

M.Sc. Computer Science

Syllabus

(For students admitted from 2021-2022 onwards)



**Department of Computer Science
Bishop Heber College (Autonomous)**

**Nationally Re-accredited at the 'A' by NAAC with a CGPA of 3.58 out of 4
Recognized by UGC as " College of Excellence"
Tiruchirappalli 620017**

Department of Computer Science

Vision

The Department of Computer Science is driven to provide excellent educational opportunities that accomplished the needs of our students, and empower them as an active technocrat in the top – notch IT industry and nation building.

Mission

- Facilitating the quality technical education through enriched curriculum to solve the real-world problems.
- Creating the knowledge of innovative and sustainable research areas of computational science to build technological advanced society/nation.
- Educating the professional ethics, attitude, human values and career building skills for their professional and personal life.

Programme Outcomes

Upon completion of MSc Computer Science degree, graduates will possess the following computer science skills and abilities.

PO1. Acquire the knowledge of mathematical foundations, algorithmic principles, computer science theory and data analytics in modelling and designing of computer-based applications.

PO2. Recognize, create and analyse computing problems in appropriate domains and to provide effective solution in the area of computing.

PO3. Design solutions for complex problems including design of experiments, design of system components or processes that meet specified needs for societal and nation development.

PO4. Efficient usage of modern computing IT tools and configure the software tools by understanding its limitations for design and development applications.

PO5. Interact effectively with teams to accomplish shared computing designs, evaluation and implementation goals.

PO6. Comprehend to write effective reports, design a valid documentation, make interactive presentations, ability to provide and obtain clear instructions

PO7. Propose the state of art in some of the areas of interest and provide solutions, resulting with a modern, user friendly tool with prolonged existence.

PO8. Understand professional and ethical responsibilities and analyze the impact of computing on individuals, organizations, and the society.

PO9. Recognize the need and have the ability to engage in independent and life-long learning in the widest context of technological change.

Programme Specific Outcomes

PSO1. Demonstrate and solve complex problems in the domain of Computer Science using software engineering best practices in the core knowledge areas such as Algorithms, Networking, Web design, Cloud Computing, Internet of Things and Data analytics.

PSO2. Acclimatize for rapid changes in tools and technologies to function in multi-disciplinary work environment, having good interpersonal skills as a leader of a team with addition to appreciation in professional and societal responsibilities

PSO3. Develop the abilities and skills to engage in independent and lifelong learning, teaching, Research & Development activities in India or Abroad in areas of their choice in computer science.

PSO4. Possess familiarity and practical proficiency in broad area of programming concepts which provide new ideas and innovations, in order to become an Entrepreneur and Software Engineer.

Programme Articulation Matrix

Sem	Course Code	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PSO1	PSO2	PSO3	PSO4
I	P21CS101	H	H	H	H	H	H	M	M	M	H	H	H	H
	P21CS102	H	H	H	H	H	H	M	M	M	H	H	H	H
	P21CS103	H	M	H	L	L	H	H	-	-	H	H	H	H
	P21CS1:1	H	M	M	M	M	M	M	H	H	H	H	M	M
	P21CS1P1	H	H	H	H	H	H	M	M	M	H	H	H	H
	P21CS1P2	H	H	H	H	H	H	M	M	M	H	H	H	H
II	P21CS204	H	H	H	H	H	H	-	-	-	H	H	H	H
	P21CS205	H	H	H	H	H	H	M	M	M	H	H	H	H
	P21CS206	H	H	H	H	H	H	-	-	-	H	H	H	H
	P21CS2:2	H	H	H	H	H	H	-	-	-	H	H	H	H
	P21CS2:3	H	H	H	H	H	H	M	M	M	H	H	H	H
	P21CS2P3	H	H	H	H	H	H	M	M	M	H	H	H	H
	P21CS2P4	H	H	H	H	H	H	M	M	M	H	H	H	H
	P18VL2:1 P18VL2:2						H	H	H	H			H	H
III	P21CS307	H	H	H	H	H	H	M	M	M	H	H	H	H
	P21CS308	H	H	H	H	H	H	M	M	M	H	H	H	H
	P21CS309	H	H	H	H	H	H	-	-	-	H	H	H	H
	P21CS3:4	H	H	H	H	H	H	-	-	-	H	H	H	H
	P21CS3P5	H	H	H	H	H	H	M	M	M	H	H	H	H
	P21CS3P6	H	H	H	H	H	H	M	M	M	H	H	H	H
IV	P21CS410	H	H	H	H	H	H	-	-	-	H	H	H	H
	P21CS411	H	H	H	H	H	H	-	M	-	H	H	H	H
	P21CS4:5	H	H	H	H	H	H	-	M	-	H	H	H	H
	P21CS4PJ	H	H	H	H	H	H	M	M	M	H	H	H	H

Programme Structure
M. Sc., Computer Science
(Applicable to Candidates admitted from the Academic Year 2021-2022 onwards)

Sem	Course	Course Title	Course Code	Hours / Week	Credits	Marks		
						CIA	ESE	Total
I	Core I	Data Analytics	P21CS101	5	4	25	75	100
	Core II	Mobile Application Development	P21CS102	5	4	25	75	100
	Core III	Mathematical Foundation for Computer Science	P21CS103	5	4	25	75	100
	Elective I	1a) Managing Software Development 1b) Design and Implementation of Compilers 1c) Multimedia Systems and Design	P21CS1:1 P21CS1:A P21CS1:B	5	4	25	75	100
	Core Practical I	Data Analytics Lab	P21CS1P1	5	3	40	60	100
	Core Practical II	Mobile Application Development Lab	P21CS1P2	5	3	40	60	100
II	Core IV	Advanced Programming with .Net	P21CS204	4	4	25	75	100
	Core V	Real Time Web App Development	P21CS205	4	4	25	75	100
	Core VI	Cloud and Big Data Computing	P21CS206	4	4	25	75	100
	Elective II	2a) Network Management 2b) Advanced Microprocessors And Microcontrollers 2c) Client Server Computing	P21CS2:2 P21CS2:A P21CS2:B	4	4	25	75	100
	Elective III	3a) Block Chain Technologies 3b) Virtual and Augmented Reality 3c) Information Retrieval	P21CS2:3 P21CS2:C P21CS2:D	4	4	25	75	100
	Core Practical III	Advanced Programming with .Net Lab	P21CS2P3	4	3	40	60	100
	Core Practical IV	Real Time Web App Development Lab	P21CS2P4	4	3	40	60	100
	VLO	RI/MI	P21VL2:1 P21VL2:2	2	2	25	75	100
III	Core VII	Image and Video Analysis	P21CS307	5	4	25	75	100
	Core VIII	Machine Learning	P21CS308	5	4	25	75	100
	Core IX	Cognitive Computing	P21CS309	5	4	25	75	100
	Elective IV	3a) Design and Implementation of Data Warehouse 3b) Mobile Computing 3c) Distributed Object Technology	P21CS3:4 P21CS3:A P21CS3:B	5	4	25	75	100
	Core Practical V	Image and Video Analysis Lab	P21CS3P5	5	3	40	60	100
	Core Practical VI	Machine Learning Lab	P21CS3P6	5	3	40	60	100
IV	Core X	Neural Networks and Deep Learning	P21CS410	5	4	25	75	100
	Core XI	Network Security	P21CS411	5	4	25	75	100
	Elective V	5a) Internet of Things 5b) Supply Chain Management 5c) Linked Open Data and Semantic Web	P21CS4:5 P21CS4:A P21CS4:B	5	4	25	75	100
	Core Project	Project	P21CS4PJ	---	8	---	---	100
			Total Credits		90			

CORE I: DATA ANALYTICS

SEMESTER: I
CREDITS: 4

CODE: P21CS101
HOURS/WEEK: 5

1. COURSE OUTCOMES

On successful completion of this course, students will be able to:

CO #	Course Outcomes	Level	Unit Covered
CO1	Experiment with the basic Python commands	K3	I
CO2	Develop array operations using NumPy and pandas data structures	K3	II
CO3	Discover data loading operations in Python	K4	III
CO4	Evaluate data wrangling methods in Python	K5	III
CO5	Construct Plotting, Visualization, Data Aggregation and Group operations	K6	IV
CO6	Create applications in Time Series and Financial analysis	K6	V

2. A. SYLLABUS

UNIT – 1: Python – Preliminaries and Environment

Significance of python in Data Analysis -Installation and setting up Python-Discussion on the essential libraries of Python- An Example – “Counting Time Zones” -Basics of IPython: Commands.

Usage of Command history in Python :Interacting with Operating System using Shell commands-Software development tools used with PythonHTML Note books in IPython-Productive code development using IPython.

UNIT – 2 Numpy and Panda Libraries in Python

The Numpy n-d array, Basic Operations on Array-Boolean indexing, Fancy indexing, Transposing and swapping-Universal Functions: Element wise array functions-Data processing with arrays-File input and output with arrays. Ex: Random Walks.

Introduction to Panda Data structure: Essential panda functionalities -Computing and summarizing descriptive statistics-Handling Missing Data-Hierarchical indexing.

UNIT – 3 Data Loading and Wrangling in Python

Reading and writing data in text format- Exploring Binary data formats-Interacting with HTML Language Interaction with MongoDB -Merging and combining of Datasets-Reshaping and pivoting process in python.

Data Transformations: Different types of String Manipulations-Introduction to USDA Food Database Working with USDA Database.

UNIT – 4 Plotting, Visualization, Data Aggregation and Group operations

Introduction to Matplotlib API - Plotting functions used in Panda-Python visualization and Tool Eco-System-Usage of Group-by mechanics in Python-Data Aggregation-Group wise operations and transformations-Quantile and bucket analysis- Filling Missing Values

Working Example: Random Sampling and Permutation-Pivot tables and Cross tabulation.

UNIT – 5 Time series, Financial and Data Applications

Date and Time data types - Basics of Time Series-Data Ranges, Frequencies and shifting-Time Zone Handling-Period and Period Arithmetic-Resampling and Frequency Conversion-Data Mingling basics.

Introduction to group transforms and analysis: Signal Frontier Analysis- Future Contract Rolling

B. TOPICS FOR SELF STUDY

- Data Analysis using Scrapy
- Data Analysis using Scikit Learn
- Visualization using Seaborn

- Visualization using Bokeh

C. TEXT BOOK(S)

1. Wes. Mc Kinney , “*Python for Data Analysis*”, First Edition,[O’Reilly], 2017 (**Chapters 1,2,3,4,5,6,7,8,9,10,11**)

D. REFERENCE BOOK(S)

1. Cyrille Rossant. Learning I “*Python for interactive Computing and data visualization*”, First edition [Packt].

E. WEB LINKS

- <https://docs.scrapy.org/en/latest/intro/tutorial.html>
- <https://scikit-learn.org/stable/tutorial/index.html>
- <https://seaborn.pydata.org/tutorial.html>
- <https://hub.gke2.mybinder.org/user/bokeh-bokeh-notebooks-9jp7mbkr/notebooks/tutorial/00%20-%20Introduction%20and%20Setup.ipynb>

3. SPECIFIC LEARNING OUTCOMES

Unit	Course Content	Learning Outcomes	Level
	After the completion of a topic in an unit, students will be able to		
I	Python preliminaries and Environment		
1.1	Significance of Python in Data Analysis	Recognize the importance of Python in data analysis	K2
1.2	Installation and setting up Python	Experiment with the installation and setup procedure of Python using different operating systems	K3
1.3	Discussion on the essential libraries of Python	Outline the essential Python libraries	K2
1.4	An example- “Counting Time Zones”	Create an example using DataFrame to count time zones with pandas	K6
1.5	Basics of IPython: Commands	Interpret the basic IPython commands using relevant examples	K2
1.6	Interacting with Operating System using Shell commands	Summarize IPython system-related Shell commands and Aliases	K2
1.7	Software development tools used with PythonHTML Notebooks in IPython	Explain different software development tools used in IPython	K2
		Execute IPython debugger commands in IPython HTML notebook	K3
1.8	Productive code development using IPython	Make use of the tips for productive code development using IPython	K3
II	Numpy and Panda libraries in Python		
2.1	The NumPy nd array: Basic operations on arrays	Develop an nd array	K3
		Analyse the basic indexing and slicing methods	K4
		Apply Boolean and Fancy Indexing methods	K3
2.2	Universal functions	Demonstrate the different unary and binary element-wise array functions	K2
2.3	Data processing using arrays	Analyze the various mathematical and statistical methods	K4
		Explain the sorting operations	K5

		Illustrate the various array set operations	K2
2.4	File input and output with arrays	Implement file input and output operations with arrays in both text and binary formats	K3
2.5	An example: Random Walks	Illustrate array operations in the simulation of random walks	K2
2.6	Essential Panda functionalities	Relate the fundamental mechanics with the data contained in a Series or DataFrame	K2
		Interpret the Panda functionalities such as Reindexing, Indexing, Selection, Filtering, Sorting and Ranking	K5
2.7	Computing and summarizing descriptive statistics	Inspect the various descriptive summary statistics	K4
		Illustrate unique values, value counts and memberships	K2
2.8	Handling missing data	Assess all the missing data handling methods	K5
2.9	Hierarchical indexing	Create a Series using hierarchical indexing	K6
III	Data Loading and Wrangling in Python		
3.1	Reading and writing data in text format	Explain read and write operations in text and delimited formats	K5
		Discuss read and write operations with HTML and XML formats	K6
3.2	Exploring binary data formats	Analyze HDF5 format for storing data	K4
3.3	Interacting with HTML language Interaction with MongoDB	Relate Python with HTML	K2
		Demonstrate data storing and loading process in MongoDB	K2
3.4	Merging and combining of datasets	Study merge operation on index	K4
		Inspect concatenation data combination	K4
3.5	Reshaping and Pivoting process in Python	Apply reshaping with hierarchical indexing	K3
		Perform pivoting process to change data format	K6
3.6	Data Transformations	Elaborate upon the various data transformation operations such as filtering and cleaning	K6
3.7	Different types of String manipulations	Interpret all the Python built-in string methods and vectorized string methods	K2
		Assess the regular expression methods	K5
3.8	Introduction to USDA Food database-Working with USDA database	Examine USDA Food database and work with it using Python functions	K4
IV	Plotting, Visualization, Data Aggregation and Group operations		
4.1	Introduction to matplotlib API	Understand about figures, subplots, colors, markers, line styles, ticks, labels, legends and annotations in matplotlib	K2
4.2	Plotting functions used in pandas	Build line plots, bar plots, histograms, density plots and scatter plots	K3
4.3	Python visualization and Tool Eco system	Compare Python visualization tools such as Chaco and Mayavi	K4
4.4	Usage of GroupBy mechanics in Python	Perform the various grouping operations in Python	K3
4.5	Data Aggregation	Implement data aggregation methods such as count, sum, mean, median, std, var, min, max, prod, first and last	K3
4.6	Group-wise operations and transformations	Experiment with transform and apply methods	K3
		Perform quantile and bucket analysis	K3
		Solve an example for filling missing values with group-specific values	K3

