5 Years Integrated MSc (Computer Science)
[Course Initiated in 2021-2023]

Semester - VII

Scheme & Course Structure

Applicable from July 2023

Course Structure with Credits

	Course Type	Subject Names & Credits	Total Credits
Semester-VII	Core Courses	• MCSCC411-System Software (4) • MCSCC412-Computer Oriented Numerical Methods (4) • MCSCC413- Computer Oriented Numerical Methods - Practical (2)	10
	Ability Enhancement Courses	• MCSAE411 – Data Visualization (2)	2
	Skill Enhancement Courses	• MCSSE411 - Project-5 (4)	4
	Discipline Specific Elective Courses	• MCSDE411 - Elective-2 (4) • MCSDE412 - Elective-2 – Practical (2)	6
	Generic Elective Courses	• MCSGE411 - Innovation and Startup (2)	2
	Semester-VII (Total Credits)		24

TEACHING & EXAMINATION SCHEME FOR 5 YEAR INTEGRATED M. Sc. (COMPUTER SCIENCE) COURSE

Semester-VII	Course		Hours per week								Credits
	No.	**Name**	Theory	Practical	Internal	Total	Lectures	Others (Tutorials)	Practical	Total	
Semester-VII	MCSCC411	System Software	70	***	30	100	3	1	***	4	4
	MCSCC412	Computer Oriented Numerical Methods	70	***	30	100	3	1	***	4	4
	MCSCC413	Computer Oriented Numerical Methods - Practical	***	35	15	50	***	***	4	4	2
	MCSAE411	Data Visualization	***	35	15	50	***	***	4	4	2
	MCSSE411	Project – 5	***	70	30	100	***	***	8	8	4
	MCSDE411	Elective-2	70	***	30	100	3	1	***	4	4
	MCSDE412	Elective-2 – Practical	***	35	15	50	***	***	4	4	2
	MCSGE411	Innovation and Startup	35	***	15	50	2	***	***	2	2
		Total	245	175	180	600	11	3	20	34	24

Theory/Tutorial Credits → 1 Hour = 1 Credit

Practical Credits → 2 Hours = 1 Credit

Track wise subjects for Elective-2			
Track	**Web Technologies**	**Artificial Intelligence & Machine Learning**	**Security**
Subject Code	MCSDE411(1)	MCSDE411(2)	MCSDE411(3)
Subject Name	Full Stack Web Development	Data Analytics	Network Security
Subject Code	MCSDE412(1)	MCSDE412(2)	MCSDE412(3)
Subject Name	Full Stack Web Development - Practical	Data Analytics – Practical	Network Security - Practical

Course Name: System Software

Course Code: MCSCC411

Objectives:
At the end of the course, student will be able to
- Understand the relationship between system software and machine architecture
- Know the design and implementation of assemblers, macro processor, linker and compiler
- Have an understanding of loader, system software tools
- Understand and know the working of device drivers

Prerequisites:
Basic knowledge of computer architecture and C programming

Contents:

1. **Introduction to System Software and software tools Language Processors**
   Introduction, Language Processing Activities, Fundamentals of Language Processing & Language Specification, Language Processor Development Tools. Data Structures for Language Processing: Search Data structures, Allocation Data Structures. Software Tools: Software Tools for Program Development, Editors, Debug Monitors, Programming Environments, and User Interfaces

2. **Assemblers & Macro Processors**
   Assemblers: Elements of Assembly Language Programming, A Simple Assembly Scheme, Pass Structure of Assemblers, Design of a Two Pass Assembler, A single pass Assembler for IBM PC.
   Macros and Macro Processors: Macro Definition and Call, Macro Expansion, Nested Macro Calls, Advanced Macro Facilities, Design of a Macro Preprocessor

3. **Compilers and Interpreters**
   Scanning and Parsing: Introduction to NFA and DFA, Approaches of parsing, Different types of parsing techniques namely, Recursive decent parser, LL(1) parser, Operator precedence parser. (First and follow technique for generating a parse table is to be taught), Phases of the Compiler, Aspects of compilation, Memory allocation. Compilation of expressions and control structures. Code optimization. Interpreters: Use and overview of interpreters, pure and impure interpreters

