Master of Science (Information Technology)

Name of Program	Master of Science (Information Technology)
Abbreviation	M.Sc. (I.T.)
Duration	5 Years Integrated Course B.Sc.(I. T.) – 3 years – Semester 1 to 6 M.Sc.(I. T.) – 2 years – Semester 7 to 10
Eligibility	H S C / Equivalent Examination from Science Stream (A / B / AB Group) or Vocational Stream or General Stream (Commerce) with English as one of the subject.
Objective of Program	The objective of the program is to transform students into professionals by indoctrinating advanced technical knowledge, enhancing technical skills, communication skills and provide outstanding placement in reputed I.T. companies.
Program Outcome	After the completion of the course, students will be able to develop and manage various types of software based on technologies learnt throughout the course and emerging technologies in IT industry which will give them excellent career prospects.
Effective From	June 2019

Program Structure — **M.Sc. (I.T.) – Semester 7 (M.Sc. (I.T.) 5 years Integrated Course)**

Course Code	Title	Teaching per week (Hrs.)		Course Credits	University Examination		Internal Marks	Total Marks
		Theory	Practical		Duration	Marks		
701	Application Development using Full Stack	4	0	4	3 Hrs	70	30	100
702	Application Development using .NET CORE	4	0	4	3 Hrs	70	30	100
703	Software Engineering	4	0	4	3 Hrs	70	30	100
704	Optimization Techniques	4	0	4	3 Hrs	70	30	100
705	Practical 15	-	3	3	2 Hrs	70	30	100
706	Practical 16	-	3	3	2 Hrs	70	30	100
707	Part Time Project 1	-	3	3	-	70	30	100
	Total	16	9	25	-	490	210	700

Program Structure — **M.Sc. (I.T.) – Semester 8 (M.Sc. (I.T.) 5 years Integrated Course)**

Course Code	Title	Teaching per week (Hrs.)		Course Credits	University Examination		Internal Marks	Total Marks
		Theory	Practical		Duration	Marks		
801	Java Web Development	4	0	4	3 Hrs	70	30	100
802	Enterprise Java	4	0	4	3 Hrs	70	30	100
803	Elective : Elective 1 Smart Device Computing using iOS Elective 2 Smart Device Computing using Android	4	0	4	3 Hrs	70	30	100
804	Artificial Intelligence	4	0	4	3 Hrs	70	30	100
805	Practical 17	-	3	3	2 Hrs	70	30	100
806	Practical 18	-	3	3	2 Hrs	70	30	100
807	Part Time Project 2	-	3	3	-	70	30	100
	Total	16	9	25	-	490	210	700

Program Passing Rules	As per University rules

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.)
Last Review / Revision	June 2019
Purpose of Course	This course is designed to augment students programming skills with latest technologies.
Course Objective	To provide understanding of the prototypal inheritance, Node, express, mongoDB, Angular and making students able to develop programs Full stack programs.
Pre-requisite	Basic Javascript
Course Out come	This would help students to understand the paradigm change in programming and help them in developing applications using express,node with document database MongoDB along with angular.
Course Content	**Unit 1 : Introduction of Node.js Ecosystem** 1.1. Architecture of Node.js Ecosystem 1.2. Familiarity with JavaScript 1.3. The Problem with I/O 1.4. Prototypal Inheritance 1.5. UI-UX, Responsive design, Security 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(), createServer() 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 2.1.10Exception Handling 2.1.11File Systems 2.1.12Node.js - RESTful API 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 Web Module 2.3.7 Routes 2.3.8 Accessing Request Headers 2.3.9 Working with Database Engine like Mongo and Mongoose to insert, update and delete data **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 RESTful APIs 3.13 Error handling **4. AngularJS** 4.1. Single-page Application Framework 4.2. Angular CLI 4.3. Model-View-Controller Architecture 4.4. Two Way Data Binding 4.5. Directives, Pipes, Components, Scope Inheritance, Method Chaining, Templates, Services, Forms and Validation 4.6. Animation and Routing 4.7. Calling API, Using Third Party API 4.8. Web-Sockets, Use of UI Frameworks Plug-ins **5. Developer tools** 5.1. Browser Tools 5.2. Version Control using Git and others Tools
Reference Book	1. Node.js, MongoDB and Angular Web Development: The definitive guide to using the MEAN stack to build web applications – Brad Dayley and Brendan Dayley-Second Edition- Kindle Edition 2. MEAN Cookbook: The meanest set of MEAN stack solutions around - Nicholas McClay-1st edition-Kindle edition 3. Node.js for PHP developers - Daniel Howard - First edition - O'Reilly 4. Full Stack JavaScript Development With MEAN: MongoDB, Express, AngularJS, and Node.JS - Colin J Ihrig and Adam Bretz-first edition- Kindle edition 5. Node.js 8 the Right Way: Practical, Server-Side JavaScript That Scales--Jim Wilson --Andy Hunt 6. Mastering Node.js - Second Edition: Build robust and scalable real-time server-side web application -- Sandro Pasquali --1st edition -- Paperback
Teaching Methodology	Class Room Teaching, Discussion and Assignment
Evaluation Method	30% Internal assessment 70% External assessment

