Re-Accredited 'B++' 2.86 CGPA by NAAC
VEER NARMAD SOUTH GUJARAT UNIVERSITY
University Campus, Udhna-Magdalla Road, SURAT - 395 007, Gujarat, India.
વીર નર્મદ દક્ષિણ ગુજરાત યુનિવર્સિટી
યુનિવર્સિટી કેમ્પસ, ઉધના-મગદલ્લા રોડ, સુરત - ૩૯૫ ૦૦૭, ગુજરાત, ભારત.
Tel : +91 - 261 - 2227141 to 2227146, Toll Free : 1800 2333 011, Digital Helpline No.- 0261 2388888
E-mail : info@vnsgu.ac.in, Website : www.vnsgu.ac.in

ક્રમાંક : એસ/પરિપત્ર/સિલેબસ/૫૦૩૬/૨૦૨૪
તા.૦૭/૦૩/૨૦૨૪

પ્રતિ,
વડાશ્રી,
જે.પી.દાવર ઈન્સ્ટીટયૂટ ઓફ ઈન્ફોર્મેશન
સાયન્સ એન્ડ ટેકનોલોજી,
વીર નર્મદ દક્ષિણ ગુજરાત યુનિવર્સિટી,
સુરત.

વિષય :– એમ. .એસસી. (આઈ.ટી.) સેમે.–૭ અને ૮ ના અભ્યાસક્રમ બાબત.

મહાશય,

સવિનય જણાવવાનું કે, શૈક્ષણિક વર્ષ ૨૦૨૪–૨૫ થી અમલમાં આવનાર M.Sc. (IT) Sem. 7 & 8નો અભ્યાસ સમિતિ દ્વારા નિયુકત પેટાસમિતિ દ્વારા તૈયાર કરવામાં આવેલ અભ્યાસક્રમ સંદર્ભે ઈન્ફોર્મેશન ટેકનોલોજી વિષયની અભ્યાસ સમિતિની તા.૧૨/૦૨/૨૦૨૪ની સભાના ઠરાવ ક્રમાંક :૩ અન્વયે નીચે મુજબ કરેલ ભલામણ કોમ્પ્યુટર સાયન્સ એન્ડ ઈન્ફોર્મેશન ટેકનોલોજી વિદ્યાશાખાના અધ્યક્ષશ્રીએ વિદ્યાશાખાની મંજૂરીની અપેક્ષાએ વિદ્યાશાખાવતી મંજૂર કરી એકેડેમિક કાઉન્સિલને કરેલ ભલામણ એકેડેમિક કાઉન્સિલની તા.૦૧/૦૩/૨૦૨૪ની સભાનાં ઠરાવ ક્રમાંકઃ ૦૫ થી મંજુર કરેલ છે. જેનો અમલ કરવા આથી જાણ કરવામાં આવે છે.

ઈન્ફોર્મેશન ટેકનોલોજી વિષયની અભ્યાસ સમિતિની તા.૧૨/૦૨/૨૦૨૪ ની સભાના ઠરાવ ક્રમાંકઃ૩

:: આથી ઠરાવવામાં આવે છે કે, શૈક્ષણિક વર્ષ ૨૦૨૪–૨૫ થી અમલમાં આવનાર એમ.એસસી. (આઈ.ટી) સેમેસ્ટર– ૭ અને ૮ નો પેટાસમિતિ દ્વારા તૈયાર કરવામાં આવેલ અભ્યાસક્રમ સર્વાનુમતે મંજૂર કરી કોમ્પ્યુટર સાયન્સ એન્ડ ઈન્ફોર્મેશન ટેકનોલોજી વિદ્યાશાખાને ભલામણ કરવામાં આવે છે.

એકેડેમિક કાઉન્સિલની તા.૦૧/૦૩/૨૦૨૪ની સભાનાં ઠરાવ ક્રમાંકઃ ૦૫

:: આથી ઠરાવવવામાં આવે છે કે, શૈક્ષણિક વર્ષ ૨૦૨૪–૨૫ થી અમલમાં આવનાર M.Sc.(IT) Sem. 7 & 8 નો અભ્યાસ સમિતિ દ્વારા નિયુકત પેટાસમિતિ દ્વારા તૈયાર કરવામાં આવેલ અભ્યાસક્રમ સંદર્ભે ઈન્ફોર્મેશન ટેકનોલોજી વિષયની અભ્યાસ સમિતિની તા.૧૨/૦૨/૨૦૨૪ની સભાના ઠરાવ ક્રમાંક :૩ અન્વયે કરેલ ભલામણ કોમ્પ્યુટર સાયન્સ એન્ડ ઈન્ફોર્મેશન ટેકનોલોજી વિદ્યાશાખાના અધ્યક્ષશ્રીએ વિદ્યાશાખાની મંજૂરીની અપેક્ષાએ વિદ્યાશાખા વતી મંજૂર કરી એકેડેમિક કાઉન્સિલને કરેલ ભલામણ સ્વીકારી મંજૂર કરવામાં આવે છે.

બિડાણઃ ઉપર મુજબ

કુલસચિવ

પ્રતિ,
૧) અધ્યક્ષશ્રી, કોમ્પ્યુટર સાયન્સ એન્ડ ઈન્ફોર્મેશન ટેકનોલોજી વિદ્યાશાખા.
૨) પરીક્ષા નિયામકશ્રી, પરીક્ષા વિભાગ, વીર નર્મદ દ. ગુ. યુનિવર્સિટી, સુરત.

...તરફ જાણ તેમજ અમલ સારૂ.

M.Sc. (I.T.) 7th Semester

Course : 701 : Application Development using Full Stack

Course Code	701
Course Title	Application Development using Full Stack
Credit	4
Teaching per Week	4 Hrs
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Effective From	June 2024
Purpose of Course	This course is designed to augment students' programming skills with the latest technologies.
Course Objective	To provide understanding of the prototypal inheritance, Node.js, express, mongoDB, React.js and making students able to develop programs Full stack architecture.
Course Outcomes	CO1 : Students will be able to learn about MVC applications using NODE.JS, Express. CO2 : Students will be able to develop backend REST API using Express and learn API Security. CO3 : Students will be able to develop front end apps using React.js and to manage application version controlling using git.
Mapping between COs with PSOs	
Pre-requisite	Basic Javascript
Course Content	**Unit 1 : Introduction of Full Stack Architecture and Node.js Ecosystem** 1.1. Introduction to Full Stack Architecture 1.2. Components of Full Stack Architecture 1.3. Frontend, Backend 1.4. Architecture of Node.js Ecosystem 1.5. Familiarity with JavaScript 1.6. Installing Node.js 1.7. REPL **Unit 2 : Node.js** 2.1 Module and npm 2.1.1 npm 2.1.2 package.json 2.1.3 The node_modules 2.1.4 require() 2.2 Node concepts 2.1.1 The Event Loop 2.1.2 Asynchronous Coding 2.1.3 Callback Functions 2.1.4 Calling Conventions 2.1.5 Exception Handling 2.1.6 Callback Hell 2.1.7 Event Emitters 2.1.8 Extending EventEmitter 2.1.9 Listening for Events 2.1.10 Promise 2.1.11 async .. await 2.3 Core Modules 2.3.1 Command Line Arguments 2.3.2 Working with the File System 2.3.3 Global objects 2.3.4 File Systems and Streams 2.3.5 Utility Modules 2.3.6 http Module 2.3.7 Routes 2.3.8 Accessing Request Headers 2.3.9 Using TypeScript for Node.js programming