4.7	Working example: Random sampling and Permutation	Construct an example for Random sampling and Permutation	K6
4.8	Pivot tables and cross tabulation	Analyze the usage of pivot tables	K4
		Compute group frequencies using cross tabulation	K3
V	Time series, Financial and Data applications		
5.1	Date and Time data types	Examine datetime module types	K4
5.2	Basics of Time Series	Utilize Time Series basics such as Indexing, Selection and Subsetting in data analysis	K3
5.3	Date Ranges, Frequencies and Shifting	Generate Date Ranges	K6
		Create Frequencies and Data Offsets	K6
		Elaborate on Shifting (Leading and Lagging) data	K6
5.4	Time zone handling	Operate with Time zone-aware Timestamp objects	K3
5.5	Period and period arithmetic	Analyze Period frequencies	K4
		Change Timestamps to Periods	K6
		Create a PeriodIndex from arrays	K6
5.6	Resampling and frequency conversion	Apply Upsampling, Downsampling and Interpolation in Time series data	K3
5.7	Data Munging topics	Infer about Time series and cross section alignment	K4
		Build operations with Time Series of different frequencies	K3
		Interpret Splicing operation with data sources	K5
5.8	Introduction to group transforms and analysis	Analyze group factor exposures	K4
		Compare Decile and Quartile analysis	K4
5.9	Signal Frontier Analysis and Future Contract Rolling	Design an example application for Signal Frontier Analysis	K6
		Create an example application for Future Contract Rolling	K6

4. MAPPING SCHEME (CO, PO & PSO)

P21CS101	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PSO1	PSO2	PSO3	PSO4
CO1	H		L		L								
CO2	H	M	M	M	H	H							
CO3	H	H	H	H	M	L	M		H	M	H	L	
CO4	H	H	M	L	H			H	M	H	M	M	H
CO5	H			H	H		H	H	H	H	H	M	H
CO6	H		H	H	H		H	H	H	H	H	H	H

5. COURSE ASSESSMENT METHODS

DIRECT:

1. Continuous Assessment Test: T1, T2 (Theory & Practical Components): Closed Book
2. Open Book Test.
3. Cooperative Learning Report, Assignment, Group Presentation, Group Discussion, project Report, Field Visit Report, Poster Presentation, Seminar, Quiz (written).
4. Pre-Semester & End Semester Theory Examination

INDIRECT:

1. Course end survey (Feedback)

Name of the Course Coordinator: Dr. Ramah Sivakumar

CORE II: MOBILE APPLICATION DEVELOPMENT

SEMESTER: I
CREDITS: 4

CODE: P21CS102
HOURS/WEEK: 5

1. COURSE OUTCOMES

On successful completion of this course, students will be able to:

CO #	Course Outcomes	Level	Unit Covered
CO1	Create android project from XML Layout	K3	I
CO2	Debug android app and create UI fragment	K3	II
CO3	Design and develop user interfaces for the Android platform	K4	III
CO4	Design mobile apps with audio play back	K5	IV
CO5	Create database and communicate with mobile application	K6	V
CO6	Apply Java programming concepts to Android application development	K6	V

2. A. SYLLABUS

UNIT I: Fundamentals mobile app

Mobile app basics - Creating an Android project - Navigating in Eclipse - Laying out the user Interface - From Layout XML to view objects - Wiring up widgets - Android build tools - Creating a new class - Adding an Icon - Logging the activity life cycle.

UNIT II: Mobile app manipulation

Debugging Android apps - The DDMS perspective - Android specific debugging - Setting up a second activity - Compatibility and Android programming - The need for UI flexibility - Hosting a UI fragment - Creating a UI fragment - Adding a UI fragment to the fragment manager - The reason all our activities will use fragments.

UNIT III: User Interfaces with layouts and widgets

XML layout attributes - Using the Graphical layout tool - Creating a list fragment - An abstract activity for hosting a fragment - Starting an activity from a fragment - Fragment arguments - Creating crime pager activity - Creating a dialog fragment - Passing data between two fragments - More dialogs.

UNIT IV: Audio playback using media player

Adding resources - Creating hello moon fragment - Audio playback - Rotation and retained fragments - Options menus - Saving and loading data in criminal Intent - Using the camera API - Taking a picture - Using implicit Intents - Styles and Includes.

UNIT V: Mobile app back ground services

Creating an Intent service - Controlling your alarm - Waking up on Boot - Filtering foreground notifications - Setting up the Drag and Draw project - Creating a custom view - Tracking the device's location - Local Databases with SQLite - Adding the maps API to Run tracker - Showing the user's location on a map.

B. TOPICS FOR SELF STUDY

- Fultter Basics Concepts
- Development of Mobile Apps using Flutter)
- Kotlin Programming – Basics
- Kotlin – Mobile Apps Development

C. TEXT BOOK(S):

1. Android Programming "**THE BIG NERD RANCH GUIDE**", BILL PHILLIPS and BRAIN HARDY, Pearson Technology Group, First edition, September 2015, ISBN-13 978-0321804334

D. REFERENCE BOOK(S):

1. Donn Felker and Joshua Dobbs, “*Android Application Development – For Dummies*”, Wiley Publishing Inc., 2011.

E. WEB LINKS

- <https://flutter.dev/docs/reference/tutorials>
- <https://kotlinlang.org/docs/reference/android-overview.html>

3. SPECIFIC LEARNING OUTCOMES

Unit	Course Content	Learning Outcomes	Level
I	Fundamentals of mobile app		
1.1	Mobile app basics	Basic knowledge of mobile technologies	K2
1.2	Creating an Android project	Develop an Android project	K3
1.3	Navigating in Eclipse	Identify location and navigate to a resource	K3
1.4	Laying out the user Interface	Create consistency and use common UI elements	K3
1.5	From Layout XML to view objects	Create View and ViewGroup objects	K4
1.6	Wiring up widgets	Ready to wire up your button widgets	K4
1.7	Android build tools	Identify the component of the Android SDK require for building Android apps	K3
1.8	Creating a new class	Launch Android studio, create a new project	K5
1.9	Adding an Icon	Free icons of Add in various design styles for web, mobile, and graphic design projects	K5
1.10	Logging the activity life cycle	To navigate transitions between stages of the activity lifecycle	K6
II	Mobile app manipulation		
2.1	Debugging Android apps	Select a device to debug your app	K4
2.2	The DDMS perspective	Identify the services provided by DDMS	K4
2.3	Android specific debugging	Examine the concepts of debugging	K5
2.4	Setting up a second activity	Design and develop a second activity	K5
2.5	Compatibility and Android programming	Explain the technical details of the Android platform	K6
2.6	The need for UI flexibility	Create user interfaces which users find easy to use	K3
2.7	Hosting a UI fragment	Fragments contributes a portion of UI to the host activity	K4
2.8	Creating a UI fragment	Design and develop a UI fragment	K3
2.9	Adding a UI fragment to the fragment manager	Static and dynamic way of adding UI fragment to the android activity	K4
2.10	The reason all our activities will use fragments	Reuse a fragments in multiple activities	K6
III	User Interfaces with layouts and widgets		
3.1	XML layout attributes	Identify the properties of XML layout	K3
3.2	Using the Graphical layout tool	Demonstrate the android graphical layout tool	K5
3.3	Creating a list fragment	Design and develop a list fragment	K4
3.4	An abstract activity for hosting a fragment	Identify generic fragment-hosting layout	K5
3.5	Starting an activity from a fragment	Demonstrate MainActivity class with a placeholder fragment	K5

	Fragment arguments	Elaborate the interaction with fragments	K6
3.6	Creating crime pager activity	Develop a crime pager activity	K4
3.7	Creating a dialog fragment	Design a dialog fragment	K4
3.8	Passing data between two fragments	Create a UI related data in the life cycle of activities and fragments.	K5
3.9	More dialogs	Users to take an action before they can proceed	K5
IV	Audio playback using media player		
4.1	Adding resources	Creating and using resource files in Android	K4
4.2	Creating hello moon fragment	Develop a hello moon fragment	K4
4.3	Audio playback	Design a Media player class	K4
4.4	Rotation and retained fragments	Activity with retained fragment is rotated	K5
4.5	Options menus	Primary collection of menu items for an activity	K3
4.6	Saving and loading data in criminal Intent	Find a convenient place to save the data	K4
4.7	Using the camera API	Control the camera hardware directly using the framework APIs	K5
4.8	Taking a picture	The camera app shoots both still images and video.	K4
4.9	Using implicit Intents	Classify all components which are registered for the specific action	K6
4.10	Styles and Includes	A style resource defines the format and look for a UI	K5
V	Mobile app back ground services		
5.1	Creating an Intent service	Develop an Intent service	K4
5.2	Controlling your alarm	Identify the components of control activity	K4
5.3	Waking up on Boot	Start activity on wake up/sleep	K5
5.4	Filtering foreground notifications	Obtain a foreground process without a permanent user notification	
5.5	Setting up the Drag and Draw project	Drag a project up and down to sort your projects	K5
5.6	Creating a custom view	Design and develop a custom view activity	K4
5.7	Tracking the device's location	Identify the device location	K5
5.8	Local Databases with SQLite	To provide local data storage for individual applications and devices	K6
5.9	Adding the maps API to Run tracker	Classify the various API for run tracker	K6
5.10	Showing the user's location on a map	Use the location component to show the Users current location on the map	K5

5. MAPPING SCHEME (CO, PO & PSO)

P21CS102	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PSO1	PSO2	PSO3	PSO4
CO1	H	H	H	H	H		M		H	H	H	H	
CO2	H	H	H	H	M		M		H	H	M	M	M
CO3	H	H	H	M	M		M		H	M	M	H	M
CO4	H	H	H	M	M		M		H	H	M	H	M
CO5	H	H	M	H	M		M		H	L	M	H	M
CO6	H	M	H	H	H	M	H	M	H	H	M	M	M

5. COURSE ASSESSMENT METHODS

DIRECT:

1. Continuous Assessment Test: T1, T2 (Theory & Practical Components): Closed Book
2. Open Book Test.
3. Cooperative Learning Report, Assignment, Group Presentation, Group Discussion, project Report, Field Visit Report, Poster Presentation, Seminar, Quiz (written).
4. Pre-Semester & End Semester Theory Examination

INDIRECT:

1. Course end survey (Feedback)

Name of the Course Coordinator: Dr. B. Karthikeyan

CORE III: MATHEMATICAL FOUNDATION FOR COMPUTER SCIENCE**SEMESTER: I**
CREDITS: 4**CODE: P21CS103**
HOURS/WEEK: 5

1. COURSE OUTCOMES

On successful completion of this course, students will be able to:

CO #	Course Outcomes	Level	Unit Covered
CO1	Examine and Evaluate the Logical propositions and predicates	K4	I
CO2	Solve the counting problems using Combinatorics and Probability	K5	II
CO3	Demonstrate the concepts of Sets and Relations for solving the complex problems	K4	III
CO4	Apply the knowledge on Graphs and Trees to real world applications.	K5	IV
CO5	Design the Languages using Grammars.	K5	V
CO6	Construct the different types of Machine using Languages and Grammars for computability.	K6	V

2. A. SYLLABUS**Unit I : The foundations: Logic and Proofs**

Propositions – Conditional statements – Converse contrapositive and inverse – Bi Conditionals and Implicit Bi conditionals – Truth tables – Operators – Translating English statements – system specification – Boolean searches and Logic puzzles – Logical equivalence and De Morgans law – Constructing new logical equivalence – Propositional satisfiability – Predicates and Quantifiers – Nested Quantifiers – Valid arguments in propositional logic – Rules of inference – Using rules of inference to build arguments – Rules of inference for quantified statements.

Unit II : Sets, Functions and Basic Counting

Sets – Set Operations- Functions – Recursive functions - The basics of Counting – The Pigeonhole Principle – Permutations – Combinations – Generalized Permutation and Combination – Generating Permutation and Combination.

Unit III: Relations

Relations and their properties – n-ary Relations and their applications – Representing relations using Matrices – Representing relations using digraphs – Closures – Paths in directed graphs and transitive closures – Warshall's Algorithm – Partial Orderings

Unit IV : Graphs and Trees

Graphs and Graph models – Graph terminology and special types of graphs – Representing Graphs and Graph Isomorphism – Connectivity – Euler and Hamilton paths – Shortest path problems – Planar Graphs – Introduction and applications of Trees – Tree Traversal – Spanning Trees and Minimum Spanning Trees.

Unit V : Modelling Computation

Languages and Grammars – Types of Grammars and Definitions – Derivation trees – Bacus Normal Form(BNF) – Finite state machines with output – Finite state machines with No output – Constructing NFA – Constructing DFA – Definition of Turing Machine and Turing Machine to recognize sets – Computing functions with Turing Machine and types.

B. TOPICS FOR SELF STUDY

- Catalan Number
- Equivalence Relations
- Functions
- Advanced Counting

C. TEXT BOOK(S)

1. Kenneth H. Rosen, “ Discrete Mathematics and Its Applications”, The Mc Graw- Hill companies, 7th edition, 2012.

D. REFERENCES BOOK(S)

1. Mahima Ranjan Adhikari and Avishek Adhikari, “*Basic Modern Algebra with Applications*”, Springer 2014.
2. Kolman, Busby and Ross, “*Discrete Mathematical Structures*”, 6th edition, PHI. 2009.