P. M. Dasai

Course : IT 702 : Application Development using .NET CORE

Course Code	702
Course Title	Application Development using .NET CORE
Credit	4
Teaching per Week	4 Hrs
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Last Review / Revision	June 2019
Purpose of Course	This course helps students to understand and use .NET advanced concepts with real world .NET applications.
Course Objective	To impart knowledge of Enterprise application development using .NET Framework.
Pre-requisite	Object Oriented Fundamental, ADO.NET, ASP.NET, Basic knowledge of C#
Course Out come	Students will be able to develop enterprise applications using .NET advanced concepts.
Course Content	**Unit : 1: ASP.NET using C#.Net** 1.1 Overview of C#.NET Language 1.2 Overview of ASP.NET Features 1.3 Generics 1.4 Serialization 1.5 Membership Functionality 1.6 Globalization and Localization 1.7 Working with AJAX 1.7.1 Client Side and Server Side AJAX 1.7.2 AJAX Toolkit 1.8 Web Services 1.8.1 Introduction to Web Services 1.8.2 RESTful API 1.8.3 Working with .NET Application 1.2 Working with RESTful Services **Unit : 2 : ASP.NET MVC** 2.1 MVC Architectural Pattern 2.2 URL Routing Engine 2.3 Routing Configuration 2.4 Wiring Controller, Model, and View 2.5 Data Access and Mo leling 2.6 TempData, ViewBag and ViewData 2.7 Working with RESTful Services 2.8 Unit Testing and ASP.NET MVC 2.9 Razor View Engine **Unit : 3 : Language Integrated Queries - LINQ** 3.1 LINQ Language Features 3.2 Object Initialization 3.3 Anonymous Types 3.4 Implicitly Typed Local Variables 3.5 Lambda Expression 3.6 Query Expression 3.7 LINQ to Objects 3.8 LINQ to SQL 3.9 LINQ to Entities **Unit : 4 : Programming using C#.Net CORE** 4.1 Overview of C#.Net CORE 4.2 .NET CORE Assemblies and Libraries 4.3 Pattern Matching 4.4 Tuples and Deconstruction

