>CSE 314N</td>
<td>Essential of Information</td>
<td>0:0:3</td>
<td>3</td>
<td>0</td>
<td>40</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Technology Lab</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>CSE 316N</td>
<td>Software Engineering Lab</td>
<td>0:0:3</td>
<td>3</td>
<td>0</td>
<td>40</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td>33</td>
<td>450</td>
<td>270</td>
<td>180</td>
</tr>
</tbody>
</table>
CSE-302N  Compiler Design

<table>
<thead>
<tr>
<th>Lecture</th>
<th>Tutorial</th>
<th>Practical</th>
<th>Major Test</th>
<th>Minor Test</th>
<th>Total</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>1</td>
<td>-</td>
<td>75</td>
<td>25</td>
<td>100</td>
<td>3 Hrs.</td>
</tr>
</tbody>
</table>

Purpose  At the end of the course, the student will be able to design and implement a compiler.

Course Outcomes (CO)

| CO1 | To understand, design and implement a lexical analyzer. |
| CO2 | To understand, design and implement a parser. |
| CO3 | To understand, design code generation schemes. |
| CO4 | To understand optimization of codes and runtime environment |

UNIT I

Introduction to Compiling
Analysis of the source program, Phases of a compiler, Cousins of the Compiler, Grouping of Phases, Compiler construction tools.
Lexical Analysis – Regular Expression, Introduction to Finite Automata and Regular Expression, Conversion of Regular Expression to NFA, Role of Lexical Analyzer, Input Buffering, Specification of Tokens.

UNIT II

Syntax Analysis
Role of the Parser, Writing Grammars, Symbol Table, Context-Free Grammars, Top Down Parsing with or without Backtracking, Recursive Descent Parsing, Non-Recursive Descent Parsing, SLR Parser, Canonical LR Parser, LALR Parser.

UNIT III

Intermediate Code Generation and Code
Intermediate languages, Declarations, Assignment Statements, Boolean Expressions, Case Statements, DAG representation of Basic Blocks, A simple Code generator from DAG, Issues in the design of code generator, The target machine, Runtime Storage management, Error Handling - Type checking,

UNIT IV

Code Optimization and Run Time Environments

TEXT BOOK

REFERENCES
Focus Area 1: Object Oriented Programming using Java

Unit I:

Unit II:
Programming Basics: Identifiers, variables, data types, operators, control structures, type conversion, casting, arrays, strings
Object Oriented Concepts fundamentals: class & object, instance variables & methods, access specifiers, reference variables, parameter passing techniques, constructors, this reference, static, and command line arguments
Introduction to UML: Use case diagrams – Class diagrams

Unit III:
Relationships: aggregation, association, Inheritance, types of inheritance, Static Polymorphism: method overloading, constructor overloading, Dynamic polymorphism: method overriding, abstract, interface, introduction to packages Industry Coding Standards and Best Practices, code tuning & optimization, clean code & refactoring

Focus Area 2: Relational Database Management System
Unit IV:
RDBMS- data processing, the database technology, data models, ER modelling concept, notations, converting ER diagram into relational schema, Logical database design, normalization (1NF, 2NF and 3NF)
SQL: DDL statements, DML statements, DCL statements, Joins, Sub queries, Views, Database design Issues, SQL fine-tuning

Books on Java
2. Programming with Java 3e A Primer by E Balagurusamy
3. Introduction to Java Programming by K. Somasundaram, Jaico Publishing House; 1 edition

Books on RDBMS, Oracle, MYSQL
5. Schaum's Outline of Fundamentals of Relational Databases by Ramon Mata-Toledo, Published November 15th 2000 by McGraw-Hill
**UNIT – I**

Introduction, issues in mobile computing, overview of wireless telephony: cellular concept, Mobile computing Architecture, Design considerations for mobile computing, Mobile Computing through Internet, Making existing applications mobile enabled. GSM: air-interface, channel structure, location management: HLR-VLR, hierarchical, handoffs, channel allocation in Cellular systems, WCDMA, GPRS 3G, 4G.