4. **Linkers, Loaders and Device Drivers**
   Introduction to linkers, Relocation and Linking Concepts. Design of a Linker, Self-Relocating Programs. A Linker for MS-DOS, Linking for Overlays. Introduction to Loaders, Design and anatomy of UNIX device driver, Types of device driver, General design of UNIX character device driver, General design of UNIX block device driver, UNIX device driver installation

References:

1. Dhamdhere D.M., "Systems Programming and Operating Systems", Tata McGraw Hill
2. Beck L., "System Software – An introduction to systems programming", Pearson Education
3. Pajari G., "Writing UNIX device drivers", Pearson Education
4. Chattopadhyay S., "System Software", Prentice Hall India
5. Aho A., Ulman J., Lam M., Sethi R., "Compilers: Principles, Techniques and Tools", Pearson Education
6. Louden K., "Compiler Construction: Principles and Practice", Cengage India

Accomplishments of the student after completing the Course:

At the end of the work students will be able to

- Good understanding of basic design and implementation of various system software
- Ability to design and implement better and more efficient programs/applications as a result of the knowledge gained

----

Course Name: Computer Oriented Numerical Methods

Course Code: MCSCC412

Objectives:
The main objective of this course is to enable students to
- Understand numerical methods for problems of numerical analysis
- Gain experience in the implementation of numerical methods using computers
- Appreciate the concept of error and the need to analyze and predict it

Prerequisites:
Foundations in Mathematics, Programming in C/C++

Contents:

1. **Introduction to Numerical Methods**
   Characteristic of Numerical Methods, Different Sources of Errors, Iterative Methods, Types of Errors, Quantification of Errors, Stopping Criterion in iterative methods

2. **Iterative Methods for finding roots of an equation f(x) = 0**
   Bisection ,False Position, Secant, Newton Raphson and Successive Approximation Method, Descarte's rule of sign, Budan's Theorem, Birge Vieta Method

3. **Polynomial Interpolation, Least Square Approximation**
   Lagrange, forward difference, backward difference, divided difference interpolation, Error Estimates, Cubic spline interpolation, Least square Curve fitting: Linear Regression and Nonlinear Regression, Approximation of Functions by Taylor series, Chebyshev Approximation, Chebyshev Economization

4. **Numerical Differentiation, Integration and Solution of Ordinary Differential Equations**

   Differentiation, Numerical Integration – Newton Cotes formulas, Trapezoidal, Simpson's 1/3 & 3/8 rules, Gauss Quadrature formulas, Taylor series, Euler, Runge-Kutta 2nd order and 4th order, Predictor Corrector methods: Milne Simpson & Adam's Moulton

References:

1. Vachharajani V., "Numerical Analysis – A programming approach", BPB Publications
2. Gerald C.F., Wheatley P.O., "Applied Numerical Analysis", Pearson Education
3. Chapra S., Canale R., "Numerical Methods for Engineers", Tata McGraw Hill Pub.
4. Somasundaram R., Chandrasekaran R., "Numerical Methods with C++ Programming", Prentice Hall Pvt. Ltd.
5. Datta N., "Computer Oriented Numerical Methods", Vikas Publications
6. Burden R., Faires D., "Numerical Analysis", Cengage Pubilcation
7. Mathews J., Fink K., "Numerical Methods Using MATLAB", Pearson Education

Accomplishments of the student after completing the Course:

After completion of this course, students will be able to

- Gain ability to solve linear and non-linear algebraic equations, perform operations of calculus, fit curves, and solve differential equations using a computer
- Appreciate problems due to rounding errors and convergence

Course Name: Computer Oriented Numerical Methods – Practical

Course Code: MCSCC413

Objectives:
The objective of the course is to enable students to
- Gain experience in the implementation of numerical methods using computers
- Appreciate the concept of error and the need to analyze and predict it

Prerequisites:
Foundations in Mathematics, Programming in C/C++

Contents:

1. **Finding Roots, Polynomial Interpolation**
   Implementation of Bisection Method, False Position Method, Secant Method, Newton Raphson Method and Successive Approximation Method, Validation of Descarte's rule of sign, Implementation of Birge Vieta Method, Implementation of Lagrange forward difference method, backward difference method, divided difference interpolation, Performing Error Estimates, Cubic spline interpolation, Implementing Least square Curve fitting: Linear Regression and Nonlinear Regression, Approximation of Functions by Taylor series, Implementation of Chebyshev Approximation

2. **Numerical Differentiation, Integration and Solution of Ordinary Differential Equations**

   Implementation of various Differentiation methods & Numerical Integration methods like – Newton Cotes formulas, Trapezoidal rule, Simpson's 1/3 & 3/8 rules, Gauss Quadrature formulas, Programming Taylor series, Euler, Runge-Kutta 2nd order and 4th order techniques, Implementing Predictor Corrector methods

References:

1. Vachharajani V., "Numerical Analysis – A programming approach", BPB Publications
2. Gerald C.F., Wheatley P.O., "Applied Numerical Analysis", Pearson Education
3. Chapra S., Canale R., "Numerical Methods for Engineers", Tata McGraw Hill Pub.
4. Somasundaram R., Chandrasekaran R., "Numerical Methods with C++ Programming", Prentice Hall Pvt. Ltd.
5. Datta N., "Computer Oriented Numerical Methods", Vikas Publications
6. Burden R., Faires D., "Numerical Analysis", Cengage Pubilcation
7. Mathews J., Fink K., "Numerical Methods Using MATLAB", Pearson Education

Accomplishments of the student after completing the course:
At the end of the course, students will be able to

- Solve linear and non-linear algebraic equations, perform operations of calculus, fit curves, and solve differential equations using a computer
- Appreciate problems due to rounding errors and convergence

Course Name: Data Visualization

Course Code: MCSAE411

Objectives:
The main objective of this course is to enable students to
- Gain knowledge of web data scraping
- Apply pre-processing on data
- Deliver data to web applications
- Generate efficient charts and plots

Prerequisites:
Fundamentals of Javascript, Fundamentals of Python

Contents:

1. **Data Processing with Python**
   Development Setup, Reading & Writing data in Python, Getting data off the web with Python, Scraping with Scrapy, Using Numpy, Using pandas, Cleaning data with pandas, Visualizing data with Matplotlib, Exploring data with pandas

2. **Delivering the Data**
   Delivering the data, RESTful data with Flask, Bringing charts to the web with Matplotlib & Plotly, A noble visualization, Building a Visualization, Using D3, Visualizing individual Prizes, Mapping with D3, Visualizing individual Winners, The Menu Bar

References:

1. Dale Kyran, "Data Visualization with Python and Javascript", O'Reilly Media
2. Camm, Cochran, et. al., "Data Visualization: Exploring and Explaining the Data", Cengage
3. Knaflic C.N., "Storytelling with Data", Wiley
4. Teller S., "Data Visualization with d3.js", Packt Publishing
5. Dixit R., "Data Analysis with Python", BPB Publications
6. Hajba G.L., "Website Scrapping with Python", Apress

Accomplishments of the student after completing the Course:
At the end of the work students will be able to

- Scrap data from web and apply preprocessing
- Create visualizations to be delivered on Web and for analytics

----

Course Name: Project - 5

Course Code: MCSSE411

Objectives:
The aim of this course is
- To introduce structured development of software systems
- To acquaint students to various techniques of requirements determination
- To introduce the concepts of analysis and design
- To model the system with various software diagrams
- To develop a system using software engineering concepts
- To prepare document/report of the system

Prerequisites:
Knowledge of Programming

Contents:

1. **Introduction to System Analysis & Design**
   Concepts of System Analysis & Design, Components of a System, Users of a System, System Development Life Cycle, Role of System Analyst, Various approaches to develop a system, Using a "systems" approach to build whole system,