P. M. Dhan

	4.5 Local/Nested Functions 4.6 Expression Bodied Members 4.7 Working with Delegates and Events 4.8 Async return types 4.9 NuGet Package **Unit : 5 : Application Designing using ASP.NET Core** 5.1 Introduction to ASP.NET Core 5.2 Working with OpenID and OAuth Login 5.3 Asynchronous Programming 5.4 Multiple Environments and Development Mode 5.5 Working with WebSockets and SignalR 5.6 Self hosting of Web Applications 5.7 Dependency Injection 5.8 Action Filters 5.9 Security and Identity 5.10 Working with SQL and No-SQL Data Storage Types
Reference Book	1. Professional C# 7 and .NET Core 2.0 by Christian Nagel, Wrox / Wiley, 2018 2. C# 7 and .NET Core Cookbook by Dirk Strauss, O'Reilly / Packt Publishing Limited, 2017 3. C# 7.1 and .NET Core 2.0 - Modern Cross-Platform Development - Third Edition by Mark J. Price, Packt Publishing Limited, 2017 4. C# 7 and .NET Core 2.0 Blueprints by Dirk Strauss and Jas Rademeyer, Packt Publishing Ltd, 2018 5. C# 7 and .NET Core: Modern Cross-Platform Development by Mark J. Price, Packt Publishing Ltd, 2017 6. Learning ASP.NET Core 2.0 by Jason De Oliveira and Michel Bruchet, Packt Publishing Ltd, 2017 7. Mastering ASP.NET Core 2.0 by Ricardo Peres, Packt Publishing Limited, 2017 8. Professional ASP.NET MVC 5 by Jon Galloway, Wrox, 2014 9. Beginning ASP.NET 4.5: in C# and VB by Imar Spaanjaars, Wiley, 2014 10. ASP.NET Core 2 Fundamentals by Onur Gumus and Mugilan T. S. Ragupathi, Packt Publishing Ltd, 2018 11. Learning ASP.NET Core MVC Programming by Mugilan T. S. Ragupathi, Packt Publishing Ltd, 2016 12. ASP.NET Core Essentials by Shahed Chowdhuri, Packt Publishing Ltd, 2016 13. Enterprise Application Architecture with .NET Core by Ganesan Senthilvel, Ovais Mehboob Ahmed Khan, Habib Ahmed Qureshi, Packt Publishing Ltd, 2017 14. ASP.NET Core 2 and Angular 5 by Valerio De Sanctis, Packt Publishing Ltd, 2017 15. ASP.NET MVC with Entity Framework and CSS by Lee Naylor, APress, 2016 16. Pro ASP.NET Core MVC by Adam Freeman, Springer, 2016
Teaching Methodology	Lectures, Discussion, Independent Study, Seminars and Assignment
Evaluation Method	30% Internal assessment 70% External assessment

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 2019
Purpose of Course	To develop skills of software engineering in students
Course Objective	To provide insights about software engineering project planning, scheduling, SCM fundamentals, pattern based design and advanced UML concepts. Students would be able to do plan, design, analyze risk and manage a software development process efficiently after learning this course.
Pre-requisite	Basic concepts of software analysis and design
Course Out come	This would help students to analyze risk , perform scheduling and design any kind of system.
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 1.3. Risk Management 1.3.1 Software Risk 1.3.2 Risk Identification and Categories of Risk 1.3.3 Projection 1.3.4 Refinement 1.3.5 RMMM Plan 1.4. Change Management 1.4.1 Software Configuration Management 1.4.2 SCM Repository 1.4.3 SCM Process 1.4.4 Version Control and Change Control 1.5. Project, task and agile development tool 1.5.1 Introduction to project and agile management tool 1.5.2 Use of tool like trello / axiom / workspace or similar **Unit 2 : Advance UML** 2.1 Introduction to UML 2.2 Structural Modeling and Use Cases 2.3 Behavioral Modeling with UML 2.4 Advanced Modeling with UML 2.5 Metadata Integration with UML, MOF and XMI **Unit 3 : Web Engineering** 3.1 Attributes of web based application 3.2 Framework of Web engineering 3.3 Analyzing Web-Based system 3.4 Design of Web-Based Application 3.5 Testing of Web Application 3.6 Management Issues **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

P. Y. Deshmukh

	Unit 5 : Software Quality Assurance 5.1 Software Quality Assurance 5.2 Cost of Quality 5.3 Framework and Standards SQA Framework 5.4 SQA Plan 5.5 Components of Software Quality Assurance 5.6 Quality Standards: ISO and companion Standards, CMM, CMMI, Six- Sigma
Reference Book	1. Software Engineering A practitioner's approach – Roger S Pressman - Seventh Edition- McGraw Hill 2. Object Oriented Modeling Design - James Rumbaugh, Michael Blaha – PHI 3. An Integrated Approach to Software Engineering – Pankaj Jalote – Narosa 4. Object-Oriented Software Engineering- Timothy C. Lethbridge, Robert Laganiere- TMH, 2008 5. Software quality assurance – from theory to implementation- Daniel Galin- Pearson education 6. Software Engineering- A programming approach- D. Bell, I. Morrey-PHI 7. UML 2.0 in a Nutshell: A Desktop Quick Reference (In a Nutshell (O'Reilly))-Day Pilone,Neil Pitman-2nd edition-OiReilly 8. UML Distilled: A Brief Guide to the Standard Object Modeling Language, M. Fowler,,3rd edition, Addison-Wesley 9. Meta Object Facility (MOF) 2.0 Query/View/Transformation Specification, V1.1, Object Management Group Std 10. UML™ Bible, Tom Pender, John Wiley & Sons 11. 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
Evaluation Method	30% Internal assessment 70% External assessment