E. Web Links

- <https://nptel.ac.in/courses/106/106/106106183/>
- http://www.cs.ucr.edu/~acald013/public/tmp/sol_dmaia_rosen.pdf
- <https://www.archerimagine.com/articles/100daysofdiscretemath/100Days-of-DiscreteMath-log-file.html>
- <https://freevideolectures.com/course/3517/discrete-mathematics-i>

3. SPECIFIC LEARNING OUTCOMES

Unit	Course Contents	Learning Outcomes	Levels
I	THE FOUNDATION: LOGIC AND PROOFS		
1.1	Propositions	<ol style="list-style-type: none">1. Define a Proposition.2. Explain Propositions.3. Construct Propositions for the statements.4. Analyze whether the proposition is a tautology or not.5. Evaluate whether the given statement is a proposition or not.	K5
1.2	Operators	<ol style="list-style-type: none">1. Recall different types of operators.2. Explain operators using truth table.3. Construct truth table using operators.	K3
1.3	Logical Equivalence and De Morgan’s Law	<ol style="list-style-type: none">1. Define and apply De Morgan’s Law.2. Evaluate and analyse whether the given statements are equivalent or not.3. Explain De Morgan’s Law.	K5
1.4	Predicates and Quantifiers	<ol style="list-style-type: none">1. Define Predicates and Quantifiers.	K1
1.5	Rules of Inference	<ol style="list-style-type: none">1. List the rules of Inference.2. Illustrate and Identify whether the given statement is valid or not.	K3
II	SETS, FUNCTIONS AND COUNTING		
2.1	Sets – Set Operations	<ol style="list-style-type: none">1. Define and apply sets and set operations.	K6
2.2	Functions	<ol style="list-style-type: none">1. Apply the functions for different applications2. Explain the uses of Functions	K5 K5
2.3	The basics of Countings	<ol style="list-style-type: none">1. Define, Demonstrate and Apply sum rule and product rule.	K3
2.4	Permutations	<ol style="list-style-type: none">1. Define and Demonstrate Permutations.2. Apply Permutation rules.	K3
2.5	Combinations	<ol style="list-style-type: none">1. Define and Demonstrate Combinations.2. Apply Combination rules	K3
III	RELATIONS		
3.1	Relations and their properties	<ol style="list-style-type: none">1. Define Relation.2. Explain the properties of Relation.3. Apply composite of Relation.	K3

3.2	Representing Relations	1. Define Relations. 2. Explain the types of Representation of Relation and solve problems.	K3
3.3	Closures-paths in directed graphs and transitive closures	1. Define, Explain and Apply Closure properties of Relation.	K3
3.4	Warshall's Algorithm	1. Explain Warshall's Algorithm.	K5
3.5	Partial Orderings	1. Define and Explain Equivalence Relation and Equivalence Classes.	K5
IV	GRAPHS & TREES		
4.1	Graphs- Graph terminology and special types of graphs	1. Define Graphs. 2. Explain various types of Graphs. 3. Solve problems.	K3
4.2	Representing Graphs and Graph Isomorphism	1. Define Graphs. 2. Explain how to construct a Graph. 3. Illustrate Graph Representations.	K3
4.3	Connectivity-Euler and Hamilton paths-shortest path problems	1. Define and Explain Euler and Hamilton paths.	K2
4.4	Planar Graphs	1. Define, Explain Planar Graphs. 2. Apply planarity rule. 3. Construct a Planar graph. 4. Prove whether the given Graph is Planar or not.	K5
4.5	Introduction and Applications of Trees- Tree Traversal – Spanning Trees and Minimum Spanning Trees	1. Define Trees. 2. Explain the properties of Trees. 3. Construct Trees.	K3
V	MODELLING COMPUTATION		
5.1	Languages and Grammars	1. Define Grammars and Languages. 2. Explain and Compare various types of Grammars. 3. Evaluate and find which type of grammar. 4. Construct Grammars for the given Languages.	K6
5.2	Derivation Trees	1. What is Derivation tree? 2. Explain the types of Derivation Trees. 3. Apply the properties. 4. Construct Derivation Trees. 5. Discuss on Trees.	K6
5.3	Bacus Normal Form	1. Recall Bacus Normal Form. 2. Explain BNF	K2
5.4	Finite State Machines	1. What is Finite State Machine? 2. Explain Finite State Automata? 3. Construct NFA and DFA.	K3
5.5	Turing Machine-Turing Machine to recognize sets – Computing functions with Turing Machine and types	1. Define Turing Machine. 2. Explain Turing Machine.	K2

4. MAPPING SCHEME (CO, PO & PSO)

P21CS103	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PSO1	PSO2	PSO3	PSO4
CO1	H	H	H	H	H		H		H	H	H	H	
CO2	H	H	H	H	M		H		H	H	M	H	M
CO3	H	H	H	H	M		M		H	H	M	H	M
CO4	H	H	H	H	H		H		H	H	M	H	M
CO5	H	H	H	H	M		M		H	H	M	H	M
CO6	H	H	H	H	H	M	H	M	H	H	M	H	M

5. COURSE ASSESSMENT METHODS

DIRECT

1. Continuous Assessment Test: T1, T2 (Theory & Practical Components): Closed Book
2. Open Book Test.
3. Cooperative Learning Report, Assignment, Group Presentation, Group Discussion, project Report, Field Visit Report, Poster Presentation, Seminar, Quiz (written).
4. Pre-Semester & End Semester Theory Examination

INDIRECT

1. Course end survey (Feedback)

Name of the Course Coordinator: Dr. J. Persis Jessintha

ELECTIVE 1A: MANAGING SOFTWARE DEVELOPMENT

SEMESTER: I
CREDITS: 4

CODE: P21CS1:1
HOURS/WEEK: 5

1. COURSE OUTCOMES

On successful completion of this course, students will be able to:

CO #	Course Outcomes	Level	Unit Covered
CO1	Apply the process to be followed in the software development life-cycle models.	K3	I
CO2	Use Agile process models for collaborative decision making.	K3	II
CO3	Analyze & design the software models using unified modeling language (UML).	K4	II
CO4	Implement quality management techniques & different types of metrics in software development.	K5	III
CO5	Schedule projects and manage risks using risk management strategies.	K5	IV
CO6	Apply project management concepts and techniques to an IT project.	K6	V

2. A. SYLLABUS

Unit 1: Software, Software Engineering & The Software Process

Introduction to Software: The nature of Software – The changing nature of Software. **Software Engineering:** Defining the discipline – The software process – Software Engineering practice – Software development myths. **Agile development:** What is Agility? – Agility and the cost of change – What is an Agile process? – Extreme programming – Other Agile process models.

Unit 2: Modeling

Understanding requirements: Requirements Engineering – Establishing the groundwork – Eliciting requirements – Developing Use Cases – Building the analysis model – Negotiating requirements. **Requirements modelling for Scenario-based methods:** Requirements analysis – Scenario-based modelling – UML models that support the Use Case.

Unit 3: Quality Management

Quality concepts: What is Quality? – Software Quality – The software quality dilemma – Achieving software quality. **Review techniques:** Cost impact of software defects – Review metrics and their use – Reviews: A formality spectrum – Informal reviews – Formal technical reviews – Post-Mortem evaluations. **Software Quality Assurance:** Elements of Software Quality Assurance – SQA Tasks, Goals and Metrics – Statistical Software Quality Assurance – Software reliability – The ISO 9000 Quality standards – The SQA plan.

Unit 4: Managing Software Projects – I

Project Management concepts: The Management spectrum – People – The Product – The Process – The Project – The W⁵HH principle. **Process and Project Metrics:** Metrics in the process and project domains – Software measurement – Metrics for software quality – Metrics for small organizations **Estimation for software projects:** Observations on estimation – The Project planning process – Software scope and feasibility – Resources – Software Project Estimation – Decomposition techniques – The Make/Buy decision.

Unit 5: Managing Software Projects – II

Project Scheduling: Basic concepts – Project scheduling – Defining a task set for the software project – Defining a task network – Scheduling – Earned Value Analysis. **Risk Management:** Reactive versus Proactive Risk Strategies – Software Risks – Risk Identification – Risk Projection – Risk Refinement – Risk Mitigation, Monitoring and Management – The RMMM plan.

B. TOPIC FOR SELF STUDY

- Selenium
- Selenium Web Driver
- Manual testing
- Automated Testing using Selenium
(Web Reference: <https://freevideolectures.com/course/3625/testing-with-selenium>)

C. TEXT BOOK(S)

1. Roger S. Pressman, Bruce R. Maxim, *“Software Engineering, A practitioner’s Approach”*, Eighth edition, McGraw Hill, 2020.

D. REFERENCE BOOK(S)

1. Tsui and Karam (T&K), *“Essentials of Software Engineering”*, Third Edition, Jones and Barrlet Publishing, 2013.

E. WEB LINKS

- https://www.tutorialspoint.com/software_engineering/index.htm
- <https://builtin.com/software-engineering-perspectives/how-to-become-a-software-engineer>
- <https://www.geeksforgeeks.org/software-engineering/>
- <https://www.udemy.com/courses/development/software-engineering>

3. SPECIFIC LEARNING OUTCOMES

Unit	Course Content	Learning Outcomes	Level
	By the end of each topic of a unit, students will be able to		
I	Software, Software Engineering & The Software Process		
1.1	The nature of Software, The changing nature of Software	Understand the nature of Software	K2
1.2	Defining the discipline	Define software engineering	K3
1.3	The software process	Explain the software development process	K5
1.4	Software Engineering practice –	Apply engineering practices to software development, and typically handle the overall system design of the software application	K3
1.5	Software development myths.	Challenge Software development myths in the name of improving business efficiency.	K4
1.6	What is Agility?	Define agility	K2
1.7	Agility and the cost of change	Examine the Agile Cost of Change Curve	K4
1.8	What is an Agile process?	Use agile process to reduce the cost of change	K3
1.9	Extreme programming	Explain extreme programming framework to produce higher quality software.	K3
1.10	Other Agile process models	Adopt Agile process models like Adaptive Software Development, Dynamic Systems Development Method, Scrum, and Crystal	K4
II	Modeling		
2.1	Requirements Engineering - Establishing the groundwork, Eliciting requirements	Perform groundwork with what the customer desires, analyse the need, and assess feasibility, negotiate a reasonable solution, specify the solution clearly, validate the specifications and manage the requirements.	K5
2.2	Developing Use Cases.	Develop use cases during system analysis to identify, clarify, and categorize system requirements.	K4

2.3	Building the analysis model	Build the analysis model to define information, behaviour and functions of the system.	K4
2.4	Negotiating requirements	Manage the stakeholders of the software project who disagrees on the requirements, by using requirements negotiation methods to avoid rework and extra costs in the software development.	K5
2.5	Requirements analysis	Focus on the tasks that determine the needs or conditions to meet the project, taking account of the possibly conflicting requirements of the various stakeholders, analyze, document, validate and manage software requirements.	K5
2.6	Scenario-based modelling	Identify the possible use cases for the system and produce the use case diagrams, to which all the other stages of requirements modeling refer.	K4
2.7	UML models that support the Use Case	Visualize the design of the system by UML diagrams.	K4
III	Quality Management		
3.1	What is Quality?	Define quality in terms of software.	K2
3.2	Software Quality.	Measure how well software is designed (quality of design), and how well the software conforms to that design (quality of conformance)	K5
3.3	The software quality dilemma	Understand software quality dilemma	K2
3.4	Achieving software quality	Implement success factors such as Software Engineering Methods, Project Management Techniques, Quality Control, Quality Assurance to achieve software quality.	K5
	Cost impact of software defects	Implement the elements of software quality assurance to avoid cost impact of software defects.	K4
3.5	Review metrics and their use	Make a technical assessment of the work product created during the software engineering process.	K5
3.6	Reviews: A formality spectrum	Conduct a meeting by technical people for technical people for software quality assurance.	K5
3.7	Informal reviews, Formal technical reviews	Examine the work product by Informal reviews, and Formal technical reviews.	K4
3.8	Post-Mortem evaluations	Perform post-mortem evaluations at the conclusion of a project, to determine and analyse elements of the project that were successful or unsuccessful.	K5
3.9	Software Quality Assurance: Elements of Software Quality Assurance	Explain the elements of Software Quality Assurance	K2
3.10	SQA Tasks, Goals and Metrics	Undertake SQA Tasks, Goals and Metrics to achieve software quality.	K3
3.11	Statistical Software Quality Assurance	Collect information on all defects, find the causes of the defects, Move to provide fixes for the process.	K4
3.12	Software reliability	Calculate mean-time-to-failure, mean-time-to-repair, mean-time-between-failure and Software availability to assure reliability.	K4

3.13	The ISO 9000 Quality standards	Familiar with The ISO 9000 Quality standards.	K2
3.15	The SQA plan	Create SQA plan for the developing software.	K6
IV	Managing Software Projects – I		
4.1	Project Management concepts	Explain Project Management concepts such as planning of project, deciding scope of software product, estimation of cost in various terms, scheduling of tasks and events, and resource management	K5
4.2	The Management spectrum	Describe the management of a software project and how to make a project successful	K2
4.3	People	Apply People Management Capability Maturity Model (PM-CMM), to undertake increasingly complex applications by helping to attract, grow, motivate, deploy, and retain the talent needed to improve the software development capability.	K3
4.4	The Product	Identify the product objectives and scope that should be established, alternative solutions that should be considered, and also technical and management constraints.	K4
4.5	The Process	Create the framework from which a comprehensive plan for software development can be established.	K6
4.6	The Project	Undertake the managerial issues which include all and everything of the total development process and take steps to avoid project failure.	K3
4.7	The W ⁵ HH principle	Raise series of questions related to project such as why, what, where, when, who, how and how much, to help project managers more efficiently manage software projects.	K5
4.8	Process and Project Metrics: Metrics in the process and project domains	Recognize Metrics in the process and project domains	K3
4.9	Software measurement	Categorize the Software measurement metrics.	K3
4.10	Metrics for software quality	Explain the metrics for software quality and identify its categories.	K2
4.11	Metrics for small organizations	Choose simple metrics that provide value to the organization and don't require a lot of effort to collect for small organizations.	K5
4.12	Estimation for software projects: Observations on estimation	Explain the four basic steps in Software Project Estimation.	K2
4.12	The Project planning process	Produce a schedule, identify and assess software risks, and negotiate commitments.	K4
4.14	Software scope and feasibility	Find out the actual operations that are going to be carried out by the software and its plus points and limitations and estimate resources needed.	K4
4.15	Resources	Identify the Project resources which include people, capital, and/or material goods required for the successful execution and completion of a project.	K4

4.16	Software Project Estimation.	Estimate of the size of software and predict the effort and time which will be needed to build the project	K5
4.17	Decomposition techniques	Compute Size, Effort and Cost estimation in a stepwise manner by breaking down a Project into major Functions or related Software Engineering Activities.	K5
4.18	The Make/Buy decision	Make make-or-buy decisions based on the important values such as the volume, the fixed cost of making, per-unit direct cost when making and per-unit cost when buying.	K4
V	Managing Software Projects – II		
5.1	Project Scheduling: Basic concepts	Explain the basic concepts of project scheduling.	K2
5.2	Project scheduling	Schedule the Project with the mechanism to communicate what tasks need to get done and which organizational resources will be allocated to complete those tasks in what timeframe and so on.	K6
5.3	Defining a task set for the software project	Create a Task Set for the Software Project which includes collection of software engineering work tasks, milestones, and deliverables that must be accomplished to complete a particular project.	K6
5.4	Defining a task network	Create a graphic representation of the task flow for a project that depicts major software engineering tasks.	K6
5.5	Scheduling	Schedule the project that includes the planned start and finish date, duration, and resources assigned to each activity.	K6
5.6	Earned Value Analysis	Compute Earned Value Analysis (EVA) to measure the project's progress at any given point in time, forecast its completion date and final cost, and analyse variances in the schedule and budget as the project proceeds.	K5
5.7	Risk Management: Reactive versus Proactive Risk Strategies.	Discriminate risk strategies.	K3
5.8	Software	Define software	K2
5.9	Risks – Risk Identification	Identify the potential risks that could prevent the program, enterprise, or investment from achieving its objectives.	K4
5.10	Risk Projection	Estimate the impact of the risk on the project and the product.	K4
5.11	Risk Refinement	Conduct the process of restating the risks as a set of more detailed risks that will be easier to mitigate, monitor, and manage	K5
5.12	Risk Mitigation,	Accomplish effectiveness throughout the project by Risk mitigation planning, Risk mitigation implementation and Risk progress monitoring.	K5
5.13	Monitoring	Monitor to track, review, and regulate the progress and performance of the project.	K4
5.14	Management	Ensure that software products and software engineering services are delivered efficiently, effectively, and to the benefit of stakeholders	K4