**UNIT – II**


**UNIT – III**

Data management issues, data replication for mobile computers, adaptive clustering for mobile wireless networks, File system, Disconnected operations Mobile Agents computing, security and fault tolerance, transaction processing in mobile computing environment.

Cloud Architecture model, Types of Clouds: Public Private & Hybrid Clouds, Resource management and scheduling, Clustering, Data Processing in Cloud: Introduction to Map Reduce for Simplified data processing on Large clusters.

**UNIT – IV**

Ad hoc networks, localization, MAC issues, Routing protocols, global state routing (GSR), Destination sequenced distance vector routing (DSDV), Dynamic source routing (DSR), Ad Hoc on demand distance vector routing (AODV), Temporary ordered routing algorithm (TORA), QoS in Ad Hoc Networks, applications.

**Text Books:**

2. J. Schiller, Mobile Communications, Addison Wesley

**Reference Books**

1. A. Mehrotra, GSM System Engineering.
### Purpose
To gain a broad understanding of the discipline of Web engineering and its application to the development and management of Web Applications.

### Course Outcomes

| CO1 | Learn the basic concepts of information and web architecture. |
| CO2 | Learn about the skills that will enable to design and build high level web enabled applications. |
| CO3 | Understand the applicability of Java Script as per current software industry standards. |
| CO4 | Acquaint the latest programming language for the implementation of object based and procedure based applications using Python. |

---

### Unit 1
**Information Architecture:** The role of Information Architect, Collaboration and communication, Organizing information, organizational challenges, Organizing web sites and Intranets, Creating cohesive organization systems, designing navigation systems, types of navigation systems, Integrated navigation elements, designing elegant navigation systems, Searching systems, Searching your web site, designing the search interface, Indexing the right stuff, To search or not to search grouping content, conceptual design, High level Architecture Blueprint. Architectural Page Mockups, Design Sketches.

### Unit 2
**Introduction to XHTML and HTML5:** Origins and Evolution of HTML and XHTML, Basic Syntax, Standard XHTML Document Structure, Basic Text Markup, Images, Hypertext Links, Lists, Tables, Forms, HTML5, Syntactic Differences between HTML and XHTML.

**Cascading Style Sheets:** Introduction, Levels of Style Sheets, Style Specification Formats, Selector Forms, Property Value Forms, Font Properties, List Properties, Color, Alignment of Text, Box Model, Background Images, Conflict Resolution.

### Unit 3
**Java Script:** Overview of JavaScript, Object Orientation and JavaScript, General Syntactic Characteristics, Primitives, Operations, and Expressions, Screen Output and Keyboard Input, Control Statements, Object Creation and Modification, Arrays, Functions, Constructors, Pattern Matching Using Regular Expressions, Errors in Scripts

### Unit 4
**Python:** Introduction to Python, Data Types and Expressions, Control Statements, Strings and Text Files, Lists and Dictionaries, Design with Functions, Design with Classes

### Text Books

### Reference Book
CSE-310N  Software Engineering

<table>
<thead>
<tr>
<th>Lecture</th>
<th>Tutorial</th>
<th>Practical</th>
<th>Major Test</th>
<th>Minor Test</th>
<th>Total</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>1</td>
<td>-</td>
<td>75</td>
<td>25</td>
<td>100</td>
<td>3</td>
</tr>
</tbody>
</table>

Purpose
To gain a broad understanding of the discipline of software engineering and its application to the development and management of software process.