Mapping between COs with PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5
CO1					
CO2					
CO3					

	3. Express 3.1 Routing 3.2 HTTP Methods 3.3 URL Building 3.4 Middleware 3.5 Templating 3.6 Static Files 3.7 Form Data 3.8 Database 3.9 Cookies 3.10 Sessions 3.11 Authentication 3.12 Working withDatabase Engine like Mongo and Mongoose to insert, update and delete data 3.13 RESTful API 3.14 API Security **4. React.js** 4.1 React.js Introduction 4.1.1 React.js application architecture 4.1.2 Component 4.1.3 JSX Overview 4.2 Virtual DOM and Single Page Application 4.3 Components 4.3.1 Class Components 4.3.2 Functional Components 4.3.3 Nested Components 4.4.4 Conditional and Looping constructs 4.4.5 State 4.4.6 Props 4.5 Event Handling 4.5.1 Event Handling in Class Components 4.5.2 Event Handling in Functional Components 4.6 Component Life Cycle Methods 4.7 React Hooks 4.8 Forms 4.9 Router 4.10 State Management 4.11 Redux 4.12 Calling Backend API **5. Developer Tools** 5.1. Browser Tools 5.2. Version Control using Git and others Tools 5.3. Client side and server side Websocket Programming
Reference Book	1. Node.js 8 the Right Way: Practical, Server-Side JavaScript That Scales--Jim Wilson --Andy Hunt 2. Mastering Node.js - Second Edition: Build robust and scalable real-time server-side web application -- Sandro Pasquali --1st edition – Paperback 3. React Explained: Your Step-by-Step Guide to React, OS Training, LLC 4. Beginning React, Greg Lim 5. Learning React: Functional Web Development with React and Redux, Shroff/O'Reilly 6. Learn React Hooks: Build and refactor modern React.js applications using Hooks, Packt Publishing Limited
Teaching Methodology	Class Room Teaching, Discussion and Assignment

M.Sc.(I.T) 7th Semester

Course : 702 : Advanced .NET

<table>
<tr><td>Course Code</td><td>702</td></tr>
<tr><td>Course Title</td><td>Advanced .NET</td></tr>
<tr><td>Credit</td><td>4</td></tr>
<tr><td>Teaching per Week</td><td>4 Hrs</td></tr>
<tr><td>Minimum weeks per Semester</td><td>15 (Including Class work, examination, preparation, holidays etc.)</td></tr>
<tr><td>Last Review / Revision</td><td>June 2024</td></tr>
<tr><td>Purpose of Course</td><td>This course helps students to understand and use .NET advanced concepts with real world .NET applications.</td></tr>
<tr><td>Course Objective</td><td>To impart knowledge of Enterprise application development using .NET Framework.</td></tr>
<tr><td>Course Outcomes</td><td>CO1 : Students will be able to learn MVC Core.

CO2 : Students will be able to learn about Web API in .NET Core and use it in other apps.

CO3 : Students will be able to learn C# language features like tuples, expression bodied members, local/nested functions, delegates & events and LINQ.</td></tr>
<tr><td>Mapping between COs with PSOs</td><td><table><tr><td></td><td>PSO1</td><td>PSO2</td><td>PSO3</td><td>PSO4</td><td>PSO5</td></tr><tr><td>CO1</td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>CO2</td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>CO3</td><td></td><td></td><td></td><td></td><td></td></tr></table></td></tr>
<tr><td>Pre-requisite</td><td>Object Oriented Fundamental, Basic knowledge of C#, Basic knowledge of web development.</td></tr>
<tr><td>Course Content</td><td>Unit : 1: Introduction to ASP.NET Core MVC and C#.Net Core
1.1 Introduction to C#.NET Core
1.1.1 Overview of Object Oriented Programming
1.1.2 Classes and Interfaces
1.1.3 Pattern Matching
1.1.4 Tuples
1.1.5 Local / Nested Functions
1.1.6 Expression Bodied Members
1.1.7 Asynchronous Programming
1.2 Overview of ASP.NET Core MVC
1.2.1 MVC Architectural Pattern
1.2.2 Application Architecture
1.2.3 Application Configuration
1.2.4 URL Routing and Routing Configuration
1.2.5 Multiple Environments and Development Mode
1.2.6 NuGet Package and Package Manager
1.2.7 Dependency Injection
1.2.8 Unit Testing and ASP.NET MVC
1.2.9 Working with MVVM Architectural Pattern

Unit : 2 : Data Access and Modeling
2.1 Generics
2.1.1. Generic Class</td></tr>
</table>

2.1.2. Generic Members
2.1.3. Generic Interface
2.1.4. Generic Collections
2.2 Entity Framework
2.2.1. Code First Approach
2.2.2. Data First Approach
2.3 LINQ
2.3.1. Writing LINQ Queries
2.3.2. Writing LINQ expressions
2.3.3. LINQ with Objects
2.3.4. LINQ with EF Core
2.4 Anonymous Types
2.5 Working with SQL and No-SQL Data Storage Types

Unit : 3 : Controllers and User Interface

3.1 ASP.NET Core MVC Controller
3.1.1. Creating a Controller
3.1.2. Role of Controller
3.1.3. Action Methods
3.1.4. Action Result
3.1.5. Action Filters
3.1.6. Data Passing using TempData, ViewBag and ViewData
3.2 Configuring Middleware
3.3 Configuring Services
3.4 Synchronous and Asynchronous Programming
3.5 Consuming Web API and Minimal API
3.6 Working with Razor View
3.6.1. Razor View Engine
3.6.2. Creating Razor View
3.6.3. Binding with Models
3.6.4. Data Manipulation using Razor Pages
3.6.5. Partial View
3.6.6. Shared Layout
3.6.7. Application Navigation and Configuration
3.7 Introduction to Blazor
3.7.1. Project Template and Architecture
3.7.2. Application Configuration
3.7.3. Difference between Blazor and Razor
3.7.4. Creating User Interface using Blazor
3.7.5. Data Manipulation using Razor Pages
3.8 Introduction to MAUI

Unit : 4 : Web Service and Web API

4.1 Introduction to Web Service
4.2 Working with Web API
4.2.1. Creating ASP.NET Core Web API
4.2.2. Difference between Web Service and Web API
4.2.3. Testing Web API
4.3 Working with Minimal API