P. M. Desai

Course : 704 : Optimization Techniques

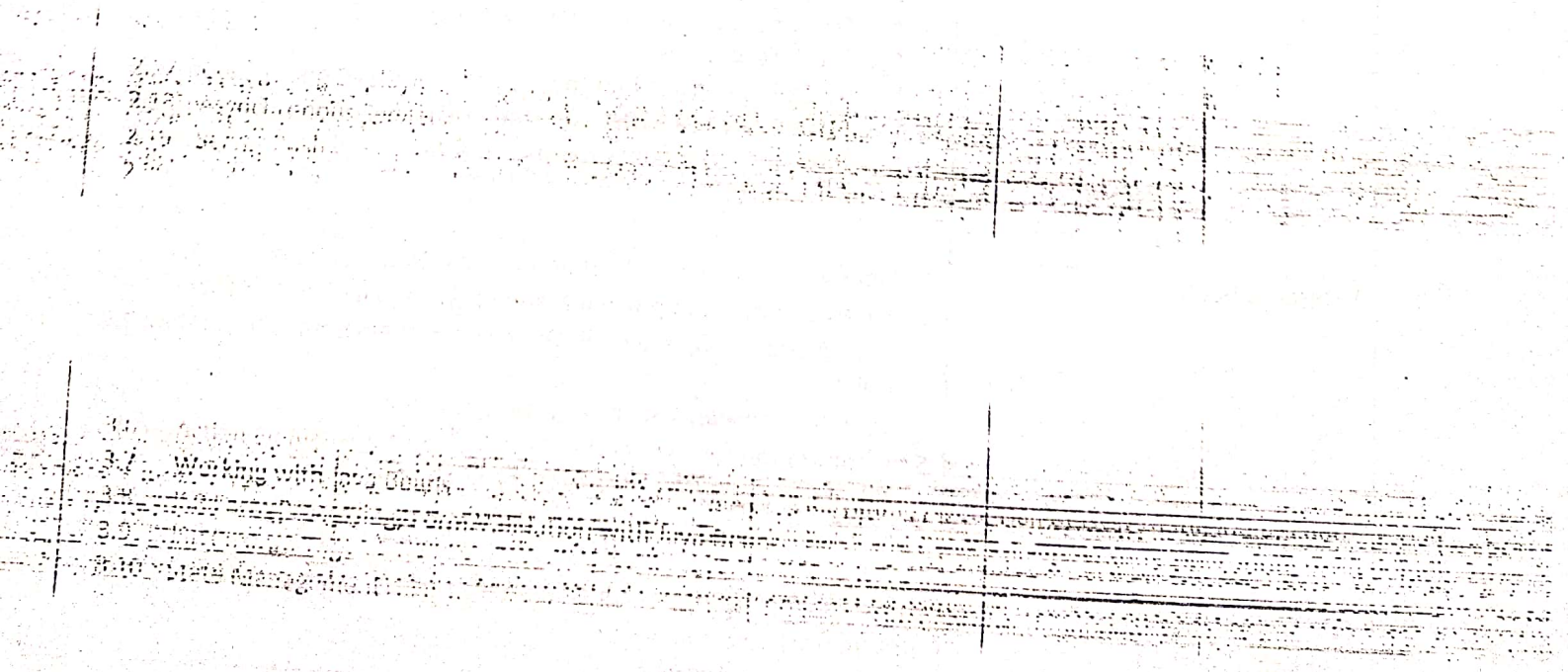
Course Code	704
Course Title	Optimization Techniques
Credit	4
Teaching per Week	4 Hrs
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Last Review / Revision	June 2019
Purpose of Course	The purpose of this course is to impart knowledge of various Optimization Techniques.
Course Objective	To impart knowledge of various Optimization Techniques to solve real world problem.
Pre-requisite	Basic knowledge of Mathematics.
Course Out come	Students will be able to apply Optimization Techniques for solving real world problems.
Course Content	**Unit : 1 : Job Sequencing** 1.1 Processing n jobs through 2 machines 1.2 Processing n jobs through 3 machines 1.3 Processing 2 jobs through m machines 1.4 Processing n jobs through m machines **Unit : 2 : Network Analysis** 2.1 Project Evaluation and Review Technique (PERT) 2.2 Critical Path Method (CPM) **Unit : 3 : Queuing Theory** 3.1 Essential features of queuing system 3.2 Performance measurement of queuing system 3.3 Classification of queuing model 3.4 Single server queuing model 3.5 Multi-server queuing model **Unit : 4 : Simulation** 4.1 Simulation Introduction 4.2 Types of simulation 4.3 Steps of simulation process 4.4 Advantages and disadvantages of simulation process 4.5 Stochastic simulation and random numbers **Unit : 5 : Dynamic programming** 5.1 Dynamic programming 5.2 Developing optimum decision policy 5.3 Dynamic programming under certainty 5.4 Shortest route problem 5.5 Multiple separable Return function and single additive constraints 5.6 Additive separable Return function and single additive constraints 5.7 Additive separable Return function and single multiplicative constraints
Reference Book	1. Operation Research, S. D. Sharma; Kedar Nath, Ram Nath & Co. 2. Kantiswarup, P.K.Gupta and Manmohan: Sultan Chand and Sons. 3. Introduction to Operation Research Computer Oriented algorithm; B.E. Gillet 4. Operation research an Introduction; H.A. Taha 5. Optimization for Engineering Design, Algorithms and Examples Prentice; Kalyanmoy Deb; Hall of New Delhi, India, 2000 6. PERT and CPM: Principles and Applications; 2nd edition, 1975; Srinath L.S.
Teaching Methodology	Lectures, Discussion, Independent Study, Seminars and Assignment
Evaluation Method	30% Internal assessment 70% External assessment