2. **Structured Systems Development Approach**
   Requirements Gathering, surveying existing systems, Performing Feasibility Study before building a system, Formulating SRS, Designing Data Flow Diagrams, Entity-Relationship Diagrams, Designing Data Dictionary, Designing effective Input and Reports, Implementing EOD, EOM and EOY Procedures, Performing Testing for the System, Evaluating Security Risks for the System, Documenting the System

Guidelines for Project:
1. Projects can be created by a single student or in a group of 2-3 students.
2. The definition of project is to be submitted within 15 days of the starting of the semester.
3. The project should be free from plagiarism.
4. A student/group is required to report to their project guides on regular basis.
5. A student/group is required to present the project during internal exams.
6. It is advisable that the project must include different forms and reports.
7. It is necessary to prepare system diagrams, data dictionary as per the convention.
8. Project documentation should be prepared for the semester end examination.
9. In the semester end examination, the project should be presented to the examiners along with the documentation(report).

References:

1. Kendall & Kendall. "System Analysis and Design", Prentice Hall India (PHI)
2. James Sen, "Analysis and Design of Information System", PHI

3. Arthur Langer, "Analysis and Design of Information Systems", Springer
4. Jeff Johnson, "Designing with the Mind in Mind", Morgan Kaufmann
5. Dennis, Wixom, Roth "System Analysis and Design", Wiley India
6. Joseph Valacich, Joey F. George, "Modern Systems Analysis and Design", Pearson
7. Priti Srinivas Sajja., "Essence of System Analysis and Design", Springer

Accomplishments of the student after completing the Course:
After completion of this course, students will be able to

- Analyse and Design a system based on user requirements
- Develop a software system using software engineering concepts
- Prepare a document for the system

Course Name: Full Stack Web Development

Course Code: MCSDE411 (1)

Objectives:

The objective of the course is to enable students to
- Learn the concepts of front end and back end development
- Implement full-stack concepts in real-life applications

Prerequisites:

Knowledge of HTML, DHTML and Object-Oriented programming

Contents:

1. **Introduction to Java Script and JSON**

   Overview of Java Script, Primitives, Operations and Expressions, Screen Output and Keyboard Input, Control Statement, Object Creation and modification, Arrays, Functions, Constructors, Pattern Matching using regular expressions, Overview of JSON, Comparison of JSON with XML and CSV, Callback functions, Promise Objects, async…await, Overview of ES6

2. **AngularJS Core Concepts**

   Introduction to AngularJS, Advantages of Angular, AngularJS MVC, Introduction to SPA, Setting up the environment, First App using MVC architecture, Understanding attributes, Expression and Data Binding, Working with directives, Angular Modules, Controller, Scope and View, Create Controller and Module, $scope hierarchy, Introduction to Lifecycle hooks

3. **Filters, Forms and WebSocket**

   Filters - Built-in filters - upper case and lower case filters, date, currency and number formatting, orderBy, filter, custom filter, Angular JS Forms – Working with AngularJS forms, model binding, form controller, Using CSS classes, form events, custom model, update triggers, custom validation, $http service, Ajax implementation using $http, Web APIs, Overview of AJAX, Fundamentals of WebSocket

4. **Dependency Injection, Services, Routing and Navigation & NodeJS**

   Dependency injection, Using dependency injection, Angular JS service –Understanding services, Using built-in service, Creating custom service, Injecting dependency in service, Overview of Routing, Routing using ngRoute and UI- Router, ngView Directive, Configuring $routeProvider, $stateProvider, Animating Angular App, Overview of TypeScript, Introduction to Node.js, Features of Node.js, Setup Development Environment- Installing Node.js, Working with REPL, Node.js Console, Node.js Module, Node Package Manager, Node.js Basics, File System, HTTP and HTTPs, Creating Web Server- Handling http request, Node.js Callbacks, Node.js Events

References:

1. Brad Dayley, "Node.js, MongoDB, and AngularJS Web Development", 2nd edition, Addison-Wesley
2. Ivan Bayross, "Web Enabled Commercial Application Development Using HTML, DHTML, PERL, Java Script" ,Revised Edition, BPB publications
3. Adam Freeman, "Pro Angular JS", Apress edition, Wiley india
4. Agus Kurniawan, "AngularJS Programming by Example", PE Press
5. Amos Q. Haviv, "Mean Web Development", 2nd edition, Ingram short title
6. Mozilla Developers Network(MDN) Web Docs, https://developer.mozilla.org/en-US/
7. Angular Documentation, https://angular.io/docs
8. Nodejs Documentation, https://nodejs.org/en/docs/

Accomplishments of the student after completing the course:

At the end of the course, students will be able to

- Design the frontend and backend application after learning full stack web development
- Apply the newly available framework for web designing

Course Name: Data Analytics

Course Code: MCSDE411 (2)

Objectives:

The objective of the course is to enable students to

- Understand key concepts of data, data mining, data analysis and data analytics
- Understand how data is created, stored, accessed
- Understand tools to effectively organize and visualize data
- Put the principles and methods of statistical analysis into practice using a range of real-world data sets

Prerequisites:

Knowledge of linear algebra, calculus, Python language and generic libraries

Contents:

1. **Introduction**

   Introduction to data, Data Analytics/Mining/Science, data sets, features, data scales, categorical and numerical data, cross-sectional and time series data. Univariate, bivariate and Multivariate data. Set and matrix representations, relations. Data Pre-processing: Handling missing values, normalization of data, dimensionality reduction, and outlier reduction. Introduction to Python libraries for data handling and statistics: Statistics library, SciPy, Pandas, NumPy

2. **Descriptive statistics**

   Graphical & Tabular methods

   Summarizing data. Frequency Distribution Relative Frequency and Percent Frequency distributions, cross-tabulations. Bar Charts, Scatter plots, line chart, area chart and Pie Charts, stem and leaf display. Dot Plot, Histogram Cumulative Distributions, Ogive. Data summarization using pandas and data visualization using MatplotLib libraries of Python

   Numerical methods

   Measures of location: Mean median, mode, percentiles, quartiles,

   Measures of variability: range, interquartile range, variance, standard deviation, coefficient of variation,

   Measures of association between two variables: covariance interpretation of the covariance correlation coefficient interpretation of the correlation coefficient, Measures of distribution: shape, relative location, and detecting outliers z-scores empirical rule detecting outliers box plot, Handling descriptive statistics through python programs and/or libraries

3.   **Regression models, Probability & Probability Distribution**
Regression Models
Simple and Multiple linear regression model, least squares method, coefficient of determination model assumptions testing for significance, estimate of σ2, t-test, and confidence interval. Implementing regression models using Sci-kit library
Probability and Probability distribution
Counting rules, combinations, and permutations assigning probabilities, events and their probabilities, some basic relationships of probability complement of an event addition law, conditional probability independent events multiplication law. Random Variables, Discrete vs. Continuous random variable. Discrete probability distribution: Binomial, Poisson, Hyper geometric. Continuous probability distribution: Normal and standard normal distributions. Implementation of distributions using python libraries

4.   **Hypothesis Testing, Time Series & Forecasting**
Focusing on Key Performance Indicators , Using KPI's for Web 2.0 , KPI's Setting Objectives and Key Results, Demonstrate how web analytics fits into the business structure
Real World Tasks, Identifying and Optimizing Poorly Performing Pages , Search Engine Optimizing, Interpret the business benefits of web analytics

References:

1. Anderson, David r., et al, "Statistics for business & economics", Cengage learning
2. Thomas a. Runkler. – Wiesbaden, *Data analytics: models and algorithms for intelligent data analysis*", Springer (cop. 2012), Verlag
3. Parag Kulkarni, Sarang Joshi,, Meta S. Brown, "Big data analytics"
4. Soraya Sedkaoui, "Data analytics and big data", Wiley
5. Emc Education Services, "Data science and big data analytics", Wiley

Accomplishments of the student after completing the course:
At the end of the course, students will be able to

- Apply key concepts of Data Analytics
- Use Data Analytics tools to create, store and access the data and information
- Implement tools to organize and visualize the data effectively
- Perform methods of statistical analysis with a range of real-world data