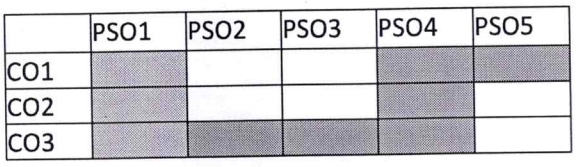
Unit : 5 : Hosting, Security and Connection Management

5.1 Hosting Web Applications
5.2 Dependency Injection
5.3 Security and Identity
5.4 Working with OpenID and OAuth Login

	5.5 Working with WebSockets and SignalR
Reference Book	1. C# 11 and .NET 7 by Mark J. Price, Packt Publishing Ltd., 2022 2. Learning Blazor by David Pine, O'Reilly, 2022 3. Microsoft Blazor by Peter Himschoot, Apress, 2022 4. .NET Core Microservices - The Complete Guide (.NET 6 MVC) by Bhrugen Patel, Packt Publishing Ltd., 2022
Teaching Methodology	Lectures, Discussion, Independent Study, Seminars and Assignment

M.Sc. (I.T.) 7th Semester

Course : 703 : Software Engineering

Course Code	703
Course Title	Software Engineering
Credit	4
Teaching per Week	4 Hrs
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Last Review / Revision	June 2024
Purpose of Course	To develop skills of software engineering in students
Course Objective	To provide insights about software engineering project planning, scheduling, SCM fundamentals and pattern based design. Students would be able to do plan, design, analyze risk and manage a software development process efficiently after learning this course.
Course Outcomes	CO1 : Students will be able to understand Software project management and scheduling of software tasks among software development team. CO2 : Students will be able to learn principles of Web development and Agile Software development Methodology. CO3 : Students will be able to learn principles of Software design pattern and Software Quality Assurance activities.
Mapping between COs with PSOs	
Pre-requisite	Basic concepts of software analysis and design
Course Content	**Unit 1 : Project Management** 1.1 Software Matrices 1.1.1 Project Management 1.1.2 Software Measurements 1.1.3 Metrics for Software Quality 1.1.4 Cost and Efforts Estimation Model 1.2. Project Scheduling 1.2.1 Relationship between People & Effort 1.2.2 Defining a Task set for the Software Project 1.2.3 Selecting & Refining Software Engineering Tasks 1.2.4 Scheduling and tracking techniques 1.2.5 Earned Value Analysis **Unit 2 : Software Risk and Change Management** 2.1. Risk Management 2.1.1 Software Risk 2.1.2 Risk Identification and Categories of Risk 2.1.3 Projection 2.1.4 Refinement 2.1.5 RMMM Plan 2.2. Change Management

	2.2.1 Software Configuration Management 2.2.2 SCM Repository 2.2.3 SCM Process 2.2.4 Version Control and Change Control **Unit 3 : Web Engineering and Agile Software Development** 3.1 Web Development Process 3.1.1 Attributes of web based application 3.1.2 Framework of Web engineering 3.1.3 Analyzing Web-Based system 3.1.4 Design of Web-Based Application 3.1.5 Testing of Web Application 3.1.6 Management Issues 3.2 Agile Methodology 3.2.1 Extreme Programming Adaptive Software Development 3.2.2 Dynamic systems Development Method 3.2.3 Scrum Method 3.2.4 Feature Driven Development 3.2.5 Agile Tools and Testing **Unit 4 : Software Design patterns** 4.1 Design Pattern Principles and Techniques 4.2 Software Architecture 4.3 Types of Design patterns 4.3.1 Creational pattern 4.3.2 Structural pattern 4.3.3 Behavioral pattern **Unit 5 : Advanced Software Engineering** 5.1 Software Quality Assurance 5.2 Quality Standards: ISO 2000, CMM, Six- Sigma 5.3 Service Oriented software engineering 5.4 Software engineering for Cloud Computing and AI 5.5 CASE tools and its effect 5.6 Component based Software systems 5.7 Mobile Development Process 5.8 Software Engineering issues in Embedded Systems 5.9 Emerging trends in Software Engineering
Reference Book	1. Software Engineering A practitioner's approach – Roger S Pressman - Seventh Edition- McGraw Hill 2. An Integrated Approach to Software Engineering – Pankaj Jalote – Narosa 3. Software quality assurance – from theory to implementation- Daniel Galin- Pearson education 4. Software Engineering- A programming approach- D. Bell, I. Morrey-PHI 5. Design Patterns: Elements of Reusable Object-Oriented Software, John Vlissides, Ralph Johnson, Richard Helm, Erich Gamma, , Addison-Wesley
Teaching Methodology	Class Room Teaching, Discussion and Assignment

M.Sc. (I.T.) 7th Semester

Course: 704 Data Analysis using Python

Course Code	704
Course Title	Data Analysis using Python
Credit	4
Teaching per Week	4 Hrs
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Last Review / Revision	June 2024
Purpose of Course	The Purpose of course is to help understanding the concepts of Python Programming and it's applications. The course also provides students with the skills necessary to develop Python programs and evaluate them.
Course Objective	The objective of the course is to impart knowledge of Python Programming, Python Libraries and Exploratory Data Analysis (EDA).
Course Outcomes	CO1 : Students will be able to understand the concepts of Python programming and it's applications. CO2 : Students will be able to demonstrate in-depth knowledge of Python Construct, Python libraries and Data Vizualization. CO3 : Students will be able to learn the use of Python libraries for Exploratory Data Analysis (EDA).
Mapping between COs with PSOs	

	PSO1	PSO2	PSO3	PSO4	PSO5
CO1					
CO2					
CO3					

Pre-requisite	Basic knowledge problem solving.
Course Content	**Unit 1 : Introduction to Python** 1.1. Features of Python 1.2. Python Environment Set-up 1.3. Working with Script files in Python 1.4. Python variables and Data Types 1.5. Python Operators - Comparison, Logical and Identity 1.6. Working with Python Strings and it's inbuilt functions 1.7. Python Conditional statements 1.8. Python User Define Functions and Lambda Expressions 1.9. Python Iterations 1.10. Matrix, Scalars and Vectors 1.11. Arrays in Python 1.12. Addition and Subtraction of matrices 1.13. Dot Product of matrices **Unit 2 : Python Sequences** 2.1 Working with Lists 2.1.1 Indexing and Slicing 2.1.2 Iterating and Merging 2.1.3 Built-in Functions 2.1.4 List Comprehension and Multidimensional Lists 2.2 Working with Tuples 2.2.1 Indexing and Slicing 2.2.2 Iterating and Merging 2.2.3 Operations on Tuples 2.2.4 Built-in Functions 2.3 Working with Dictionaries 2.3.1 Indexing and Slicing 2.3.2 Iterating and Merging