Course Outcomes (CO)

<table>
<thead>
<tr>
<th>CO</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>To understand the basic concepts of Software Engineering.</td>
</tr>
<tr>
<td>CO2</td>
<td>To learn about the skills that will enable to construct high quality software.</td>
</tr>
<tr>
<td>CO3</td>
<td>To understand the software process models.</td>
</tr>
<tr>
<td>CO4</td>
<td>To understand the fundamental concept of requirements engineering and Analysis Modelling.</td>
</tr>
<tr>
<td>CO5</td>
<td>To understand the different design techniques and their implementation.</td>
</tr>
<tr>
<td>CO6</td>
<td>To learn about software testing and maintenance measures.</td>
</tr>
</tbody>
</table>

Unit-I


Unit-II


Unit-III


Unit-IV

Software Construction: Software construction fundamentals, minimizing complexity, Top-Down and Bottom-Up programming, structured programming, Compliance with Design and Coding Standards.
Maintenance: key issues, Types of software Maintenance, Cost of Maintenance, Software Re-Engineering.

Text Books:

Reference Books:
1. Pankaj Jalote, Software Engineering, Wiley India.
### Course Outcomes

| CO1 | Students will be able understand who the entrepreneurs are and what competences needed to become an Entrepreneur |
| CO2 | Students will be able understand insights into the management, opportunity search, identification of a Product; market feasibility studies; project finalization etc. required for small business enterprises. |
| CO3 | Students can be able to write a report and do oral presentation on the topics such as product identification, business idea, export marketing etc. |
| CO4 | Students be able to know the different financial and other assistance available for the establishing small industrial units. |

### Unit - I
Entrepreneurship: Concept and Definitions; Entrepreneurship and Economic Development; Classification and Types of Entrepreneurs; Entrepreneurial Competencies; Factor Affecting Entrepreneurial Growth – Economic, Non-Economic Factors; EDP Programmes; Entrepreneurial Training; Traits/Qualities of an Entrepreneurs; Entrepreneur; Manager Vs. Entrepreneur.

### Unit - II
Opportunity / Identification and Product Selection: Entrepreneurial Opportunity Search and Identification; Criteria to Select a Product; Conducting Feasibility Studies; Project Finalization; Sources of Information.

### Unit - III
Small Enterprises and Enterprise Launching Formalities : Definition of Small Scale; Rationale; Objective; Scope; Role of SSI in Economic Development of India; SSI; Registration; NOC from Pollution Board; Machinery and Equipment Selection; Project Report Preparation; Specimen of Project Report; Project Planning and Scheduling using Networking Techniques of PERT / CPM; Methods of Project Appraisal.

### Unit - IV
Role of Support Institutions and Management of Small Business : Director of Industries; DIC; SIDO; SIDBI; Small Industries Development Corporation (SIDC); SISI; NSIC; NISBUD; State Financial Corporation SIC; Marketing Management; Production Management; Finance Management; Human Resource Management; Export Marketing; Case Studies-At least one in whole course.

### Text Books:
### Course Outcomes (CO)

<table>
<thead>
<tr>
<th>CO</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO1</td>
<td>Design webpages using HTML, JavaScript and CSS.</td>
</tr>
<tr>
<td>CO2</td>
<td>Design and test simple function/program to implement Searching and sorting techniques using Python.</td>
</tr>
<tr>
<td>CO3</td>
<td>Develop program in Java Script for pattern matching using regular expressions and errors in scripts.</td>
</tr>
<tr>
<td>CO4</td>
<td>Design client-server based web applications.</td>
</tr>
</tbody>
</table>

### Purpose

To introduce the concepts of HTML5, JavaScript and Python.

### Lecture

<table>
<thead>
<tr>
<th>Lecture</th>
<th>Tutorial</th>
<th>Practical</th>
<th>Minor Test</th>
<th>Practical</th>
<th>Total</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>3</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>3 Hrs.</td>
</tr>
</tbody>
</table>