----

Course Name: Network Security

Course Code: MCSDE411 (3)

Objectives:

The objective of the course is to enable students to

- Understand the state-of-the-art in network security and computer security
- Study the security mechanisms at various layers of TCP/IP Model
- Develop strong analysis, testing and troubleshooting skills regarding to computer network security
- Implement various open source tools related to network security

Prerequisites:

Basics of Computer Networking

Contents:

1. **Network Security Fundamentals, Issues and Challenges**
   <u>Computer Network Security Fundamentals</u>
   Introduction, Securing the Computer Network, Forms of Protection, Security Standards
   <u>Security Threats and Threat Motives to Computer Networks</u>
   Sources of Security Threats, Security Threat Motives, Security Threat Management, Security Threat Correlation, Security Threat Awareness
   <u>Computer Network Vulnerabilities</u>
   Definition, Sources of Vulnerabilities, Vulnerability Assessment
   <u>Cybercrimes and Hackers</u>
   Introduction, Cybercrimes, Hackers, Dealing with Cybercrimes
   <u>Security Assessment, Analysis, and Assurance</u>
   System Security Policy, Building a Security Policy, Security Requirements Specification, Threat Identification, Threat Analysis, Vulnerability Identification and Assessment, Security Certification, Security Monitoring and Auditing, Products and Services

2. **Cryptography & Access Control, Authorization and Authentication**
   <u>Cryptography</u>
   Overview of Cryptography, Classical Algorithms, Symmetric Key Cryptography, Asymmetric Key Cryptography, Public Key Cryptography, Integrity Check Algorithms, Authentication Algorithms, Overview of Key Management
   <u>Access Control and Authorization</u>
   Definitions, Access Rights, Access Control Systems, Authorization, Types of Authorization System, Authorization Principles, Authorization Granularity, Web Access and Authorization
   <u>Authentication</u>
   Definition, Multiple Factors and Effectiveness of Authentication, Types of Authentication, Authentication Methods, Developing an Authentication Policy

3. **Firewalls, IDS and IPS**
   <u>Firewalls</u>

Definition, Types of Firewalls, Configuration and Implementation of a Firewall, The Demilitarized Zone (DMZ), Improving Security Through the Firewall, Firewall Forensics, Firewall Services and Limitations

<u>Intrusion Detection & Prevention</u>

Definition, Intrusion Detection, Intrusion Detection Systems (IDSs), Types of Intrusion Detection Systems, The Changing Nature of IDS Tools, Other Types of Intrusion Detection Systems, Response to System Intrusion, Challenges to Intrusion Detection Systems, Implementing an Intrusion Detection System, Intrusion Prevention Systems (IPSs), Intrusion Detection Tools

4.   **Dealing with Computer Network Security Challenges**

<u>Computer and Network Forensics</u>

Introduction, Computer Forensics, Network Forensics, Forensic Tools

<u>Virus and Content Filtering</u>

Definitions, Scanning, Filtering, and Blocking, Virus Filtering, Content Filtering, Spam

<u>Standardization and Security Criteria</u>

Introduction, Product Standardization, Security Evaluations, Major Security Evaluation Criteria

<u>Computer Network Security Protocols</u>

Introduction, Application-Level Security, Security in the Transport Layer, Security in the Network Layer, Security in the Physical Layer

<u>Security Beyond Wired Networks</u>

Overview of Security in Wireless Networks and Sensor Networks, Security in Cloud Environment, Security in Virtualization, Need of Security in IoT and Mobile Systems

Case Study of Open Source Tools
Nmap utility, Wireshark, iptables, OpenSSL, OSSEC, OpenVAS

References:

1. Kizza J.M., "Guide to Computer Network Security", 4th Ed., Springer International Publishing
2. Kaufman C., Perlman R., Speciner M., "Network Security: Private Communication in Public World", Prentice Hall India
3. Perez A., "Network Security", John Wiley & Sons
4. Stallings W., "Network Security Essentials – Applications and Standards", Pearson India
5. Forouzan B., "Cryptography and Network Security", Tata McGraw Hill
6. Howlett T., "Open Source Security Tools", Prentice Hall

Accomplishments of the student after completing the Course:
At the end of the course, students will be able to

- Analyze and develop solutions to solve network security related problems.
- Have thorough understanding of various standards and protocols related to computer network security

- Effectively use various tools and utilities for computer networking.