5.15	The RMMM plan	Analyse the risk which is related to the project and prepare RMMM plan.	K4
------	---------------	---	----

4. MAPPING SCHEME (CO, PO & PSO)

P21CS1:1	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PSO1	PSO2	PSO3	PSO4
CO1	H		L		L								
CO2	H	M	M	M	H	H							
CO3	H	H	H	H	M	L	M		H	M	H	L	
CO4	H	H	M	L	H			H	M	H	M	M	H
CO5	H			H	H		H	H	H	H	H	M	H
CO6	H		H	H	H		H	H	H	H	H	H	H

5. COURSE ASSESSMENT METHODS

DIRECT

1. Continuous Assessment Test: T1, T2 (Theory & Practical Components): Closed Book
2. Open Book Test.
3. Cooperative Learning Report, Assignment, Group Presentation, Group Discussion, project Report, Field Visit Report, Poster Presentation, Seminar, Quiz (written).
4. Pre-Semester & End Semester Theory Examination

INDIRECT

1. Course end survey (Feedback)

Name of the Course Coordinator: Prof. C. Sathish Kumar

CORE PRACTICAL I: DATA ANALYTICS LAB

SEMESTER: I
CREDITS: 3

CODE: P21CSP1
HOURS/WEEK: 5

1. COURSE OUTCOMES

On successful completion of this course, students will be able to:

CO #	Course Outcomes	Level
CO1	Experiment with the basic Python commands	K3
CO2	Develop array operations using NumPy and pandas data structures	K3
CO3	Discover data loading operations in Python	K4
CO4	Evaluate data wrangling methods in Python	K5
CO5	Construct Plotting, Visualization, Data Aggregation and Group operations	K6
CO6	Create applications in Time Series and Financial analysis	K6

2. SYLLABUS

- 1a). Practice using the Python interpreter / Jupyter Notebook as a calculator:
- The volume of a sphere with radius r is $\frac{4}{3}\pi r^3$. What is the volume of a sphere with radius 5?
 - Suppose cover price of a book is INR 24.95, but a bookstore offers a 40% discount. Shipping costs INR 3.00 for the first copy and 75 Paise for each additional copy. What is the total wholesale cost for 60 copies?
 - If I leave my house at 6:52 am and run 1 km at an easy pace (8:15 per km), then 3 km at tempo (7:12 per km) and 1 km at easy pace again, what time do I get home for breakfast?
- 1b). Develop an application in Python that repeatedly reads numbers until the user enters “done”. Once “done” is entered, print out the total, count, and average of the numbers. If the user enters anything other than a number, detect their mistake using try and except and print an error message and skip to the next number.
- 2a). Develop a Python function `front_x()`. Given a list of strings, return a list with the strings in sorted order, except group all the strings that begin with 'x' first. Eg. ['mix', 'xyz', 'apple', 'xanadu', 'aardvark'] yields ['xanadu', 'xyz', 'aardvark', 'apple', 'mix']. Hint: this can be done by making 2 lists and sorting each of them before combining them.
- 2b). Develop a Python function `sort_last()`. Given a list of non-empty tuples, return a list sorted in increasing order by the last element in each tuple. E.g. [(1, 7), (1, 3), (3, 4, 5), (2, 2)] yields [(2, 2), (1, 3), (3, 4, 5), (1, 7)]. Hint: use a custom `key=` function to extract the last element from each tuple.
- 2c). Develop a Python function `remove_adjacent()`. Given a list of numbers, return a list where all adjacent same elements have been reduced to a single element, so [1, 2, 2, 3] returns [1, 2, 3]. List [2, 2, 3, 3, 3] returns [2, 3]. List [] returns []. You may create a new list or modify the passed in list.
- 3a). Develop an application in Python to open the file (say, romeo.txt) and read it line by line. For each line, split the line into words using the `split` function. For each word, check to see if the word is already in a list. If the word is not in the list, add it to the list. When the program completes, sort and print the resulting words in alphabetical order.
- 3b). Develop an application in Python to read through the email data and when you find line that starts with “From”, you will split the line into words using the `split` function. We are interested in who sent the message, which is the second word on the From line: From stephen.marquard@uct.ac.za Sat Jan 5 09:14:16 2008. You will parse the From line and print out the second word for each From line, then you will also count the number of From (not From:) lines and print out a count at the end.
- 4a). [NumPy]: Read a CSV file with 10 numbers in each row (5 rows), and store them in numpy array, print the values and maximum and minimum value of the array. Also save the same array in another file in TSV format

- 4b). [NumPy]: Create a small dataset of floating point numbers with 10 rows and 5 columns in ARFF format with proper header values. Read the files and store the numbers in numpy array. Reshape the array to 5 rows and 10 columns using numpy. Print the following properties of the array for both before and after reshaping. Print number of dimensions, shape of the array, size of the array, data type of the array and item size of the array.
- 5a). [NumPy]: Create a 3*3 matrix A [1 1 1, 2 2 2, 3 3 3] and B [4 4 4, 5 5 5, 6 6 6]. Print the following values. 1. Add A and B, 2. Subtract A from B, 3. Element wise multiplication A and B, 4. Divide B by A. 5. Square root of B, 6. Square of A.
- 5b). [NumPy]: Create a 4*4 matrix A [1 1 1 1, 2 2 2 2, 3 3 3 3, 4 4 4 4] in numpy. Print the following matrix operations using built-in numpy functions. 1. Determinant of the matrix, 2. Inverse of a matrix, 3. Transpose of the matrix A, 4. Scalar multiplication with number, 5. Matrix multiplication with matrix B [2 2 2 2], 6. Dot product between rows of A and matrix B [2 2 2 2]. 7. Determinant of the matrix A.
- 6a). [NumPy]: Create a 3*3 matrix A [4 5 6, 1 2 3, 7 8 9]. Flat the matrix into an array and print the sorted array in ascending and descending order using numpy functions. Print the mean, standard deviation and variance of the sorted array.
- 6b). [Pandas]: Create an array of 10 records of cities and their population and write them to a CSV file using pandas. Read the same CSV and store it in pandas dataframe. Print the values inside the dataframe and print information about the pandas dataframe variable using pandas builtin function. Print the first 3 and last 4 records using pandas functions. Print the indices of the cities with maximum and minimum population.
- 7a). [Pandas]: Create an array of 10 records of people names and their ages and write them to a TXT file using pandas. Read the same TXT file and store it in pandas variable. Give the column names as "Name" and "Age" for pandas variable. Print the description of the Name and Age using pandas functions. Sort and print the records in ascending order based on "Name". Sort and print the records in descending order based on "Age".
- 7b). [Pandas]: Read the data using pandas. Transpose the dataframe and save it to another file in JSON format. Read the JSON file and print the transposed dataframe values.
- 8a). [Pandas]: Read the text "You cannot end a sentence with because because because is a conjunction" from a text file. Find the unique words and their frequency from the text and store it in pandas variable with 2 columns "Words" and "Frequency". Do the following using pandas builtin functions, 1). Number of rows, 2). Sum of "Frequency", 3). Mean "Frequency", 4). standard deviation of "Frequency", 5). Upper case of "Words", 6). Lower case of "Words".
- 8b). [Pandas]: Generate a dataframe in pandas with two columns "Name", "Mark 1". Insert 5 records to the dataframe. Print the dataframe values. Insert one more column "Mark 2" with randomly generated integer numbers between range 50 to 100. Print the dataframe values after insertion. Compute the total of Mark 1 and Mark 2 and insert to dataframe as "Total" and print the dataframe. Compute average from "Total" and insert it as "Avg" and remove "Total" from dataframe. Print the dataframe values after deletion.
- 9a). [Matplotlib]: Create array of 20 numbers representing sin function using numpy, similarly arrays for cos function and tan function. Plot the sin and cos functions as green and red colored lines using matplotlib. Give the labels to the lines and the axes in the plot.
- 9b). [Matplotlib]: Plot a histogram plot with bars in red color. Present the the mean and standard deviation of the values in the graph.
- 10a). [Matplotlib]: Populate Pandas Dataframe with two columns "Date" and "Temperature". Generate dates 01.01.2017 to 31.01.2017 and generate random temperature from 20 to 40 degrees for each day. Plot the data using matplotlib and mark the maximum and minimum temperature of the week in the line graph. Give proper data labels and legend titles.
- 10b). [Matplotlib]: Create a dataset of 20 records with 4 columns for persons' "Name", "Age", "Height", "Weight". The height and weight can be linearly increasing as the age increases. Plot the dataset in Scatterplot using matplotlib with x axis for "age", y-axis for "height" and the size of the datapoints representing the "weight" of each person. Give proper labels to the axes in the plot.
- 10c). [Matplotlib]: Generate a dataset of 5 rows representing students, each row with 10 columns representing marks in 10 exams. Compute the standard deviation of each student's marks and plot

the individual students in box plot using matplotlib representing the range of marks and standard deviation.

3. SPECIFIC LEARNING OUTCOMES

S.No.	Exercise List	Level
1a)	Practice using the Python interpreter / Jupyter Notebook as a calculator:	
	i). The volume of a sphere with radius r is $\frac{4}{3}\pi r^3$. What is the volume of a sphere with radius 5?	K1
	ii). Suppose cover price of a book is INR 24.95, but a bookstore offers a 40% discount. Shipping costs INR 3.00 for the first copy and 75 Paise for each additional copy. What is the total wholesale cost for 60 copies?	K1
	iii). If I leave my house at 6:52 am and run 1 km at an easy pace (8:15 per km), then 3 km at tempo (7:12 per km) and 1 km at easy pace again, what time do I get home for breakfast?	K1
1b)	Develop an application in Python that repeatedly reads numbers until the user enters “done”. Once “done” is entered, print out the total, count, and average of the numbers. If the user enters anything other than a number, detect their mistake using try and except and print an error message and skip to the next number	K3
2a)	Develop a Python function front_x(). Given a list of strings, return a list with the strings in sorted order, except group all the strings that begin with 'x' first. Eg. ['mix', 'xyz', 'apple', 'xanadu', 'aardvark'] yields ['xanadu', 'xyz', 'aardvark', 'apple', 'mix']. Hint: this can be done by making 2 lists and sorting each of them before combining them	K3
2b)	Develop a Python function sort_last(). Given a list of non-empty tuples, return a list sorted in increasing order by the last element in each tuple. E.g. [(1, 7), (1, 3), (3, 4, 5), (2, 2)] yields [(2, 2), (1, 3), (3, 4, 5), (1, 7)]. Hint: use a custom key= function to extract the last element form each tuple	K3
2c)	Develop a Python function remove_adjacent(). Given a list of numbers, return a list where all adjacent same elements have been reduced to a single element, so [1, 2, 2, 3] returns [1, 2, 3]. List [2, 2, 3, 3, 3] returns [2, 3]. List [] returns []. You may create a new list or modify the passed in list	K3
3a)	Develop an application in Python to open the file (say, romeo.txt) and read it line by line. For each line, split the line into words using the split function. For each word, check to see if the word is already in a list. If the word is not in the list, add it to the list. When the program completes, sort and print the resulting words in alphabetical order	K6
3b)	Develop an application in Python to read through the email data and when you find line that starts with “From”, you will split the line into words using the split function. We are interested in who sent the message, which is the second word on the From line: From stephen.marquard@uct.ac.za Sat Jan 5 09:14:16 2008. You will parse the From line and print out the second word for each From line, then you will also count the number of From (not From:) lines and print out a count at the end	K6
4a)	[NumPy]: Read a CSV file with 10 numbers in each row (5 rows), and store them in numpy array, print the values and maximum and minimum value of the array. Also save the same array in another file in TSV format	K2
4b)	[NumPy]: Create a small dataset of floating point numbers with 10 rows and 5 columns in ARFF format with proper header values. Read the files and store the numbers in numpy array. Reshape the array to 5 rows and 10 columns using numpy. Print the following properties of the array for both before and after reshaping. Print number of dimensions, shape of the array, size of the array, data type of the array and item size of the array	K6
5a)	[NumPy]: Create a 3*3 matrix A [1 1 1, 2 2 2, 3 3 3] and B [4 4 4, 5 5 5, 6 6 6] Print the following values. 1. Add A and B, 2. Subtract A from B, 3.	K3

	Element wise multiplication A and B, 4. Divide B by A, 5. Square root of B, 6. Square of A	
5b)	[NumPy]: Create a 4*4 matrix A [1 1 1 1, 2 2 2 2, 3 3 3 3, 4 4 4 4] in numpy. Print the following matrix operations using built-in numpy functions. 1. Determinant of the matrix, 2. Inverse of a matrix, 3. Transpose of the matrix A, 4. Scalar multiplication with number, 5. Matrix multiplication with matrix B [2 2 2 2], 6. Dot product between rows of A and matrix B [2 2 2 2]. 7. Determinant of the matrix A	K3
6a)	[NumPy]: Create a 3*3 matrix A [4 5 6, 1 2 3, 7 8 9]. Flat the matrix into an array and print the sorted array in ascending and descending order using numpy functions. Print the mean, standard deviation and variance of the sorted array	K3
6b)	[Pandas]: Create an array of 10 records of cities and their population and write them to a CSV file using pandas. Read the same CSV and store it in pandas dataframe. Print the values inside the dataframe and print information about the pandas dataframe variable using pandas builtin function. Print the first 3 and last 4 records using pandas functions. Print the indices of the cities with maximum and minimum population	K6
7a)	[Pandas]: Create an array of 10 records of people names and their ages and write them to a TXT file using pandas. Read the same TXT file and store it in pandas variable. Give the column names as "Name" and "Age" for pandas variable. Print the description of the Name and Age using pandas functions. Sort and print the records in ascending order based on "Name". Sort and print the records in descending order based on "Age"	K6
7b)	[Pandas]: Read the data using pandas. Transpose the dataframe and save it to another file in JSON format. Read the JSON file and print the transposed dataframe values	K2
8a)	[Pandas]: Read the text "You cannot end a sentence with because because because is a conjunction" from a text file. Find the unique words and their frequency from the text and store it in pandas variable with 2 columns "Words" and "Frequency". Do the following using pandas built-in functions, 1). Number of rows, 2). Sum of "Frequency", 3). Mean "Frequency", 4). standard deviation of "Frequency", 5). Upper case of "Words", 6). Lower case of "Words"	K4
8b)	[Pandas]: Generate a dataframe in pandas with two columns "Name", "Mark 1". Insert 5 records to the dataframe. Print the dataframe values. Insert one more column "Mark 2" with randomly generated integer numbers between range 50 to 100. Print the dataframe values after insertion. Compute the total of Mark 1 and Mark 2 and insert to dataframe as "Total" and print the dataframe. Compute average from "Total" and insert it as "Avg" and remove "Total" from dataframe. Print the dataframe values after deletion	K6
9a)	[Matplotlib]: Create array of 20 numbers representing sin function using numpy, similarly arrays for cos function and tan function. Plot the sin and cos functions as green and red colored lines using matplotlib. Give the labels to the lines and the axes in the plot	K6
9b)	[Matplotlib]: Plot a histogram plot with bars in red color. Present the the mean and standard deviation of the values in the graph	K3
10a)	[Matplotlib]: Populate Pandas Dataframe with two columns "Date" and "Temperature". Generate dates 01.01.2017 to 31.01.2017 and generate random temperature from 20 to 40 degrees for each day. Plot the data using matplotlib and mark the maximum and minimum temperature of the week in the line graph. Give proper data labels and legend titles	K3, K6
10b)	[Matplotlib]: Create a dataset of 20 records with 4 columns for persons' "Name", "Age", "Height", "Weight". The height and weight can be linearly increasing as the age increases. Plot the dataset in Scatterplot using matplotlib with x axis for "age", y-axis for "height" and the size of the datapoints representing the "weight" of each person. Give proper labels to the axes in the plo	K3, K6
10c)	[Matplotlib]: Generate a dataset of 5 rows representing students, each row with 10 columns representing marks in 10 exams. Compute the standard deviation of each	K4, K6

	student's marks and plot the individual students in box plot using matplotlib representing the range of marks and standard deviation	
--	--	--