	2.3.3 Built-in Functions 2.3.4 Operations on Dictionaries and Nested Dictionaries **Unit 3 : Working with Python Libraries** 3.1 Working with Numpy 3.1.1 Numpy Arrays - ndarrays 3.1.2 Indexing, Slicing and Merging 3.1.3 Numpy Arithmetic Operations 3.1.4 Array shape manipulation and sorting 3.1.5 Saving and Loading Array 3.2 Working with Pandas 3.2.1 Pandas data structures - Series and Data Frames 3.2.2 Reading data from CSV and TXT file and databases 3.2.3 Indexing, Slicing and Merging 3.2.4 Data Frame Row/Column manipulation 3.2.5 Filtering(Query) operations 3.3 Introduction to Pytorch, Tensor Flow, Keras and Scipy **Unit 4 : Data Visualization and Exploratory Data Analysis** 4.1 Introduction to Plot 4.1.1 Line Appearance and Multiple Lines 4.1.2 Axis, Tricks and Grid 4.1.3 Labels, Annotation and Legends 4.2 Data Visualization with Matplotlib 4.2.1 Scatter plot 4.2.2 Line plot 4.2.3 Bar plot 4.2.4 Histogram 4.2.5 Box plot 4.2.6 Pie plot 4.3 Getting insights into Dataset 4.4 Handling Missing Values 4.5 Data Encoding and Data Normalization 4.6 Handling Correlation 4.7 Handling Outliers 4.8 Covariance and Correlation **Unit 5 : Files, OOps and Modules** 5.1 I/O operations on File 5.1.1 Create and read file 5.1.2 Modify and delete file 5.2 Object Oriented Programming in Python 5.2.1 Class and Object Creation 5.2.2 Python initializers 5.2.3 Inheritance and Polymorphism 5.3 Working with Modules 5.3.1 Creating a Module 5.3.2 Importing a Module 5.3.3 Python in-built Modules
Reference Books:	1. Python : The Complete Reference by Martin C. Brown, McGrawHill publication. 2. Python Programming by Ch Satyanarayana, M Radhika Mani and B N Jagadesh, Universities Press Publication 3. Python Programming by S. Sridhar, J.Indumathi and V. M. Hariharan, Pearson Publication. 4. Python Distilled by David M. Beazely, Pearson Publication. 5. Introduction to Python for Engineers and Scientists by Sandeep Nagar,Apress Publication. 6. Python : Visual Quick Start Guide by Toby Donaldson, Pearson Publication.
Teaching Methodology:	Lectures, Discussion, Independent Study, Hands-on-Session, Seminars and Assignment

M.Sc. (I.T.) 7th Semester

Course : 705 : Practical 15

<table>
<tr><td>Course Code</td><td>705</td></tr>
<tr><td>Course Title</td><td>Practical 15</td></tr>
<tr><td>Credit</td><td>3</td></tr>
<tr><td>Teaching Per Week</td><td>3 Hrs</td></tr>
<tr><td>Minimum Weeks Per Semester</td><td>15 (Including Practical Work, Examination, Preparation, Holidays etc.)</td></tr>
<tr><td>Review/Revision</td><td>June 2024</td></tr>
<tr><td>Purpose of Course</td><td>The course provides practical knowledge of web application development using full stack development frameworks like Node.js, React.js, Express, etc.</td></tr>
<tr><td>Course Objective</td><td>The course prepares students to develop applications using full stack development frameworks like Node.js, React.js, Express, etc.</td></tr>
<tr><td>Course Outcomes</td><td>CO1 : Students will be able to develop MVC based programs using NODE.JS, Express.

CO2 : Students will be able to develop backend REST API using Express and implement API Security.

CO3 : Students will be able to develop front end apps using React.js and implement version controlling using git practically.</td></tr>
<tr><td>Mapping between COs with PSOs</td><td><table><tr><th></th><th>PSO1</th><th>PSO2</th><th>PSO3</th><th>PSO4</th><th>PSO5</th></tr><tr><td>CO1</td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>CO2</td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>CO3</td><td></td><td></td><td></td><td></td><td></td></tr></table></td></tr>
<tr><td>Prerequisite</td><td>Object Oriented Programming Concepts</td></tr>
<tr><td>Course Content</td><td>Practical based on Paper No. 701 - Application Development using Full Stack.</td></tr>
<tr><td>Reference Books</td><td>NIL</td></tr>
<tr><td>Teaching Methodology</td><td>Lab Work</td></tr>
</table>

M.Sc. (I.T.) 7th Semester

Course : 706 : Practical 16

Course Code	706
Course Title	Practical 16
Credit	3
Teaching Per Week	3 Hrs
Minimum Weeks Per Semester	15 (Including Practical Work, Examination, Preparation, Holidays etc.)
Review/Revision	June 2024
Purpose of Course	The course provides practical knowledge of C#, LINQ, .NET Core, MVC and Python Programming.
Course Objective	The course prepares students to develop .NET based web applications and Python programs.
Course Out comes	CO1 : Students will be able to develop Web API in .NET framework and .NET Core and use it in other apps. CO2 : Students will be able to implement C# language features like tuples, expression bodied members, local/nested functions, delegates & events and LINQ practically. CO3 : Students will be able to develop Python programs with inheritance, polymorphism and other features.
Mapping between COs with PSOs	
Prerequisite	Object Oriented Programming Concepts
Course Content	Practical based on Paper No. 702- Advanced .NET and Paper 704 - Data Analysis using Python.
Reference Books	NIL
Teaching Methodology	Lab Work

	PSO1	PSO2	PSO3	PSO4	PSO5
CO1					
CO2					
CO3					

M.Sc. (I.T.) 7th Semester

Course: 707: Part Time Project 1

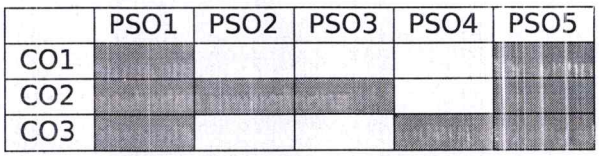
<table>
<tr><td>Course Code</td><td>707</td></tr>
<tr><td>Course Title</td><td>Part Time Project 1</td></tr>
<tr><td>Credit</td><td>3</td></tr>
<tr><td>Teaching per Week</td><td>3 Hrs</td></tr>
<tr><td>Minimum weeks per Semester</td><td>15 (Including Practical Work, Examination, Preparation, Holidays etc.)</td></tr>
<tr><td>Effective From</td><td>June 2024</td></tr>
<tr><td>Purpose of Course</td><td>The project work is introduced to make students implement their theory and practical knowledge they learned during this semester to solve real life problems for software applications.</td></tr>
<tr><td>Course Objective</td><td>To help students to develop software applications using React.js, Node.js and .NET.</td></tr>
<tr><td>Course Outcomes</td><td>CO1 : Students will be able to develop project using different technologies like React.js, Node.js and .NET.
CO2 : Students will be able to apply software engineering concepts to solve real world problems.
CO3 : Students will be able to apply database related concepts to design database for the project.</td></tr>
<tr><td>Mapping between COs with PSOs</td><td>
<table>
<tr><th></th><th>PSO1</th><th>PSO2</th><th>PSO3</th><th>PSO4</th><th>PSO5</th></tr>
<tr><td>CO1</td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO2</td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO3</td><td></td><td></td><td></td><td></td><td></td></tr>
</table>
</td></tr>
<tr><td>Pre-requisite</td><td>Knowledge of Object Oriented Programming, Web Technology Fundamentals, Software Engineering.</td></tr>
<tr><td>Course Content</td><td>The students must prepare documentation of the project completed as per the Software Engineering Guidelines.
At the end of the semester, the students have to submit their project report in bounded form to the institution.
The Project Presentation and Viva – Voce will be conducted as per the University exam schedule.