P. M. Dan

Course : IT 705 : Practical 15

Course Code	705
Course Title	Practical 15
Credit	3
Teaching Per Week	3 Hrs
Minimum Weeks Per Semester	15 (Including Practical Work, Examination, Preparation, Holidays etc.)
Review/Revision	June 2019
Purpose of Course	The course provides practical knowledge of web application development using full stack development frameworks like Node.js, AngularJs, Express, etc.
Course Objective	The course prepares students to develop light weight application using full stack development frameworks like Node.js, AngularJs, Express, etc.
Prerequisite	Object Oriented Programming Concepts
Course Outcome	After completion of this course, students will be able to develop light weight applications using full stack development frameworks like Node.js, AngularJs, Express, etc.
Course Content	Practical based on Paper No. 701 - Application Development using Full Stack.
Reference Books	NIL
Teaching Methodology	Lab Work
Evaluation Method	30% Internal Assessment 70% External Assessment

P. V. Das sir



Course : IT 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 2019
Purpose of Course	The course provides practical knowledge of C#, LINQ, .NET Core, MVC, etc.
Course Objective	The course prepares students to develop .NET based web applications.
Prerequisite	Object Oriented Programming Concepts
Course Outcome	After completion of this course, students will be able to develop .NET based web applications.
Course Content	Practical based on Paper No. 702- Application Development using .NET CORE.
Reference Books	NIL
Teaching Methodology	Lab Work
Evaluation Method	30% Internal Assessment 70% External Assessment

P. M. Desai

Course : 707 : Part Time Project 1

Course Code	707
Course Title	Part Time Project 1
Credit	3
Teaching Per Week	3 Hrs
Duration	-
Minimum Weeks Per Semester	15 (Including Practical Work, Examination, Preparation, Holidays etc.)
Review/Revision	June 2019
Purpose of Course	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.
Course Objective	To help students to develop software applications using AngularJS, Node.js and .NET.
Prerequisite	Knowledge of Object Oriented Programming, Web Technology Fundamentals, Software Engineering.
Course Outcome	After completion of this course, students will be able to develop software applications.
Course Content	The students are required to develop a project using .NET technologies and popular JavaScript based frameworks. 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)
Reference Books	NIL
Teaching Methodology	Project guidance, Review
Evaluation Method	30% Internal Assessment 70% External Assessment