4. MAPPING SCHEME (CO, PO & PSO)

P21CSP1	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PSO1	PSO2	PSO3	PSO4
CO1	H	M	L		L								
CO2	H	M	M	M	H	H							
CO3	H	H	H	H	M	L	M		H	M	H	L	
CO4	H	H	M	L	H	H		H	M	H	M	M	H
CO5	H	H		H	H		H	H	H	H	H	M	H
CO6	H		H	H	H		H	H	H	H	H	H	H

5. COURSE ASSESSMENT METHODS

DIRECT

1. Continuous Assessment Test: T1, T2 (Theory & Practical Components): Closed Book
2. Open Book Test.
3. Cooperative Learning Report, Assignment, Group Presentation, Group Discussion, project Report, Field Visit Report, Poster Presentation, Seminar, Quiz (written).
4. Pre-Semester & End Semester Theory Examination

INDIRECT

1. Course end survey (Feedback)

Name of the Course Coordinator: Dr. Ramah Sivakumar

MOBILE APPLICATION DEVELOPMENT LAB**SEMESTER: I**
CREDITS : 3**CODE: P21CS1P2**
HOURS/WEEK : 5

1. COURSE OUTCOMES

On successful completion of this course, students will be able to:

CO #	Course Outcomes	Level	Ex. No.
CO1	Design the Mobile app for general purposes with preliminary concepts	K6	1
CO2	Design the Mobile App with Database connectivity	K6	2-6
CO3	Develop the app with Graphical Primitives	K5	7
CO4	Develop the app with multithread concepts	K5	8
CO5	Create the app GUI components	K6	9-10
CO6	Create the Mobile apps with Background designs	K6	9-10

2. SYLLABUS

1. Develop a native calculator mobile app for arithmetic operations
2. Develop an application that makes use of student database for end semester mark list
3. Design a mobile app for hotel menu card using list view activity
4. Develop a mobile app for expense tracker monitoring system of monthly budget
5. Implement a mobile app for digital diary for creating alert message for our day to day activity
6. Develop a mobile app for bus ticket reservation system
7. Develop an application that draws basic graphical primitives (Line, Circle, Rectangle, ellipse) on the screen
8. Implement an application to change the back ground activity using multi threading
9. Develop an application that writes data to the SD card
10. Design a mobile app for marriage Invitation that uses GUI components, Font and colors

3. SPECIFIC LEARNING OUTCOMES

S.No.	Exercise List	Level
1	Develop a native calculator mobile app for arithmetic operations	K6
2	Develop an application that makes use of student database for end semester mark list	K6
3	Design a mobile app for hotel menu card using list view activity	K6
4	Develop a mobile app for expense tracker monitoring system of monthly budget	K6
5	Implement a mobile app for digital diary for creating alert message for our day to day activity	K6
6	Develop a mobile app for bus ticket reservation system	K6
7	Develop an application that draws basic graphical primitives (Line, Circle, Rectangle, ellipse) on the screen	K6
8	Implement an application to change the back ground activity using multi threading	K6
9	Develop an application that writes data to the SD card	K6
10	Design a mobile app for marriage Invitation that uses GUI components, Font and colors	K6

4. MAPPING SCHEME (CO, PO & PSO)

P21CS1P2	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PSO1	PSO2	PSO3	PSO4
CO1	H	M	H		H					H		M	
CO2	H	M	M	M	H	H							
CO3	H	H	H	H	M	L	M		H	M	H	L	
CO4	H	H	M	L	H	H		H	M	H	M	M	H
CO5	H	H		H	H		H	H	H	H	H	M	H
CO6	H		H	H	H		H	H	H	H	H	H	H

5. COURSE ASSESSMENT METHODS

DIRECT

1. Continuous Assessment Test: T1, T2 (Theory & Practical Components): Closed Book
2. Open Book Test.
3. Cooperative Learning Report, Assignment, Group Presentation, Group Discussion, project Report, Field Visit Report, Poster Presentation, Seminar, Quiz (written).
4. Pre-Semester & End Semester Theory Examination

INDIRECT

1. Course end survey (Feedback)

Name of the Course Coordinator: Dr. B. Karthikeyan

CORE VI: ADVANCED PROGRAMMING WITH .NET**SEMESTER: II**
CREDITS : 4**CODE: P21CS204**
HOURS/WEEK : 4

1. COURSE OUTCOMES

On successful completion of this course, students will be able to:

CO #	Course Outcomes	Level	Unit
CO1	Understand architecture of .NET Core, MVC web based Application.	K6	I
CO2	Implement real time applications using RazorView & signalR.	K6	II
CO3	Build interactive web based UI using AJAX & jQuery.	K5	III
CO4	Build interactive UI for desktop application.	K4	IV
CO5	Use advanced concepts related to Web Services and desktop application in project development.	K6	IV
CO6	Design the cloud applications using Azure Cloud	K6	V

2. A. SYLLABUS**Unit-I: ASP .NET Core**

Understanding .NET Core, .NET Core Features, .NET Core Framework Architecture, .NET Core 3.0 vs .NET 4.5 Framework, .NET Core Supports, Advantages of .NET Core, .NET App Model, .NET CLI. ASP .NET Core Project structure, Configuration, Creating & Injecting Service, MVC: MVC Architecture, MVC Middleware, Model, View, Controller, Action Methods, ASP.NET MVC Framework, Life Cycle of ASP.MVC Request, MVC State Management, Simple MVC based Web Application, Integration with Modern UI Frameworks, MVC Validation, Data Annotations, Security, Routing in MVC, Web API, CRUD using Web API, Authentication and Authorization: Role Based Authorization, JWT Token

Unit-II: Razor Views & SignalR

Razor View Engine, Types of Views, Razor Syntax Fundamentals, Razor Statements, Loops, Enum Support, Layout view Razor (Master Pages), Working with Sections, Working with Partial Views, SignalR with chat application. **Entity Framework Core:** Object Relational Mapping (ORM), Entity Framework, Database Modeling, CRUD Operations, Database Migration, Repository Design Pattern, Unit of Work Design Pattern, Dependency Injection

Unit-III: AJAX & JQuery

ASP.NET Ajax Introduction, Architecture , Using AJAX.NET Controls ,Accordion, Calendar ,Cascading Dropdown, Collapsible Panel , Filtered Textbox ,NumericUpDown, Modal Popup, Popup Control , ASP.NET Ajax Server Controls , ASP.NET Ajax Server Data, ASP.NET Ajax Client-side Library, ASP.NET Ajax Control Toolkit

Unit-IV: Web Services

The Motivation for XML Web Services, Creating an XML Web Service, Designing XML Web Services, Web Service Consumers, Discovering Web Services Using UDDI, Web Service using RESTful, Web Service SOAP, Developing Secure WebService. **Dot Net Core Desktop Application:**Application Architecture, Application Life Cycle, Navigation, Designing XAML UI, MVVM architecture, Data Binding, Entity Framework

Unit-V: Introduction to Azure Cloud and Advanced Topics

Cloud, WebAPI using Azure Cloud, Azure services, Microsoft Azure Storage, Azure Virtual Network, Azure Virtual Machines, Avalonia Framework, Xamarin

B. TOPICS FOR SELF STUDY

- ASP.NET Core with Razor Pages
- Database CRUD Application
- SQL Injection

C. TEXT BOOKS

1. Adam Freeman, “Pro ASP.NET MVC 5”, 6th Edition, Apress publication., 2016.
2. Mark J. Price, “C# 8.0 and .NET Core 3.0 – Modern Cross-Platform Development”, 4th Edition, Packt Publishing, 2019.

D. REFERENCE BOOKS

1. Kenneth Yamikani Fukizi, Jason De Oliveira, Michel Bruchet, “Learn ASP.NET Core 3” , Packt Publishing, 2019,
2. Tugberk Ugurlu, Alexander Zeitler, Ali Kheyrollahi , “Pro ASP.NET Web API”, Apress, 2013.
3. Dan Hermes, “Xamarin Mobile Application Development: Cross-Platform C# and Xamarin.Forms Fundamentals”, Springer, 2015
4. Ritesh Modi, “Azure for Architects”, 1st Edition, Microsoft, 2017

E. WEB LINKS

- <https://dotnet.microsoft.com/en-us/apps/aspnet>
- https://www.w3schools.com/asp/webpages_intro.asp
- https://www.halvorsen.blog/documents/programming/web/asp_net.php
- <https://www.tutorialspoint.com/asp.net/index.htm>

3. SPECIFIC LEARNING OUTCOMES

Unit	Course Content	Learning Outcomes	Level
I	ASP .NET Core		
1.1	Understanding .NET Core, .NET Core Features	Explain the features of the .NET Core	K5
1.2	.NET Core Framework Architecture .NET Core 3.0 vs .NET 4.5 Framework,	Construct the .Net Core Framework Compare the .NET Core 3.0 & 4.5	K6 K6
1.3	.NET Core Supports, Advantages of .NET Core	List the advantages of .NET Core	K2
1.4	NET App Model, .NET CLI. ASP .NET Core Project structure, Configuration, Creating & Injecting Service,	Understand the .NET App Model Configure the .NET App Model Create the Injecting Service	K3 K6 K6
1.5	MVC: MVC Architecture, MVC Middleware, Model, View, Controller, Action Methods, ASP.NET MVC Framework, Life Cycle of ASP.MVC Request, MVC State Management,	Explain the MVC Architecture Develop the MVC View and Controller Explain the MVC Life cycle List the MVC State management	K5 K6 K4 K3
1.6	Simple MVC based Web Application, Integration with Modern UI Frameworks, MVC Validation, Data Annotations, Security, Routing in MVC	Develop the MVC based Web application Assess the validation of the web application Design the securable web application using MVC	K6 K6 K6
1.7	Web API, CRUD using Web API,	Create the Web API using CRUD	K6

1.8	Authentication and Authorization: Role Based Authorization, JWT Token	Develop the Role based authorization for Web applications	K6
II Razor Views & SignalR			
2.1	Razor View Engine, Types of Views, Razor Syntax Fundamentals	Explain the Razor View Engine List the Razor Syntax	K5 K3
2.2	Razor Statements, Loops, Enum Support, Layout view Razor (Master Pages)	Develop the application using Razer	K6
2.3	SignalR with chat application.	Create the chat application using SignalIR	K6
2.4	Entity Framework Core: Object Relational Mapping (ORM), Entity Framework, Database Modeling	Design the Object Relational Mapping (ORM)	K6
		Construct the Database modelling	K6
2.5	CRUD Operations, Database Migration, Repository Design Pattern	Apply the CRUD operations on web applications Explain the Database Migration Design the Repository Design Pattern	K6 K5 K6
2.6	Unit of Work Design Pattern, Dependency Injection	Design the Work Design Pattern Create the Dependency Injection	K6 K6
III AJAX & JQuery			
3.1	ASP.NET Ajax Introduction, Architecture	Explain the AJAX architecture	K5
3.2	Using AJAX.NET Controls ,Accordion, Calendar	Construct the web application using AJAX Accodion & Calendar controls	K6
3.3	Cascading Dropdown, Collapsible Panel	Design the website using Ajax Cascading Dropdown, Collapsible Panel	K6
3.4	Filtered Textbox ,NumericUpDown	Develop the Web page using AJAX Filtered Textbox ,NumericUpDown	K6
3.5	Modal Popup, Popup Control	Design the Web App using Ajax Modal Popup, Popup Control	K6
3.6	ASP.NET Ajax Server Controls	Create the Website using ASP.NET Ajax Server Controls	K6
3.7	ASP.NET Ajax Server Data	Develop the web app using ASP.NET Ajax Server Data	K6
3.8	ASP.NET Ajax Client-side Library	Explain the Ajax Client-side Library	K4
3.9	ASP.NET Ajax Control Toolkit	Develop the web application using ASP.NET Ajax Control Toolkit	K6
IV Web Services			
4.1	The Motivation for XML Web Services	Explain the features of XML Web Services	K4
4.2	Creating an XML Web Service	Create the XML Web services	K6
4.3	Designing XML Web Services	Design the XML based Web services	K6
4.4	Web Service Consumers	Explain he Web service consumers	K4
4.5	Web Service using RESTful	Design the web services using RESTful	
4.6	Web Service SOAP	Design the SOAP based web services	K6
4.7	Developing Secure WebService	Create the securable webservice	K6
4.8	Dot Net Core Desktop Application: Application Architecture	Explain the Desktop Application Architecture	K3
4.9	Application Life Cycle, Navigation	List the steps in application life cycle	K3
4.10	Designing XAML UI, MVVM architecture	Design the XAML UI, MVVM architecture	K6
4.11	Data Binding, Entity Framework	Construct the entity framework	K6
V Introduction to Azure Cloud and Advanced Topics			
5.1	Cloud, WebAPI using Azure Cloud	Create web API using Azure Cloud	K6
5.2	Azure services	Explain the Azure the services	K3
5.3	Microsoft Azure Storage	List the Azure Storage	K3
5.4	Azure Virtual Network	Construct the Azure Virtual Network	K6
5.5	Azure Virtual Machines	Design the Virtual Machines	K6

5.6	Avalonia Framework	Explain the Avalonia Framework	K4
5.7	Xamarin	Construct the Xamarin	K6

4. MAPPING SCHEME (CO, PO & PSO)

P21CS204	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PSO1	PSO2	PSO3	PSO4
CO1	H		L		L								
CO2	H	M	M	M	H	H							
CO3	H	H	H	H	M	L	M		H	M	H	L	
CO4	H	H	M	L	H			H	M	H	M	M	H
CO5	H			H	H		H	H	H	H	H	M	H
CO6	H		H	H	H		H	H	H	H	H	H	H

5. COURSE ASSESSMENT METHODS

DIRECT:

1. Continuous Assessment Test: T1, T2 (Theory & Practical Components): Closed Book
2. Open Book Test.
3. Cooperative Learning Report, Assignment, Group Presentation, Group Discussion, project Report, Field Visit Report, Poster Presentation, Seminar, Quiz (written).
4. Pre-Semester & End Semester Theory Examination

INDIRECT:

1. Course end survey (Feedback)

Name of the Course Coordinator: Dr. B. Karthikeyan

CORE V: REAL TIME WEB APP DEVELOPMENT

SEMESTER: II
CREDITS: 4

CODE: P21CS205
HOURS/WEEK: 5

1. COURSE OUTCOMES

On successful completion of this course, students will be able to:

CO #	Course Outcomes	Level	Unit
CO1	Design and Develop the Web Applications using Node.js.	K5	I
CO2	Construct Web Applications using Express.js	K4	II
CO3	Understand MongoDB and its applicability.	K2	III
CO4	Illustrate the features of MongooseDB.	K4	IV
CO5	Build Web Applications using AngularJS.	K6	V
CO6	Develop various server side web applications using tools, middleware technologies and use them to develop web applications	K5	V

2. A. SYLLABUS

Unit I : Web App Development using Node.js

Three-tier web application development – Introducing NPM: Using NPM – Introduction to Node.js : JavaScript event-driven programming, Node.js event-driven programming – JavaScript closures – Node modules : CommonJS modules , Node.js core modules – Node modules : Node.js third-party modules, Node.js File modules, Node.js folder modules – Developing Node.js web applications: Meet the Connect module – Developing Node.js web applications: Connect middleware – Developing Node.js web applications: Understanding the order of Connect middleware – Developing Node.js web applications: Mounting Connect middleware.