The students have to submit the following reports at the institution:
1. Project Joining Report
2. Project Title Report
3. Progress Report
4. Project Completion Certificate
5. Institution Certificate
6. Non-disclosure of Source Code Certificate (In case the student is unable to demonstrate project source code)

Note : If a student's performance is not satisfactory then as per the direction of the internal project guide / external examiner student may have to do coding in the lab according to the project work submitted during internal submission / external examination.</td></tr>
<tr><td>Reference Books</td><td>NIL</td></tr>
<tr><td>Teaching Methodology</td><td>Project guidance</td></tr>
</table>

M.Sc. (I.T.) 8th Semester

Course : 801 : Enterprise Java

Course Code	801
Course Title	Enterprise Java
Credit	4
Teaching per Week	4 Hrs
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Last Review / Revision	June 2024
Purpose of Course	To use Java in Web and Enterprise application development
Course Objective	To make students, understand and implement the Web based Enterprise Project Development Model of Java
Pre-requisite	Understanding of OOPS concept and its implementation by Java Language
Course Out come	CO1 : Students will be able to design Large scale Enterprise Application in Java design patterns CO2 : Students will be able use Java EE for writing java code for data driven enterprise applications CO3 : Students will able to use modern security statndards to secure Enterprise applications
Mapping between COs and PSOs	
Course Content	**Unit 1 : Java EE and SERVLETS** 1.1 Java EE Architecture 1.2 Introduction to Java Servlets 1.3 The Java Servlet API 1.4 Servlet Life Cycle 1.4 Request and Response 1.5 Working with Databases 1.6 Dispatching and forwarding the request 1.7 Working with HTTP Headers 1.8 Session Tracking 1.9 ServletConfig and ServletContext 1.10 Servlet Filters 1.11 Servlet Web Listeners 1.12 Introduction to Java Server Pages(JSP) 1.13 Lifecycle of JSP 1.14 JSP Scripting Elements 1.15 Implicit Objects 1.16 JSP Directive Elements 1.17 Action Elements 1.18 JSTL/EL

Unit 2: - Enterprise Java Beans

2.1 Introduction to EJB
2.2 Stateless Session Bean
2.3 Statefull Session Bean
2.4 Java Messaging Service Architecture
2.5 Message Driven Beans
2.6 Singleton Beans
2.7 Timers and Schedulers,
2.8 Asynchronous Beans

Unit 3 -JAVA PERSISTANCE and REST API

3.1 JPA architecture
3.2 ORM with Entities
3.3 Working with Relationships
3.4 Named Queries
3.5 Dynamic Queries AND Native Queries
3.6 REST services with JAX-RS
3.7 Using HTTP Methods in REST
3.8 JERSEY Client for REST Services

Unit -4 ENTERPRISE APPLICATION SECURITY

4.1 The Need of Security ,
4.2 Security Threats,Realm,
4.3 Users, Group and Roles,
4.4 Java EE Security Model
4.5 Credentials and Identity Stores
4.6 Authentication and Authorization Mechanisms
4.7 Data Integrity and Confidentiality
4.8 Securing Enterprise Applications
4.9 JWT based Authorization
4.10 OAuth and OpenIdConnect

Unit 5 : THE JAVA WEB APPLICATION FRAMEWORKS

5.1 Component Based Framework – JAVA SERVER FACES

5.1.1 Introduction to JSF
5.1.2 Request Processing Lifecycle
5.1.3 JSF Managed Beans
5.1.4 JSF UI Components
5.1.5 JSF Validators and Converters
5.1.6 Event Handling
5.1.7 Composite Components
5.1.8 Templating in JSF
5.1.9 Working with primefaces

5.2 Action Based Framework – SPRING

5.2.1 Introduction to Spring
5.2.2 Lifecycle of Spring MVC
5.2.3 DispatcherServlet
5.2.4 Multiple Controllers

	5.2.5 Working with databases 5.2.6 Spring Boot
Reference Book	1. JDBC 4.2, Servlet 3.1, and JSP 2.3 Includes JSF 2.2 and Design Patterns, Black Book, 2ed - Santosh Kumar, Dreamtech Press 2. Servlet & JSP: A Beginner's Tutorial - Budi Kurniawan, Brainy Software 3. The Definitive Guide to JSF in Java EE 8: Building Web Applications with JavaServer Faces - Bauke Scholtz, Arjan Tijms – Apress 4. Mastering Enterprise JavaBeans and the Java 2 Platform, Enterprise Edition, *by Ed Roman* 5. Beginning Java™ EE 7 Platform with Payara™ 5: From Novice to Professional by Antonio Goncalves 6. Mastering JavaServer Faces 2.2 - Anghel Leonard - Packt Publishing 7. Spring in Action 4ed - Craig Walls – Manning 8. Getting Started With Spring Framework: A Hands-on Guide to Begin Developing Applications Using Spring Framework - Ashish Sarin, J Sharma - Createspace Independent Pub 9. Spring 5 Design Patterns - Dinesh Rajput – Packt 10. Learning Spring Boot 2.0 - Greg L. Turnquist - Packt
Teaching Methodology	Lectures, Discussion, Independent Study, Seminars and Assignment

M.Sc. (I.T.) 8th Semester

Course: 802 Artificial Intelligence and Machine Learning

Course Code	802
Course Title	Artificial Intelligence and Machine Learning
Credit	4
Teaching per Week	4 Hrs
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Last Review / Revision	June 2024
Purpose of Course	The Purpose of course is to help understanding the concepts of Artificial Intelligence and it's applications. The course also provides students with the skills necessary to develop Supervised Machine learning models and evaluate them.
Course Objective	The objective of the course is to impart knowledge of Artificial Intelligence and Supervised Machine Learning applications.
Course Outcomes	CO1 : Students will be able to understand the concepts of Artificial Intelligence and it's applications. CO2 : Students will be able to demonstrate in-depth knowledge of supervised machine learning algorithms in the field of machine learning. CO3 : Students will be able to demonstrate in-depth knowledge of unsupervised machine learning algorithms in the field of machine learning.
Mapping between COs with PSOs	