### Course Outcomes (CO)

1. Create your own page with your favorite hobbies using HTML, JavaScript and CSS.
2. Create a frameset in HTML that is divided into three sections. The frameset should have three zones.
   a. The Topmost section of the frameset should take up about just 15% of the browser window. Name this frame title.
   b. The middle section should be 75% of the browser window. Name this frame title.
   c. The lower section should be 10% of the browser window. Name this frame menu.
3. Create pages for each section. For the lowermost section, create page that loads the content into the middle section. The topmost section should contain a page describing the web page itself.
4. Create a web page, which displays the map of your country Link, each city/state on the image map, such that the respective HTML page of the city/state is displayed when the user selects an area.
5. Add the tickertape applet to your page by customizing it for the following settings:
   a. Increase the count by one.
   b. Accordingly update the message count.
   c. Change the text color to (237,192,171)
   d. Experiment with changing the scrolling speed.
   e. Customize the message text as per your page requirement.
6. Incorporate a quest book into the Diary Food Webpage and use Java Script to build validations into the form.
7. Use Cascading Style sheets (CSS) to modify the following:
   a. Change background.
   b. Change font type, face and color.
   c. Align Text.
   d. Remove underlines from hyperlinks.
8. Write the program for using JavaScript by using for – loops (through a block of code a number of times), for/in - loops (through the properties of an object), while - loops (through a block of code while a specified condition is true), do/while - loops (through a block of code while a specified condition is true).
9. Write a program in Java Script for the following:
   a. Copying, passing, and comparing by value
   b. Copying, passing, and comparing by reference
   c. References themselves are passed by value
10. Write program in Java Script for pattern matching using regular expressions and errors in scripts.
11. Write a Python function/program that accepts the lengths of three sides of a triangle as inputs. The program output should indicate whether or not the triangle is an equilateral triangle.
12. Write the Python functions for linear search, binary search, selection sort, Bubble Sort, Insertion Sort and converting Fibonacci to a linear algorithm.
13. Write program in Python using Lists and dictionaries, Control statements and Strings and text files.
### Essentials of Information Technology Lab

<table>
<thead>
<tr>
<th>Lecture</th>
<th>Tutorial</th>
<th>Practical</th>
<th>Minor Test</th>
<th>Practical</th>
<th>Total</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>3</td>
<td>40</td>
<td>60</td>
<td>100</td>
<td>3 Hrs.</td>
</tr>
</tbody>
</table>

**Purpose**

To introduce the concepts of Object Oriented Programming using Java and RDBMS

**Course Outcomes (CO)**

- **CO1**: Do Problem Solving using algorithms
- **CO2**: Design and test simple programs to implement Object Oriented concepts using Java
- **CO3**: Document artifacts using common quality standards
- **CO4**: Design simple data store using RDBMS concepts and implement

Students should implement at least 4-5 problems from the real world related to concern engineering branch for following both focus area during Practical hours:

1. Programs using Java Language
2. RDBMS Queries using MySQL

**Tools:**

- Understanding basic programming constructs using Scratch Tool - Flowcharts implementation through RAPTOR tool
- Eclipse IDE for Java programming
<table>
<thead>
<tr>
<th>CSE-316N</th>
<th>Software Engineering Lab</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Lecture</strong></td>
<td><strong>Tutorial</strong></td>
</tr>
<tr>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

**Purpose**: To gain a broad understanding of the discipline of software engineering implementation.

**Course Outcomes**

- **CO1**: To understand the basic concepts of Software Engineering.
- **CO2**: To learn about the reasons for the software crisis.
- **CO3**: To understand the software testing techniques.
- **CO4**: To understand the software metrics.
- **CO5**: To understand the different design techniques and their implementation.
- **CO6**: To learn about software testing and maintenance measures.

**List of Practical's**

1. To identify the role of the software in today's world across a few significant domains related to day to day life.
2. To identify the problem related to software crisis for a given scenario.
3. To classify the requirement into functional and non-functional requirements.
4. To implement at least four software metrics.
5. Preparation of requirement document for standard application problems in standard format (e.g. Library Management System, Railway Reservation system, Hospital management System, University Admission system).
6. To prepare Project Schedule for standard application problems in standard format.
7. To implement the functional testing techniques.
8. To implement the structural testing techniques.