Course Name: Full Stack Web Development - Practical

Course Code: MCSDE412 (1)

Objectives:
The objective of the course is to enable students to
- Implement the concepts of front end and back end development
- Implement full-stack programming features in real-life applications

Prerequisites:
Knowledge of HTML, DHTML and Object-Oriented programming

Contents:

1. **Java Script, JSON & AngularJS**
   Java Script
   Basic Programming in Java Script, Use of Primitives, Operations and Expressions, Screen Output and Keyboard Input, Control Statement, Object Creation and modification, Arrays, Functions, Constructors, Pattern Matching using regular expressions, Creating XML and JSON documents, Implementation of Callback functions, Promise Objects, async…await
   AngularJS
   Implementing simple programs using AngularJS, Using MVC architecture, Implementing expression and data binding, Using directives, Using Angular Modules, Creating Controller, Scope, View and Modules, Using AngularJS forms, Applying various filters and formatting, Using AngularJS services, Creating custom services

2. **WebSocket & NodeJS**
   Ajax implemention using $http and Web APIs, Implementation of AJAX programs, Implementation of WebSockets, Using dependency injection, Setting up NodeJS environment, Using Node.js console, Using Node.js module, Using Node Package Manager, Implementing programs that uses file system, Using Node.js callbacks, Using Node.js events, Creating webserver and handling requests

References:

1. Brad Dayley, "Node.js, MongoDB, and AngularJS Web Development", 2nd edition, Addison-Wesley
2. Ivan Bayross, "Web Enabled Commercial Application Development Using HTML, DHTML, PERL, Java Script" ,Revised Edition, BPB publications
3. Adam Freeman, "Pro Angular JS", Apress edition, Wiley india
4. Agus Kurniawan, "AngularJS Programming by Example", PE Press
5. Amos Q. Haviv, "Mean Web Development", 2nd edition, Ingram short title
6. Mozilla Developers Network(MDN) Web Docs, https://developer.mozilla.org/en-US/
7. Angular Documentation, https://angular.io/docs
8. Nodejs Documentation, https://nodejs.org/en/docs/

Accomplishments of the student after completing the course:
At the end of the course, students will be able to

- Design the frontend and backend application after learning full stack web development
- Apply the newly available framework for web designing

Course Name: Data Analytics – Practical

Course Code: MCSDE412 (2)

Objectives:
The objective of the course is to enable students to
- Understand key concepts of data, data mining, data analysis and data analytics
- Understand how data is created, stored, accessed
- Understand tools to effectively organize and visualize data
- Put the principles and methods of statistical analysis into practice using a range of real-world data sets

Prerequisites:
Knowledge of linear algebra, calculus, Python language and generic libraries

Contents:

1. **Data Analytics Tools & Technologies**
   Features of R, Python for Data Analytics, Using Pandas & Matplotlib for Data Analytics, Data extraction and cleaning, Data transformation and normalization, Data visualization and exploration

2. **Descriptive & Diagnostic Analysis**
   Measures of central tendency (mean, median, mode), Measures of dispersion (variance, standard deviation, range), Boxplots and histograms, Correlation analysis, Regression analysis, Outlier detection

References:

1. Anderson, David r., et al, "Statistics for business & economics", Cengage learning
2. Thomas a. Runkler. – Wiesbaden, "Data analytics: models and algorithms for intelligent data analysis", Springer (cop. 2012), Verlag
3. Parag Kulkarni, Sarang Joshi,, Meta S. Brown, "Big data analytics"
4. Soraya Sedkaoui, "Data analytics and big data", Wiley
5. Emc Education Services, "Data science and big data analytics", Wiley

Accomplishments of the student after completing the course:
At the end of the course, students will be able to