	PSO1	PSO2	PSO3	PSO4	PSO5
CO1					
CO2					
CO3					

Pre-requisite	Basic Python Programming knowledge.
Course Content	**Unit 1 : Introduction to Artificial Intelligence** 1.1. Artificial Intelligence and it's Applications 1.2. Introduction to Knowledge Representation 1.3. Introduction to Expert System 1.4. Introduction to Reasoning 1.5. Introduction to Intelligent Agent 1.6. Introduction to NLP **Unit 2 : Introduction to Machine Learning** 2.1 Overview of ML 2.2 Applications of ML 2.3 ML types 2.4 ML Tools 2.5 ML Steps 2.6 Characterization of Machine Learning Problems **Unit 3 : Supervised Machine Learning** 3.1 Data set Preparation and Pre-processing 3.2 Classification Algorithms 3.2.1 Decision Tree 3.2.2 k-Nearest Neighbor - k-NN Concept and Intuition, Effect of k 3.2.3 Naïve Bayes and Bayes Classifier 3.3 Regression Algorithms 3.3.1 Simple and Multiple Regressions 3.3.2 Linear Regression 3.3.3 Polynomial Regression 3.3.4 Logistic Regression

	Unit 4 : Unsupervised Machine Learning 4.1 Types of Unsupervised ML 4.2 Introduction to Clustering 4.3 K-means clustering 4.4 Hierarchical clustering 4.5 Principle Component Analysis (PCA) **Unit 5 : Model Performance Evaluation** 5.1 Cross Validation 5.2 Accuracy and Confusion Matrix 5.3 Precision, Recall, F1-Score, 5.4 AUC-ROC Curve 5.5 RMSE, MSE, R-Square 5.6 Case Study
Reference Books:	1. Introduction to artificial intelligence, Akerkar. Raiend, PHI Leaning 2. A First Course in Artificial Intelligence, Deepak Khemani. MCGrawHilL Publication 3. Machine Learning by Tom M. Mitchell, MCGraw Hill 4. Machine Leaming using Python by DineshKumar Manaranjan Pradhan, Wiley India Publication 5. Real-World Machine Leaning by Henrik Brink, Joseph Richards, Mark Fetherolf, Manning Publications. 6. Introduction to Machine Leaning with Python: A Guide for Data Scientists by Andreas C. Muller, Sarah Guido, O'Reilly Publication
Teaching Methodology:	Lectures, Discussion, Independent Study, Hands-on-Session, Seminars and Assignment

P. M Desai

M.Sc. (I.T.) 8th Semester

Course : 803 Smart Device Computing Using iOS

Course Code	803
Course Title	Smart Device Computing Using iOS
Credit	4
Teaching per Week	4 Hrs
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Last Review / Revision	June 2024
Purpose of Course	The Purpose of course is to help understanding the components and structure of mobile application development using iOS. The course also provides students with the skills necessary to develop an iOS App from scratch to deploying it on the Apple Store.
Course Objective	The objective of the course is to impart knowledge of Swift and Apple iOS application Design and Development.
Course Outcomes	CO1 : Students will be able to understand Apple based smart device application development CO2 : Students will be able to learn about various components of iOS application development tools CO3 : Students will be able to publish iOS application on Apple store.
Mapping between COs with PSOs	

	PSO1	PSO2	PSO3	PSO4	PSO5
CO1					
CO2					
CO3					

Pre-requisite	Object Oriented Programming knowledge.
Course Content	**Unit 1 : Introduction to iOS with Swift Language** 1.1. Introduction iOS and iOS Architecture 1.1.1. Foundation Framework 1.1.2. Cocoa Framework 1.2. Introduction to Xcode IDE 1.2.1. Setting up Development Environment 1.2.2. Xcode Development Tools – Interface Builder and Simulator 1.2.3. Testing and Debugging 1.3. Introduction to Swift 1.3.1. Data types, Variables in Swift 1.3.2. Tuples, Constants, Literals in Swift 1.3.3. Working with Strings in Swift 1.4. Optionals in Swift - Implicit and Explicit 1.5. Control Flows and Functions in Swift 1.6. Collections in Swift : Dictionaries, Arrays, and Sets 1.7. Functions and Closures 1.8. Object Oriented Programming in Swift 1.8.1. Custom Class and Instance Creation 1.8.2. Initializers in swift 1.8.3. Inheritance and Polymorphism 1.9. Protocols and Extensions 1.10. Information Property List File and App Permissions **Unit 2 : iOS Design Patterns** 2.1 Introduction to Storyboard 2.2 Introduction to UIView, UIWindow and UIViewController 2.3 Model View Controller (MVC) Pattern in Interface Design 2.4 Application Life Cycle and View Controller Life Cycle 2.5 Storyboard and Interface builder 2.6 Working with Basic UIElements 2.6.1 UILabel, UIButton, UITextFeild, UIImageView etc. 2.7 IBActions and IBOutlets

	2.8 Auto Layout Constraints to create Adaptive UI 2.9 UIAnimation 2.9.1 Animation using Auto Layout Constraints 2.9.2 Animation with UIImageView 2.10 Recognizing and Handling Gestures 2.10.1 Working with different types of Gestures 2.10.2 Gestures with UIElements **Unit 3 : UIControls in iOS** 3.1 Navigation Controller and its Usage 3.2 Navigation Techniques 3.2.1 Segue, Push, Pop, Present and Dismiss 3.3 Working with TableView 3.3.1 Static TableViewController 3.3.2 Dynamic TableView 3.4 Working with PickerView 3.5 Working with Miscellaneous Controls in iOS 3.5.1 UICollectionView and UITabBarController 3.5.2 UIScrollView, UIWebView and ContainerView 3.6 Working with AlertController and its Types **Unit 4 : Data Persistence and Data Manipulation Techniques** 4.1 Working with UserDefaults for data persistence 4.2 Introduction to FileManager 4.3 Frameworks and Library Configurations 4.4 Data Persistence Techniques 4.4.1 SQlite Framework 4.4.2 Core Data Framework 4.5 Data Manipulation Techniques 4.5.1 JSON Parsing 4.5.2 XML Parsing **Unit 5 : Advance Programming in iOS** 5.1 API integration 5.2 Location based Services 5.1.1. Core Location Services 5.1.2. CLLocation and CLLocationManager Classes 5.1.3. MapKit, MapView and MKPointAnnotation 5.1.4. Location Based Call-outs 5.3 Publishing iOS App to Apple Store 5.4 Introduction to SwiftUI
Reference Books:	1. Swift Programming: The Big Nerd Ranch Guide (2nd Edition) (Big Nerd Ranch Guides) 2nd Edition by Matthew Mathias (Author), John Gallagher (Author) 2. Swift: A Comprehensive Intermediate Guide to Learn and Master the Concept of Swift Programming Kindle Edition by MG Martin (Author) 3. iOS 12 Programming Fundamentals with Swift: Swift, Xcode, and Cocoa Basics 1st Edition by Matt Neuburg (Author) 4. iOS Programming: The Big Nerd Ranch Guide, by Christian Keur and Aaron Hillegass 5. Beginning Swift by Rob kerr and Kare Morstol, Packt Publication 6. iOS 16 Programming for Beginners by Ahmad Sahar and Craig Clayton, Packt Publication.
Teaching Methodology:	Lectures, Discussion, Independent Study, Hands-on-Session, Seminars and Assignment