Unit II : Built Web App using Express.js and MongoDB

Introduction to Express, Creating your First Express application – The application, request, and response objects, External middleware – Implementing the MVC pattern: Application folder structure, Horizontal folder structure, Vertical folder structure, File-naming conventions, Implementing the horizontal folder structure – Configuring an Express application: Environment configuration files – Rendering views: Configuring the view system, Rendering EJS views, Serving static files, Configuring sessions – MongoDB: Introduction to NoSQL, Introducing MongoDB – Key features of MongoDB: The BSON format, MongoDB ad hoc queries, MongoDB indexing, MongoDB replica set, MongoDB sharding – MongoDB shell, MongoDB databases, MongoDB collections – MongoDB CRUD operations: Creating a new document, Creating a document using insert(), Creating a document using update(), Creating a document using save(), Reading documents, Finding all the collection documents, Using an equality statement – MongoDB CRUD operations: Using query operators, Building AND/OR queries, Updating existing documents, Updating documents using update(), Updating documents using save(), Deleting documents, Deleting all documents.

Unit III : Understand the features of Mongoose

Introducing Mongoose: Connecting to MongoDB – Understanding Mongoose schemas: Creating the user schema and model, Registering the User model – Understanding Mongoose schemas: Creating new users using save(), Finding multiple user Documents using Find(), Advanced querying using Find(), Reading a single user document using FindOne()– Understanding Mongoose schemas: Updating an existing user document, Deleting an existing user document – Extending your Mongoose schema: Defining default values, Using schema modifiers – Extending your Mongoose schema: Predefined modifiers, Custom setter modifiers, Custom getter modifiers – Extending your Mongoose schema: Adding virtual attributes, Optimizing queries using indexes – Defining custom model methods: Defining custom static methods, Defining custom instance methods – Model validation: Predefined validators, Custom validators – Using Mongoose middleware: Using pre middleware, Using post middleware, Using Mongoose DBRef:

Unit IV : Key concepts of AngularJS

Introducing AngularJS – Key concepts of AngularJS: The core module of AngularJS, The angular global object – AngularJS modules: Application modules, External modules, Third-party modules – Two-way data binding, Dependency injection: Dependency injection in AngularJS – AngularJS directives: Core directives, Custom directives – Bootstrapping an AngularJS application: Automatic bootstrap, Manual bootstrap – Structuring an AngularJS application, Bootstrapping your AngularJS application, AngularJS MVC entities: AngularJS views, AngularJS controllers and scopes – AngularJS routing: Installing the ngRoute module, Configuring the URL scheme, AngularJS application routes – AngularJS services: AngularJS prebundled services, Creating AngularJS services, Using AngularJS services – Managing AngularJS authentication: Rendering the user object, Adding the Authentication service, Using the Authentication service.

Unit V : Creating a MEAN CRUD Module

Introducing CRUD modules, Setting up the Express components: Creating the Mongoose model, Setting up the Express controller – Setting up the Express components: The error handling method of the Express controller, The create() method of the Express controller, The list() method of the Express controller, The read() middleware of the Express controller – Setting up the Express components: The update() method of the Express controller, The delete method of the Express controller – Setting up the Express components: Implementing an authentication middleware, Implementing an authorization middleware – Setting up the Express components: Wiring the Express routes, Configuring the Express application – Introducing the ngResource module: Installing the ngResource module, Using the \$resource service – Implementing the AngularJS MVC module: Creating the AngularJS module service, Setting up the AngularJS module controller – Implementing the AngularJS MVC module: The create() method of the AngularJS controller, The Find() and FindOne() methods of the AngularJS controller – Implementing the AngularJS MVC module: The update() method of the AngularJS controller, The delete() method of the AngularJS controller, Implementing the AngularJS module views, The create-article view, The view-article view, The edit-article view, – Implementing the AngularJS MVC module: The list-articles view, Wiring the AngularJS module routes – Finalizing your module implementation.

B. TOPICS FOR SELF STUDY

- ReactJS – ReactJS Properties
- Class Components & Life Cycle of Methods
- State in ReactJS
- User Inputs, Forms and Events in ReactJS

C. TEXT BOOK(S)

1. Amos Q Vaviv, “*MEAN Web Development*”, Packt Publishing Ltd, 2016.
2. David Flanagan, “*JavaScript The Definitive Guide*”, O’Reilly Publishing Ltd, 2020.

D. REFERENCES BOOK(S)

1. Shane A Stillwell. “*Mastering MEAN web app development*”: Expert fullstack Javascript, PACKT Publishers, 2016. ISBN 9781785882159

E. WEB LINKS

- <https://www.udemy.com/course/react-learn-react-js-from-scratch-with-hands-on-projects/>
- <https://www.coursera.org/courses?query=angularjs>
- <https://www.codecademy.com/learn/learn-angularjs>

3. SPECIFIC LEARNING OUTCOMES

Unit	Topics	Unit Learning Outcomes	Level
I	Web App Development using Node.js		
1.1	Three-tier web application development	Describing three-tier web application architecture	K2
1.2	Introducing NPM	Annotating NPM and its applications.	K2

1.3	Using NPM		
1.4	Introduction to Node.js	Develop JavaScript and Node.js along with event driven programming.	K6
1.5	JavaScript event-driven programming		
1.6	Node.js event-driven programming		
1.7	JavaScript closures		
1.8	CommonJS modules		
1.9	Node.js core modules	Illustrating different types of JS modules	K4
1.10	Node.js third-party modules		
1.11	Node.js File modules		
1.12	Node.js folder modules		
1.13	Developing Node.js web applications	Creating Node.js code to connect module .	K6
1.14	Meet the Connect module		
1.15	Connect middleware	Construct Node.js web application program to link middleware	K4
1.16	Understanding the order of Connect middleware	Execute Node.js web applications to understanding the order and mounting for connect the middleware.	K6
1.17	Mounting Connect middleware		
II	Built Web App using Express.js and MongoDB		
2.1	Introduction to Express	Recalling Express.js	K2
2.2	Creating your First Express application	Executing simple Express.js application.	K3
2.3	The application, request, and response objects		
2.4	External middleware	Describe the external middleware.	K2
2.5	Implementing the MVC pattern	Demonstrating the MVC pattern using three tier architecture	K3
2.6	Application folder structure	Distinguish different folder structures in express.js	K2
2.7	Horizontal folder structure		
2.8	Vertical folder structure		
2.9	File	Converting file name in express.js application.	K2
2.10	naming conventions		
2.11	Implementing the horizontal folder structure	Applying the horizontal folder structure in express.js.	K3
2.12	Configuring an Express application	Develop an express application and configuration.	K6
2.13	Environment configuration files		
2.14	Rendering views	Illustrating the various views in the express js	K4
2.15	Configuring the view system		
2.16	Rendering EJS views		
2.17	Serving static files	Accessing static files in express.js	K5
2.18	Configuring sessions	Identifying session in express.js application.	K5
2.19	Introduction to NoSQL	Describing NoSQL database.	K2
2.20	Introducing MongoDB	Explaining MongoDB.	K2
2.21	Key features of MongoDB	Expressing the various MongoDB features of BSON format, indexing, replica set, shell.	K5
2.22	The BSON format		
2.23	MongoDB ad hoc queries		
2.24	MongoDB indexing		
2.25	MongoDB replica set		
2.26	MongoDB sharding		
2.27	MongoDB shell		
2.28	MongoDB databases		
2.29	MongoDB collections		
2.30	MongoDB CRUD operations		

2.31	Creating a document using save()		
2.32	Reading documents		
2.33	Finding all the collection documents		
2.34	Using an equality statement	Justifying specific condition in MangoDB using query operator such as AND and OR .	K5
2.35	MongoDB CRUD operations		
2.36	Using query operators		
2.37	Building AND/OR queries		
2.38	Updating existing documents		
2.39	Updating documents using update()	Examine the update(),save(),and delete() documents using MangoDB	K3
2.40	Updating documents using save()		
2.41	Deleting documents		
2.42	Deleting all documents		
III	Understand the features of Mongoose		
3.1	Introducing Mongoose	Recallthe basic concept of Mongoose .	K2
3.2	Connecting to MongoDB	Building a MongoDB connection	K6
3.3	Understanding Mongoose schemas	Identifying the structure of schema in Mongoose	K2
3.4	Creating the user schema and model	Developing the model based on user specific schema	K6
3.5	Registering the User model	Extending the user model with registration	K2
3.6	Creating new users using save()	Execute save() by creating authorised user	K3
3.7	Finding multiple user Documents using Find()	Apply find() in query to select multiple user documents in Mongoose	K3
3.8	Advanced querying using Find()		
3.9	Reading a single user document using FindOne()	Using FindOne() , to read a single user document	K3
3.10	Updating an existing user document	Examineupdated and deleted documents in Mongoose	K3
3.11	Deleting an existing user document		
3.12	Extending your Mongoose schema	Apply default values in Mongoose schema level.	K3
3.13	Defining default values		
3.14	Using schema modifiers		
3.15	Predefined modifiers		
3.16	Custom setter modifiers	Computing Mongoose predefined getter and setter modifier	K3
3.17	Custom getter modifiers		
3.18	Extending your Mongoose schema		
3.19	Adding virtual attributes	Createvirtual attributes to extendMongooseDB.	K6
3.20	Optimizing queries using indexes	Executequery with index	K3
3.21	Defining custom model methods	Explaining custom model, static and instance methods .	K2
3.22	Defining custom static methods		
3.23	Defining custom instance methods		
3.24	Model validation	Differentiate model validation with predefined and custom built-in validators .	K2
3.25	Predefined validators		
3.26	Custom validators		
3.27	Using Mongoose middleware	IllustratingMongoose and pre middleware	K4
3.28	Using pre middleware		
IV	Key concepts of AngularJS		
4.1	Introducing AngularJS	Describing AngularJS concept	K2
4.2	Key concepts of AngularJS	Explaining the basic concepts of AngularJS and its modules, global object.	K2
4.3	The core module of AngularJS		
4.4	The angular global object		
4,5	AngularJS modules	Classifying various modules of AngularJS .	

4.6	Application modules		K4
4.7	External modules		
4.8	Third-party modules		
4.9	Two-way data binding		
4.10	Dependency injection	Discussing dependency injection in AngularJS	K2
4.11	Dependency injection in AngularJS		
4.12	AngularJSdirectives	Exemplifying built-in directives in AngularJS.	K2
4.13	Core directives		
4.14	Custom directives	Creating a custom directive	K6
4.15	Bootstrapping an AngularJS application	Developing an AngularJS application by bootstrapping.	K6
4.16	Automatic bootstrap	Distinguish Automatic bootstrap with manual bootstrap in AngularJS	K2
4.17	Manual bootstrap		
4.18	Structuring an AngularJS application	Constructing an AngularJS application	K3
4.19	AngularJS MVC entities	Explaining MVC framework.	K4
4.20	AngularJS views	Illustrating the views of MVC framework.	K4
4.21	AngularJS controllers and scopes	Describing AngularJS controllers and it understand scopes.	K2
4.22	AngularJS routing	Explain routing in AngularJS.	K2
4.23	Installing the ngRoute module	Preparing ngRoute module	K3
4.24	Configuring the URL scheme	Constructing the URL scheme in AngularJS application.	K3
4.25	AngularJS application routes	Create single page application using routes in AngularJS .	K6
4.26	AngularJS services	Demonstrate theprebundled services in AngularJS.	K2
4.27	AngularJSprebundled services		
4.28	Creating AngularJS services	Develop a code for AngularJS services	K6
4.29	Using AngularJS services	Applying AngularJS services	K3
4.30	Managing AngularJS authentication	Adapting AngularJS authentication.	K6
4.31	Rendering the user object		
4.32	Adding the Authentication service	Managing theAuthentication services in AngularJS authentication.	K6
4.33	Using the Authentication service.		
V	Creating a MEAN CRUD Module		
5.1	Introducing CRUD modules	Describe the basic building blocks of CRUD modules	K2
5.2	Setting up the Express components	Linkthe Express components for CRUD modules.	K4
5.3	Creating the Mongoose model	Develop the Mongoose model using CRUD modules.	K6
5.4	Setting up the Express controller	Recommend the Express controller set-up in the CRUD modules.	K4
5.5	The error handling method of the Express controller	Prioritize the error handling methods of theExpress controller.	K4
5.6	The create() method of the Express controller	Apply create(), list(), read(), update(), delete() methods of the Express controller in AngularJS.	K3
5.7	The list() method of the Express controller		
5.8	The read() middleware of the Express controller		
5.1	The update() method of the Express controller		
5.2	The delete method of the Express controller		
5.3	Implementing an authentication middleware	Facilitating middleware authentication and authorization techniques.	K6