- Apply key concepts of Data Analytics
- Use Data Analytics tools to create, store and access the data and information
- Implement tools to organize and visualize the data effectively
- Perform methods of statistical analysis with a range of real-world data

----

Course Name: Network Security - Practical

Course Code: MCSDE412 (3)

Objectives:
The objective of the course is to enable students to
- Understand the state-of-the-art in network security and computer security
- Study the security mechanisms at various layers of TCP/IP Model
- Develop strong analysis, testing and troubleshooting skills regarding to computer network security
- Implement various open source tools related to network security

Prerequisites:
Basics of Computer Networking

Contents:

1. **Cryptography Tools:**
   NMAP & Wireshark
   Scanning the network using Nmap for open ports, Analyzing the protocols and network using Wireshark, Using various filters of Wireshark
   OpenSSL
   Setting up environment for OpenSSL, Generating random passwords with OpenSSL, Generating Public and Private Key using OpenSSL, Encrypting files and strings using DES, 3DES, AES, RSA algorithm, Using different cipher modes in OpenSSL, Creating Message Digests using OpenSSL, Creating and Installing certificates using OpenSSL

2. **Firewalls & IDS/IPS**
   Firewall
   Enabling/Disabling a Firewall, Setting up rules for firewall to drop packets, Setting up rules for firewall to accept packets
   IDS/IPS
   Setting up environment for IDS/IPS, Applying signatures, Using sniffer mode, Using packet logger mode, Using NIDS mode, Implementing snort rules format, using snort command line options, Setting up alert rules
   OSSEC and OpenVAS
   Overview and demonstration of OSSEC and OpenVAS tools

References:

1. Kizza J.M., "Guide to Computer Network Security", 4th Ed., Springer International Publishing
2. Kaufman C., Perlman R., Speciner M., "Network Security: Private Communication in Public World", Prentice Hall India
3. Perez A., "Network Security", John Wiley & Sons

4. Stallings W., "Network Security Essentials – Applications and Standards", Pearson India
5. Forouzan B., "Cryptography and Network Security", Tata McGraw Hill
6. Howlett T., "Open Source Security Tools", Prentice Hall

Accomplishments of the student after completing the Course:
At the end of the course, students will be able to

- Analyze and develop solutions to solve network security related problems
- Have thorough understanding of various standards and protocols related to computer network security
- Effectively use various tools and utilities for computer networking

----

Course Name: Innovation and Startup

Course Code: MCSGE411

Objectives:
The main objective of this course is to enable students to
- Gain knowledge regarding building a startup
- Prepare an organization for Innovation
- Apply innovation in startups

Prerequisites:
Fundamentals of Computers

Contents:

1. **Building a Startup**
   Starting with an Idea, Making a Business Plan, Securing Funds for Startup, Identifying the right people to be Involved, Following Legal Steps, Setting up a Location, Developing a Marketing Plan, Setting up a Customer Base, Preparing the organization for Innovation, Promoting and Communication, Personal creativity for Entrepreneurs, Corporate Creativity and Innovation, Innovation theory for Entrepreneurs, Business Essentials for Innovative Entrepreneurs, Innovation Readiness and Deployment

2. **Innovation and Startup**
   Innovation Process Model, Innovation and Consumer/Customer Insight, Building a strong Organizational Foundation for Innovation, Innovation Methods, Managing an Intellectual Property Portfolio, Importance of STEM and TRIZ in an Innovation Infrastructure, Global Innovation Problem Areas and Quality of R&D Ideas, Total Innovation Management for Excellence (TIME)

References:

1. Nager M., Nelsen C., Nouyrigat Y., "Startup Weekend: How to Take a Company From Concept to Creation in 54 Hours", John Wiley & Sons
2. Fernandex et. al., "The framework for innovation: an entrepreneur's guide to the body of Innovation Knowledge", CRC Press
3. Blank S., Dorf B., "The Startup Owner's Manual", John Wiley & Sons
4. Kennard M., "Innovation and Entrepreneurship", Routledge

Accomplishments of the student after completing the Course:
At the end of the work students will be able to

- Build their own startup
- Sustain the startup with innovations