M.Sc. (I.T.) 8th Semester

Course : 804 : Blockchain Computing

Course Code	804
Course Title	Blockchain Computing
Credit	4
Teaching per Week	4 Hrs
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Last Review / Revision	June 2024
Purpose of Course	To learn the concepts of Blockchain, cryptocurrencies , NFTs and smart contracts.
Course Objective	To introduce the comprehensive concepts of Blockchain and it's transactions. At the end of this course, a student will be able to comprehend the fundamental concepts required for usage and development of Blockchain.
Course Outcomes	CO1 : Students will be able to learn technical foundations of Blockchain technology CO2 : Students will be able to understand bitcoin, ethereum cryptocurrency and the transactions CO3 : Students will be able to understand the concepts of Smart Contracts, decentralization,security mechanisms used in blockchain and NFTs
Mapping between COs with PSOs	
Pre-requisite	Security algorithms like public key,private key,Hash and Java

	PSO1	PSO2	PSO3	PSO4	PSO5
CO1					
CO2					
CO3					

Course Content	**Unit 1. Introduction to Blockchain** 1.1 Understanding Blockchain 1.2 The growth of Blockchain Technology 1.3 Distributed systems 1.4 History of Blockchain 1.5 Common Misconceptions **Unit 2. Cryptrography** 2.1 Introduction to crptography 2.2 Types of cryptrography 2.2.1 Symmetric cryptography 2.2.2 Asymetric cryptography 2.2.3 Hash algorithms 2.2.4 Digital Signatures 2.3 Merkle Tree **Unit 3. Building Blockchain** 3.1 Essentials of Blockchain

	3.2 Blockchain architecture 3.3 Generic elements of a Blockchain 3.4 Types of Blockchain 3.5 Consensus 3.5.1 Byzantine Fault 3.5.2 Proof of Work 3.5.3 Proof of Stake 3.5.4 Double-spending 3.6 Creating blocks and links 3.7 Inserting Hashes 3.8 Forking in block chain **Unit 4. Smart contract based Blockchains – A case of Ethereum Blockchain** 4.1 Overview of Ethereum 4.2 Ethereum network 4.3 Ethereum structure 4.4 Proof of Stake 4.5 Smart contracts 4.6 Ether and gas points 4.7 Ethereum operations 4.8 Ethereum wallets 4.9 Mining Ether 4.10 Decentralized Autonomous Organization (DAO) and Decentralized Finance 4.11 Creating Smart Contracts using Solidity **Unit 5. Hyperledger and NFT (Non Fungible Tokens)** 5.1 Introduction to Web 3.0 5.1.1 Development Frameworks 5.1.2 Decentralize Applications (DApps) 5.2 Hyperledger as a Protocol 5.2.1 Reference Architecture 5.2.2 Hyperledger Fabric 5.2.3 Distributed Ledger 5.3 NFT (Non Fungible Tokens) 5.3.1 Introduction to NFT 5.3.2 Difference between NFT and cryptocurrency 5.3.3 Types of NFT 5.3.4 Creating,buying and selling NFT 5.3.5 Impact of NFT on environment 5.3.6 NFT Usage and Rights 5.3.7 Innovative and popular NFT 5.4 Use Cases of Blockchain 5.4.1 Financial technology 5.4.2 Real Estate 5.4.3 Insurance 5.4.4 Governance

	5.4.5 Other industries 5.4.6 Governance
	5.5 Blockchain Security 5.5.1 Blockchain layers and attacks 5.5.2 Threat modelling 5.5.3 Regulation and compliance
Reference Books	1. Mastering Blockchain, fourth edition,, Imran Bashir, Packt Publisher 2. Blockchain with Hyperledger Fabric, Second Edition, Niton G., Anthony O'Dowd, Petr Novotny, Luc Desrosiers, Venkataraman Ramakrishna ,Salman A.Baset, ,Packt Publisher 3. Mastering Blockchain,Imran Bashir, Packt Publisher 4. Blockchain for dummies, Tiana laurence, Wiley 5. Blockchain Consensus: An Introduction to Classical, Blockchain, and Quantum Consensus Protocols, Imran Bashir, Apress 6. Bitcoin and Cryptocurrency Technologies: Arvind narayaran, Joseph Bonneau, Edward Felten, Andrew Miller, Steven Goldfeder, Priceton university press 7. Blockchain Applications: A Hands-On Approach , Arshdeep Bahga ,Vijay Madisetti - VPT 8. Introducing Blockchain Applications: Understand and Develop Blockchain Applications Through Distributed Systems, Joseph Thachil George, Apress 9. Mastering Ethereum Building Smart Contracts and DApps, Andreas Antonopoulos, Gavin Wood, O'Reilly 10. Blockchain and Ethereum Smart Contract Solution Development: Dapp Programming with Solidity, Weijia Zhang,Tej Anand, Apress 11. Mastering BitCoin 2/ED programming the open blockchain, Andreas M. Antonopoulos, O'Reilly 12. Introducing Blockchain with Java: Program, Implement, and Extend Blockchains with Java, Spiro Buzharovski, Apress 13. Getting Started with Ethereum: A Step-by-Step Guide to Becoming a Blockchain Developer, Davi Pedro Bauer, Apress 14. Blockchain: The Blockchain For Beginners Guide To Blockchain Technology And Leveraging Blockchain Programming, Josh Thompsons 15. The NFT Handbook: How to Create, Sell and Buy Non-Fungible Tokens, QuHarrison Terry, Matt Fortnow, Wiley
Teaching Methodology	Lectures, Discussion, Self Study, Seminars, Case Study and Assignment

M.Sc. (I.T.) 8th Semester

Course : 805 : Practical 17

Course Code	805
Course Title	Practical 17
Credit	3
Teaching Per Week	3 Hrs
Minimum Weeks Per Semester	15 (Including Practical Work, Examination, Preparation, Holidays etc.)
Review/Revision	June 2024
Purpose of Course	The course provides practical knowledge of web application development using JAVA technologies like JSP, Servlests, JSF, EJB, JMS, JPA etc and Supervised Machine learning.
Course Objective	The course prepares students to develop web applications using JAVA based frameworks and Supervised Machine learning models.
Course Out comes	CO1 : Students will be able to develop web applications using JAVA based frameworks and design patterns. CO2 : Students will be able to implement java security and enterprise java security with authentication and authorization. CO3 : Students will be able to implement web services, REST services and REST patterns in their web applications.
Mapping between COs with PSOs	
Prerequisite	Object Oriented Programming Concepts and Core JAVA.
Course Content	Practical based on Paper No. 801 – Enterprise JAVA.
Reference Books	NIL
Teaching Methodology	Lab Work