5.4	Implementing an authorization middleware		
5.5	Wiring the Express routes	Examine the Express routes.	K3
5.6	Configuring the Express application	Organize the Express application.	K4
5.7	Introducing the ngResource module	Describe ngResource module	K2
5.8	Installing the ngResource module	Order the steps in ngResource module.	K4
5.9	Using the \$resource service	Create a RESTfulAPI using \$resource service	K6
5.10	Implementing the AngularJS MVC module	Applying the AngularJS MVC module	K3
5.11	Creating the AngularJS module service	Develop a module as a service AngularJS	K6
5.12	Setting up the AngularJS module controller	Prepare an AngularJS module controller set up.	K3
5.13	Implementing the AngularJS MVC module	Running the AngularJS MVC module	K3
5.14	The create() method of the AngularJS controller	Apply create(), find (), findone(), update(), delete() methods in the AngularJS controller.	K6
5.15	The Find() and FindOne() methods of the AngularJS controller		
5.16	The update() method of the AngularJS controller		
5.17	The delete() method of the AngularJS controller		
5.18	Implementing the AngularJS module views	Examine multiple views in the AngularJS module.	K3
5.19	The create-article view	Choosing appropriate views in the AngularJS MVC module	K3
5.20	The view-article view		
5.21	The edit-article view		
5.22	The list-articles view		
5.23	Wiring the AngularJS module routes	Develop a code using AngularJS module routes.	K6
5.24	Finalizing your module implementation	Concluded the AngularJS module implementation procedure.	K5

4. MAPPING SCHEME (CO, PO & PSO)

P21CS205	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PSO1	PSO2	PSO3	PSO4
CO1	H	L	H	H	M	L	M	-	M	H	M	M	M
CO2	M	L	H	M	M	-	H	M	H	H	H	M	H
CO3	H	M	M	L	L	L	-	-	-	M	L	-	-
CO4	H	M	L	L	-	-	-	-	-	L	L	M	L
CO5	M	M	H	H	H	H	H	M	H	M	H	M	H
CO6	M	H	H	H	M	M	M	H	H	H	H	M	H

5. COURSE ASSESSMENT METHODS

DIRECT

1. Continuous Assessment Test: T1, T2 (Theory & Practical Components): Closed Book
2. Open Book Test.
3. Group Discussion, Poster Presentation, Seminar, Quiz (written).
4. Pre-Semester & End Semester Theory Examination

INDIRECT

1. Course end survey (Feedback)

Name of the Course Coordinator: Dr. M. P. Anuradha

CORE IV: CLOUD AND BIG DATA COMPUTING

SEMESTER: II

CODE: P21CS206

CREDITS : 4

HOURS/WEEK : 4

1. COURSE OUTCOMES

On successful completion of this course, students will be able to:

CO. #	Course Outcomes	Level	Unit
CO1	Understand the concepts of Cloud computing	K6	I
CO2	Develop and evaluate the cloud services	K6	II
CO3	Assess the SLA and Data Security in Cloud	K5	III
CO4	Explain the Big Data Fundamentals	K4	IV
CO5	Design the Big Data Storage concepts	K6	IV
CO6	Analyse the Big Data models	K5	V

2. A. SYLLABUS

UNIT I: Cloud Computing Fundamentals

Roots of Cloud Computing-Layers and Types of Clouds-Desired Features of a Cloud-Cloud Infrastructure Management-Challenges and Risks-Migrating into a Cloud: Broad Approaches to Migrating into the Cloud-The Seven-Step Model of Migration into a Cloud-Integration as a Service: An Introduction-Onset of Knowledge Era-Evolution of SaaS-Integration Scenarios-Integration Methodologies-SaaS Integration Products and Platforms - Integration Services- Businesses-to-Business Integration (B2Bi) Services.

UNIT II: Cloud Services

Infrastructure As A Service: Introduction-Virtual Machines Provisioning and Manageability-Virtual Machine Migration Services- Platform And Software As A Service: Introduction -Technologies and Tools for Cloud Computing-Cloud- Based Solutions for Business Applications: Introduction -What Enterprises Demand of Cloud Computing - Dynamic ICT Services-Workflow Engine for Clouds: Workflow Management Systems and Clouds - Architecture of Workflow Management Systems-MapReduce Programming Model: MapReduce Programming Model-Major MapReduce Implementations for the Cloud.

UNIT III: SLA and Data Security

SLA Management in Cloud Computing: Types of SLA-Life Cycle of SLA-SLA Management in Cloud-AWS Cloud: Understanding the Amazon Web Services Cloud-Resource Cloud Mashups: Concepts of a Cloud Mashup-Data Security in the Cloud: Introduction-Current State of Data Security in the Cloud-Cloud Computing and Data Security Risk - Cloud Computing and Identity - The Cloud, Digital Identity, and Data Security-Legal Issues in Cloud Computing: Data Privacy and Security Issues -Cloud Contracting models.

UNIT IV: Fundamentals of Bigdata and Big Data Processing

Understanding Big Data: Concepts and Terminology-Big Data Characteristics- Different Types of Data-Business Motivations and Drivers for Big Data Adoption: Information and Communications Technology-Internet of Everything (IoE)- Big Data Adoption and Planning Considerations: Big Data Analytics Lifecycle-Enterprise Technologies and Big Data Business Intelligence- Big Data Storage Concepts: Clusters- File Systems and Distributed File Systems- NoSQL- Sharding- Replication.

UNIT V: Analyzing Bigdata

Big Data Processing Concepts: Distributed Data Processing-Hadoop- Parallel Data Processing- Processing Workloads-cluster-Processing in Batch Mode-Big Data Analysis Techniques: Quantitative Analysis- Qualitative Analysis- Statistical Analysis- Machine Learning- Semantic Analysis- Visual Analysis.

B. TOPICS FOR SELF STUDY

- Big Data with Spark
- Hbase & Kafka
- Big Data Machine Learning
- Big Data Predictive Analysis

C. TEXT BOOKS

1. Rajkumar Buyya, James Broberg, Andrzej Goscinski, ‘Cloud Computing Principles and Paradigms’, Wiley Publishers, 2013.
2. Paul Buhler, WajidKhattak and Thomas Erl, “Big Data Fundamentals: Concepts, Drivers & Techniques”, Prentice Hall Publications, 2016

D. REFERENCE BOOKS

1. Barrie Sosinsky, “ Cloud Computing Bible” John Wiley & Sons, 2010
2. SoumendraMohanty, MadhuJagadeesh, and HarshaSrivatsa, “Big Data Imperatives: Enterprise Big Data Warehouse, BI Implementations and Analytics”, Published by Apress Media, 2013.
3. Tom White, “Hadoop: The Definitive Guide”, Third Edition, O’reilly Media, 2012

E. WEB LINKS

- <https://nptel.ac.in/courses/106/104/106104189/>
- <https://www.coursera.org/specializations/big-data>
- <https://cognitiveclass.ai/courses/what-is-big-data/>
- <https://www.edx.org/learn/big-data>

3. SPECIFIC LEARNING OUTCOMES

Unit	Course Content	Learning Outcomes	Level
I	Cloud Computing Fundamentals		
1.1	Roots of Cloud Computing	Explain the evolution of Cloud Computing	K4
1.2	Layers and Types of Cloud	Explain the layer and types of cloud	K4
1.3	Desired Features of a Cloud-Cloud Infrastructure Management	Assess the Cloud infrastructure management	K6
1.4	Challenges and Risks	Evaluate the Challenges and risks in cloud computing	K5
1.5	Migrating into a Cloud: Broad Approaches to Migrating into the Cloud	Use the approaches to traditional to cloud	K5
1.6	The Seven-Step Model of Migration into a Cloud	Explain the steps of migration to cloud	K5
1.7	Integration as a Service: An Introduction	Explain the Integration as a Service	K5
1.8	Onset of Knowledge Era	Assess the features of Onset of Knowledge Era	K5
1.9	Evolution of SaaS	Explain the evolution of SaaS	K5
1.10	Integration Scenarios	List the Integration Scenarios	K4
1.11	Integration Methodologies	Evaluate the methods for integration	K5
1.12	SaaS Integration Products and Platforms	Build the SaaS Integration platform and products	K6

1.13	Integration Services	Explain the Integration services	K4
1.14	Businesses-to-Business Integration (B2Bi) Services.	Construct the B2Bi services	K5
II	Cloud Services		
2.1	Infrastructure As a Service: Introduction-Virtual Machines Provisioning and Manageability-Virtual Machine Migration Services	Explain the IaaS Design the Virtual Machines Migration services	K5 K6
2.2	Platform And Software As A Service: Introduction -Technologies and Tools for Cloud Computing	Explain the platform and Software As A Service List the Tools for SaaS & PaaS	K5 K4
2.3	Cloud- Based Solutions for Business Applications: Introduction -What Enterprises Demand of Cloud Computing - Dynamic ICT Services	Evaluate the solutions of Business applications Design the Dynamic ICT Services using Cloud computing	K5 K5
2.4	Workflow Engine for Clouds: Workflow Management Systems and Clouds - Architecture of Workflow Management Systems	Design the Workflow engine for clouds Explain the architecture of workflow management systems	K6 K5
2.5	MapReduce Programming Model: MapReduce Programming Model-Major MapReduce Implementations for the Cloud	Explain the MapReduce Programming Model Implement the MapReduce Programming Model in cloud	K5 K6
III	SLA and Data Security		
3.1	SLA Management in Cloud Computing: Types of SLA-Life Cycle of SLA-SLA Management in Cloud	Explain the Types of SLA Explain the Lifecycle of SLA Evaluate the SLA management in cloud	K4 K4 K5
3.2	AWS Cloud: Understanding the Amazon Web Services Cloud	Design the Amazon web services using AWS Cloud	K6
3.3	Resource Cloud Mashups: Concepts of a Cloud Mashup	Explain the Resources Cloud Mashups	K5
3.4	Data Security in the Cloud: Introduction-Current State of Data Security in the Cloud-Cloud Computing and Data Security Risk	Evaluate the Data security in the cloud Assess the risks in cloud computing	K6 K6
3.5	Cloud Computing and Identity - The Cloud, Digital Identity, and Data Security	Explain the Cloud identity Assess the data security in cloud	K5 K6
3.6	Legal Issues in Cloud Computing: Data Privacy and Security Issues -Cloud Contracting models.	Explain the legal issues in cloud Assess the cloud contracting models	K4 K6
IV	Fundamentals of Bigdata and Big Data Processing		
4.1	Understanding Big Data: Concepts and Terminology-Big Data Characteristics-Different Types of Data	Explain the concepts of Big Data Explain the types of Big Data	K4 K4
4.2	Business Motivations and Drivers for Big Data Adoption: Information and Communications Technology-Internet of Everything (IoE)	Evaluate the Big Data adoption in ICT Construct the IoE	K6 K6
4.3	Big Data Adoption and Planning Considerations: Big Data Analytics Lifecycle-Enterprise Technologies and Big Data Business Intelligence	Explain the Big Data Analytics Lifecycle Build the Enterprise Technologies	K4 K6
4.4	Big Data Storage Concepts: Clusters- File Systems and Distributed File Systems-NoSQL- Sharding- Replication.	Explain the strategies of Big Data Storage Concepts Uses of NoSQL in Big Data	K5 K4
V	Analyzing Bigdata		
5.1	Big Data Processing Concepts: Distributed Data Processing-Hadoop-	Construct the Hadoop model for Big Data Processing	K5

5.2	Parallel Data Processing- Processing Workloads- Cluster-Processing in Batch Mode	Explain the Parallel Data Processing Evaluate the Cluster processing in batch mode	K4 K5
5.3	Big Data Analysis Techniques: Quantitative Analysis	Evaluate the Quantitative analysis in Big Data	K5
5.4	Qualitative Analysis - Statistical Analysis	Assess the methods for statistical analysis	K6
5.5	Machine Learning- Semantic Analysis- Visual Analysis.	Evaluate the ML and visual analysis for big data	K6

4. MAPPING SCHEME (CO, PO & PSO)

P21CS206	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PSO1	PSO2	PSO3	PSO4
CO1	H		L		L								
CO2	H	M	M	M	H	H							
CO3	H	H	H	H	M	L	M		H	M	H	L	
CO4	H	H	M	L	H			H	M	H	M	M	H
CO5	H			H	H		H	H	H	H	H	M	H
CO6	H		H	H	H		H	H	H	H	H	H	H

5. COURSE ASSESSMENT METHODS

DIRECT:

1. Continuous Assessment Test: T1, T2 (Theory & Practical Components): Closed Book
2. Open Book Test.
3. Cooperative Learning Report, Assignment, Group Presentation, Group Discussion, project Report, Field Visit Report, Poster Presentation, Seminar, Quiz (written).
4. Pre-Semester & End Semester Theory Examination

INDIRECT:

1. Course end survey (Feedback)

Name of the Course Coordinator: Prof. L. Priscilla

ELECTIVE 2 A: NETWORK MANAGEMENT

SEMESTER: II
CREDITS: 4

CODE: P21CS2:1
HOURS/WEEK: 5

1. COURSE OUTCOMES

On successful completion of this course, students will be able to:

CO #	Course Outcomes	Level	Unit Covered
CO1	Interpret the communication protocols and standards	K4	I
CO2	Elaborate various aspects of network management	K5	II
CO3	Adapt different technologies for networking	K4	III
CO4	Analyze transmission technologies and management models	K5	IV
CO5	Illustrate the model of Simple Network Management Protocol	K5	V
CO6	Discover and apply Network Management Tools, System, and Engineering	K6	V

2. A. SYLLABUS

Unit I :Introduction to Data Communication and Network Management

Analogy of Telephone Network management, Data (Computer) and Telecommunication Network - Distributed Computing Environment - TCP Based Network - Communication Protocols and Standard : Communication Architecture, Protocol Layer Services - Network Management: Goals, Organization and function : Goal of Network Management, Network Provisioning, Network Operation and NOC, Network Installation and Maintenance - Network Management Architecture and Organization - Network Management Perspective.

Unit II :Network and Technology

Network Technology - Local Area Network : Ethernet, Fast Ethernet, Gigabit Ethernet, Full-Duplex Ethernet, Virtual LAN, Token ring, FDDI, Wireless LAN - Network Interconnecting Device : Hubs, Bridges, Remote Bridges, Transparent Bridges, Source-Routing Bridges, Routers, Gateways and Protocols Converter, Multiprotocol and Tunneling, Half Bridge Configuration of Router, Edge Router, Switches.

Unit III :Transmission Technology & Basic foundations of Network Management :

Standards, Models and Languages. Wide Area Network, Transmission Technology: Wired Transmission - Transmission Technology : Wireless Transmission - Transmission Model - Network Management Standards - Network Management Models, Organization Model - Information Model - Communication Model - Abstract Syntax Notation One ANS.1 – Macros - Functional Model.

Unit IV : SNMPv1 Network Management

Managed Network - History of SNMP Management, Internet Organization and Standards - SNMP Model - SNMP Model : Organization Model - Information Model : Structure of Management Information, Information Model, Management of Information Base.