	PSO1	PSO2	PSO3	PSO4	PSO5
CO1					
CO2					
CO3					

M.Sc. (I.T.) 8th Semester

Course : 806 : Practical 18

Course Code	806
Course Title	Practical 18
Credit	3
Teaching Per Week	3 Hrs
Minimum Weeks Per Semester	15 (Including Practical Work, Examination, Preparation, Holidays etc.)
Review/Revision	June 2024
Purpose of Course	The course provides practical knowledge of application development for smart devices using iOS .
Course Objective	The course prepares students to develop applications for smart devices using iOS.
Course Out comes	CO1 : Students will be able to write programs using Swift programming concepts like Optional, Closure, Protocol, Extensions etc. CO2 : Students will be able to develop mobile applications Apple Compatible devices using Swift GUI components and Core Data. CO3 : Students will be able to develop Supervised Machine learning models for Classification and Regression problems .
Mapping between COs with PSOs	
Prerequisite	Basic Object Oriented Programming Concepts and basics of Python.
Course Content	Practical based on Paper No. 802 - Artificial Intelligence and Machine Learning and Paper No. 803 – Smart Device Computing Using iOS.
Reference Books	NIL
Teaching Methodology	Lab Work

	PSO1	PSO2	PSO3	PSO4	PSO5
CO1					
CO2					
CO3					

M.Sc. (I.T.) 8th Semester

Course: 807: Part Time Project 2

<table>
<tr><td>Course Code</td><td>807</td></tr>
<tr><td>Course Title</td><td>Part Time Project 2</td></tr>
<tr><td>Credit</td><td>3</td></tr>
<tr><td>Teaching per Week</td><td>3 Hrs</td></tr>
<tr><td>Minimum weeks per Semester</td><td>15 (Including Practical Work, Examination, Preparation, Holidays etc.)</td></tr>
<tr><td>Effective From</td><td>June 2024</td></tr>
<tr><td>Purpose of Course</td><td>The project work is introduced to make students implement their theory and practical knowledge they learned during this semester to solve real life problems for software applications.</td></tr>
<tr><td>Course Objective</td><td>To help students to develop software applications using Java Enterprise Edition.</td></tr>
<tr><td>Course Outcomes</td><td>CO1 : Students will be able to develop multi layered and MVC based Java applications.
CO2 : Students will be able to apply Software engineering concepts to solve real world problems.
CO3 : Students will be able to apply database related concepts to design database for the project.</td></tr>
<tr><td>Mapping between COs with PSOs</td><td>
<table>
<tr><td></td><td>PSO1</td><td>PSO2</td><td>PSO3</td><td>PSO4</td><td>PSO5</td></tr>
<tr><td>CO1</td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO2</td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO3</td><td></td><td></td><td></td><td></td><td></td></tr>
</table>
</td></tr>
<tr><td>Pre-requisite</td><td>Knowledge of Object Oriented Programming, Web Technology Fundamentals, Software Engineering.</td></tr>
<tr><td>Course Content</td><td>The students must prepare documentation of the project completed as per the Software Engineering Guidelines.
At the end of the semester, the students have to submit their project report in bounded form to the institution.
The Project Presentation and Viva – Voce will be conducted as per the University exam schedule.

The students have to submit the following reports at the institution:
1. Project Joining Report
2. Project Title Report
3. Progress Report
4. Project Completion Certificate
5. Institution Certificate
6. Non-disclosure of Source Code Certificate (In case the student is unable to demonstrate project source code)

Note : If a student's performance is not satisfactory then as per the direction of the internal project guide / external examiner student may have to do coding in the lab according to the project work submitted during internal submission / external examination.</td></tr>
<tr><td>Reference Books</td><td>NIL</td></tr>
<tr><td>Teaching Methodology</td><td>Project guidance</td></tr>
</table>

Master of Science (Information and Communication Technology)

Name of Program	**Master of Science (Information and Communication Technology)**
Abbreviation	**M.Sc. (I.C.T.)**
Duration	**2 Years**
Eligibility Criteria	Graduate in the discipline of computer application / computer science / computer engineering / Information Science / Information Technology
Objective of Program	To prepare human resources for cutting edge technologies in the field of ICT.
Program Outcome	**PO1 : Fundamental Knowledge Enrichment** Program trains students with the core computer science and Information Technology (IT) knowledge domains. It also makes students capable of using core concepts in the conceptualization of domain specific application development. **PO2 : Critical Thinking Development** The program develops the skills of critical thinking, problem solving, evaluative learning of various techniques, and understanding the essence of the problem. **PO3 : Advanced Emerging Technology Awareness** The program trains students with the latest technologies that is being used in the industry. The continuous syllabi review adds value to the program for the outgoing students and make them ready to face challenging demands of the industry. **PO4 : Advanced Tools Usage** The program teaches the students to apply the advanced tools to solve real world problems. **PO5 : Nurturing Project Planning and Management Capabilities** The program trains students for designing and conceptualizing the software architecture, planning and managing the product development process of complex and live software projects. It also makes students understand the decision making for selection of an appropriate project management capabilities. **PO6 : Real World Problem / Project Development** Real world projects provide the candidates exposure to work in the challenging and demanding environment of the industry. The project development training makes students employable and industry ready. **PO7 : Team Work and Leadership Development** Trains students to work in a team and also to take leadership of the of the project management team.
Program Specific Outcomes	PSO1 : Students will learn various aspects of Digital Communication Technologies. PSO2 : Students will be able to utilize knowledge of communication technologies in I.C.T. based applications. PSO3 : Students will be able to solve complex programming problems. PSO4 : Students will be able to learn emerging technologies and

	apply them for the development of Web applications, Mobile applications, IOT applications, etc.... PSO5: Students will develop necessary Entrepreneur and Technical skills to start their own business in I.C.T domain.
Mapping between POs and PSOs	

	PSO1	PSO2	PSO3	PSO4	PSO5
PO1					
PO2					
PO3					
PO4					
PO5					
PO6					
PO7					

Medium of Instruction	English
Program Structure	Semester 3

Course Code	Title	Teaching per week		Course Credits	University Examination		Internal Marks	Total Marks
		Theory	Practical		Duration	Marks		
ICT 301	Introduction to Python and Data Science	4	0	4	3 Hrs	70	30	100
ICT 302	Data Communication and Internet of Things	4	0	4	3 Hrs	70	30	100
ICT 303	Cloud Computing	4	0	4	3 Hrs	70	30	100
ICT 304	Open Source Web Development	4	0	4	3 Hrs	70	30	100
ICT 305	Practical 5	-	3	3	2 Hrs	70	30	100
ICT 306	Practical 6	-	3	3	2 Hrs	70	30	100
ICT 307	Part Time Project 3	-	3	3	-	70	30	100
	Total	**16**	**9**	**25**	**-**	**490**	**210**	**700**

Program Structure	Semester 4

Course Code	Title	Teaching per week		Course Credits	University Examination		Internal Marks	Total Marks
		Theory	Practical		Duration	Marks		
ICT 401	Project	-	-	25	-	490	210	700
	Total	**-**	**-**	**25**	**-**	**490**	**210**	**700**