Unit V :SNMP Communication and Functional Models & Network Management Tools, System, and Engineering

SNMP Communication Model: SNMP Architecture , Administrative Model, SNMP Protocol Specification , SNMP Operation,SNMP MIB Group. Functional Model - **Network Management Tools, System, and Engineering** System Utilities for Management : Basic Tools , SNMP Tools, Protocol Analyzer, Network Statistics -Measurement System: Traffic Load Monitoring - Network Statistics Measurement System : Protocol Statistics , Data and Error Statistics, Using MRTG to collect Traffic Statistics - MIB Engineering : General Principles and Limitations of SMI, Object-Oriented Approach t MIB Engineering, SMI Tables, SMI Actions, SMI Transactions - NMS Design - Network Management System.

B. TOPICS FOR SELF STUDY

- Centralized management and monitoring of Cisco network infrastructure
- Configure standard network monitoring protocols: SNMP, Syslog, NTP
- Configure standard network monitoring protocols: SNMP, Syslog, NTP
- Out-Of-Band (Oob) Management For Back Door Access

C. TEXT BOOK(S)

1. Mani Subramanian; Timothy A. Gonsalves; N. Usha Rani, “*Network Management: Principles and Practice*”, Pearson Education India, 2015

D. REFERENCES BOOK(S)

1. James F Kurose and Keith W. Ross, “*Computer Networking*” – A Top-Down Approach, Sixth Edition, 2013, Pearson Publication.

E. Web Links

- <https://www.udemy.com/course/managing-cisco-network-devices-netops/>
- <https://mcc.iavalley.edu/academic/computer-science-information-technologies/computer-network-mgmt/>
- <https://www.edx.org/learn/computer-networking>

3. SPECIFIC LEARNING OUTCOMES

Unit	Topics	Unit Learning Outcomes	Level
I	Introduction to Data Communication and Network Management		
1.1	Analogy of Telephone Network management, Data (Computer) and Telecommunication Network	Categorize the different features of networks	K4
1.2	Distributed Computing Environment	Elaborate the distributed environment	K6
1.3	TCP Based Network	Discuss the transmission control protocol	K6
1.4	Communication Protocols and Standard : Communication Architecture, Protocol Layer Services	Compare the services offered by layers and architectures	K4
1.5	Network Management: Goals, Organization and function : Goal of Network Management	Analyze the goal of network management	K4
1.6	Network Provisioning, Network Operation and NOC	Build the network based on provisions	K6
1.7	Network Installation and Maintenance	Construct the network and maintain it	K3
1.8	Network Management Architecture and Organization	Discuss the organization of network and apply it for any network	K6
1.9	Network Management Perspective	Perceive management perspective	K5
II	Network and Technology		
2.1	Network Technology - Local Area Network	Identify the various technology of network	K3
2.2	Ethernet, Fast Ethernet, Gigabit Ethernet, Full-Duplex Ethernet, Virtual LAN, Token ring, FDDI, Wireless LAN	Compare the different Ethernets and its usages	K5
2.3	Network Interconnecting Device	Elaborate the devices used for interconnection	K6
2.4	Hubs, Bridges, Remote Bridges, Transparent Bridges, Source-Routing Bridges, Routers, Gateways and Protocols Converter, Multiprotocol and Tunneling, Half Bridge	Compare various routing hardware and its features	K5

	Configuration of Router, Edge Router, Switches		
III	Transmission Technology & Basic foundations of Network Management		
3.1	Standards, Models and Languages	Analyze the different standards, models and languages available for networking	K4
3.2	Wide Area Network	Construct a WAN through different technologies	K6
3.3	Transmission Technology: Wired Transmission	Elaborate the features of wired transmission	K6
3.4	Wireless Transmission	Elaborate the features of wireless transmission	K6
3.5	Transmission Model	Analyze various models of transmission	K4
3.6	Network Management Standards	Agree with the right standard for establishing network	K5
3.7	Network Management Models, Organization Model	Discuss the model of organization level	K6
3.8	Information Model	Discuss the model of information	K6
3.9	Communication Model	Discuss the model of communication	K6
3.10	Abstract Syntax Notation One ANS.1	Build a notation for networking	K6
3.11	Macros - Functional Model	Inference the function model	
IV	SNMPv1 Network Management		
4.1	Managed Network	Discuss on the managed network	K6
4.2	History of SNMP Management, Internet Organization and Standards	Perceive the history and the standards	K5
4.3	SNMP Model	Model the protocol called SNMP	K3
4.4	Organization Model	Discuss the model of organization	K6
4.5	Information Model: Structure of Management Information	Elaborate the structure of MI	K6
4.6	Management of Information Base	Discuss the information base	K6
V	SNMP Communication and Functional Models & Network Management Tools, System, and Engineering		
5.1	SNMP Communication Model	Model and apply the SNMP for networking	K3
5.2	SNMP Architecture	Examine the architecture of SNMP for networking	K4
5.3	Administrative Model	Adapt the model for administration	K6
5.4	SNMP Protocol Specification	Make use of the SNMP protocol specification for any network for communication	K3
5.5	SNMP Operation, SNMP MIB Group	Function the operations of SNMP and distinguish the MIB group	K4
5.6	Functional Model	Analyze the functional model of SNMP	K4
5.7	Network Management Tools, System, and Engineering System Utilities for Management	Make use of various things like tools etc for management	K3
5.8	Basic Tools , SNMP Tools, Protocol Analyzer, Network Statistics	Adapt the tools like analyzer for networking	K6
5.9	Measurement System: Traffic Load Monitoring	Experiment with various measurement system for performance tuning of any network	K3
5.10	Network Statistics, Protocol Statistics , Data and Error Statistics, Using MRTG to collect Traffic Statistics	Compile various statistical methods for networking	K6
5.11	MIB Engineering : General Principles and Limitations of SMI, Object-Oriented Approach t	Discuss various aspects of MIB and SMI	K6

	MIB Engineering, SMI Tables, SMI Actions, SMI Transactions		
	NMS Design - Network Management System	Discuss NMS with various networks as examples	K6

4. MAPPING SCHEME (CO, PO & PSO)

P21CS2:1	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PSO1	PSO2	PSO3	PSO4
CO1	H	L	H	H	M	L	M	-	M	H	M	M	M
CO2	M	L	H	M	M	-	H	M	H	H	H	M	H
CO3	H	M	M	L	L	L	-	-	-	M	L	-	-
CO4	H	M	L	L	-	-	-	-	-	L	L	M	L
CO5	M	M	H	H	H	H	H	M	H	M	H	M	H
CO6	M	H	H	H	M	M	M	H	H	H	H	M	H

5. COURSE ASSESSMENT METHODS

DIRECT

1. Continuous Assessment Test: T1, T2 (Theory & Practical Components): Closed Book
2. Open Book Test.
3. Cooperative Learning Report, Assignment, Group Presentation, Group Discussion, project Report, Field Visit Report, Poster Presentation, Seminar, Quiz (written).
4. Pre-Semester & End Semester Theory Examination

INDIRECT

1. Course end survey (Feedback)

Name of the Course Coordinator: Dr. M. Jayakkumar

ELECTIVE 3a: BLOCK CHAIN AND CRYPTO-CURRENCIES

SEMESTER: II
CREDITS: 4

CODE: P21CS2:3
HOURS/WEEK: 4

1. COURSE OUTCOMES

On successful completion of this course, students will be able to:

CO #	Course Outcomes	Level	Unit Covered
1	Perceive the role cryptography plays in securing block chain based cryptocurrencies	K5	I
2	Analyze transactions and operations on block chains and smart contracts required to plan more robust financial systems	K4	II
3	Discuss how emerging block chain and cryptocurrencies ecosystems are set to impact economies, finance, government and regulation	K5	III
4	Examine the technologies behind block chain and explore their application in the business context	K4	IV
5	Design, build and deploy smart contracts and distributed applications	K5	V
6	Elaborate the intrinsic value of leading cryptocurrencies, Bitcoin and Ethereum	K6	V

2. A. SYLLABUS

Unit I: Blockchain, Decentralization

Blockchain :The growth of blockchain technology - Distributed systems - The history of blockchain and Bitcoin - Blockchain - Consensus - CAP theorem and blockchain. **Decentralization:** Decentralization using blockchain - Methods of decentralization -Routes to decentralization - Blockchain and full ecosystem decentralization - Pertinent terminology - Platforms for decentralization - Innovative trends.

Unit II: Public Key Cryptography, Consensus Algorithms and Smart Contracts

Public Key Cryptography: Asymmetric cryptography - Cryptographic constructs and blockchain technology. **Consensus Algorithms:** Introducing the consensus problem -Analysis and design - Classification - Algorithms - Choosing an algorithm. **Smart Contracts:** History - Definition - Ricardian contracts - Smart contract templates – Oracles - Deploying smart contracts - DAO

Unit III: Bitcoin

Bitcoin: Bitcoin—an overview - Cryptographic keys - Transactions - Blockchain – Mining. **Bitcoin Network and Payments:** The Bitcoin network - Wallets - Bitcoin payments -Innovation in Bitcoin - Advanced protocols - Bitcoin investment and buying and selling Bitcoin. **Bitcoin Clients and APIs:** Bitcoin client installation - Experimenting further with bitcoin-cli - Bitcoin programming.

Unit IV: Alternative Coins

Alternative Coins: Theoretical foundations - Difficulty adjustment and retargeting algorithms - Bitcoin limitations - Extended protocols on top of Bitcoin -Development of altcoins. **Ethereum:** **Ethereum** – an overview - Ethereum network - Components of the Ethereum ecosystem - Ethereum

Virtual Machine (EVM) - Smart contracts. - Blocks and blockchain - Wallets and client - Nodes and miners - APIs, tools, and DApps - Supporting protocols - Programming languages.

Unit V: Development Tools and Frameworks, Use Cases & Security

Development Tools and Frameworks : Languages - Compilers - Tools and libraries - Frameworks - Contract development and deployment - Layout of a Solidity source code file - Solidity language. **Use Cases:** IoT – Government - Health -Finance – Media. **Scalability and Other Challenges:** Scalability - Privacy - Security - Other challenges.

B. TOPICS FOR SELF STUDY

- Bitcoin and Anonymity
- Alternative Mining Puzzles
- Altcoins and the Cryptocurrency Ecosystem
- Future of Bitcoins

C. TEXT BOOKS

1. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, Steven Goldfeder. Bitcoin and Cryptocurrency Technologies. Princeton University Press, 2016. ISBN 978-0691171692

D. REFERENCES

1. Andreas Antonopoulos. Mastering Bitcoin: Programming the open block chain. Oreilly Publishers, 2017. ISBN 978-9352135745

E. Web Links

- <https://www.coursera.org/learn/cryptocurrency>
- <https://www.edx.org/learn/blockchain>
- <https://online.stanford.edu/courses/soe-xcs0001-blockchain-and-cryptocurrency-what-you-need-know>

3. SPECIFIC LEARNING OUTCOMES

Unit	Course Content	Learning Outcomes	Level
I	Blockchain, Decentralization		
1.1	Blockchain : The growth of blockchain technology - Distributed systems - The history of blockchain and Bitcoin -	Explain the distributed system Explain the evolution of blockchain & bitcoin	K4 K4
1.2	Blockchain - Consensus - CAP theorem and blockchain.	Apply the CAP Theorem on blockchain	K5
1.3	Decentralization: Decentralization using blockchain - Methods of decentralization - Routes to decentralization -	Evaluate the methods of decentralization	K5
1.4	Blockchain and full ecosystem decentralization - Pertinent terminology -	Construct the ecosystem of decentralization	K6
1.5	Platforms for decentralization - Innovative trends.	List the platforms for decentralization	K3
II	Public Key Cryptography, Consensus Algorithms and Smart Contracts		
2.1	Public Key Cryptography: Asymmetric cryptography - Cryptographic constructs and blockchain technology.	Compare the public key cryptography algorithms Apply the cryptographic algorithm for block chain	K6 K5
2.2	Consensus Algorithms: Introducing the consensus problem -Analysis and design -	Analyze the Consensus algorithms Evaluate the consensus algorithms	K5 K6

	Classification - Algorithms - Choosing an algorithm.	Compare and choose the best algorithms	K6
2.3	Smart Contracts: History - Definition - Ricardian contracts - Smart contract templates – Oracles - Deploying smart contracts - DAO	Explain the smart contracts Deploy the smart contracts	K5 K6
III			
3.1	Bitcoin: Bitcoin—an overview - Cryptographic keys - Transactions - Blockchain – Mining.	Explain the transaction on block chain Evaluate the mining techniques	K5 K6
3.2	Bitcoin Network and Payments: The Bitcoin network - Wallets - Bitcoin payments -Innovation in Bitcoin - Advanced protocols - Bitcoin investment and buying and selling Bitcoin	List the different types of wallets Build the advanced protocols for bitcoin	K3 K6
3.3	Bitcoin Clients and APIs: Bitcoin client installation - Experimenting further with bitcoin-cli - Bitcoin programming.	Explain the Bitcoin Clients Design the APIs for bitcoin	K5 K6
IV			
Alternative Coins			
4.1	Alternative Coins: Theoretical foundations - Difficulty adjustment and retargeting algorithms - Bitcoin limitations - Extended protocols on top of Bitcoin - Development of altcoins.	Develop the altcoins Explain the different types of altcoins	K6 K5
4.2	Ethereum: Ethereum – an overview - Ethereum network - Components of the Ethereum ecosystem	Construct the Ethereum ecosystem	K6
4.3	Ethereum Virtual Machine (EVM) - Smart contracts. - Blocks and blockchain - Wallets and client - Nodes and miners	Build the EVM Design the smart contracts, wallets and client for Ethereum	K6 K6
4.4	APIs, tools, and DApps - Supporting protocols - Programming languages.	List the tools for Ethereum Design the protocols for Ethereum List the programming languages for Ethereum	K3 K6 K4
V			
Development Tools and Frameworks, Use Cases & Security			
5.1	Development Tools and Frameworks : Languages - Compilers - Tools and libraries - Frameworks - Contract development and deployment - Layout of a Solidity source code file - Solidity language.	Develop the altcoins using tools and frameworks List the tools and compilers for bit coin programming	K6 K4
5.2	Use Cases: IoT – Government - Health - Finance – Media.	Design applications of blockchain in IoT, Government, Health, Finance	K6
5.3	Scalability and Other Challenges: Scalability - Privacy - Security - Other challenges.	Assess the security of block chain technology	K6

4. MAPPING SCHEME (CO, PO & PSO)

P21CS2:3	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PSO1	PSO2	PSO3	PSO4
CO1	H	H	H	H	H		M		H	H	H	H	
CO2	H	H	H	H	M		M		H	H	M	H	M
CO3	H	H	H	H	M		M		H	H	M	H	M
CO4	H	H	H	H	M		M		H	H	M	H	M
CO5	H	H	H	H	M		M		H	H	M	H	M
CO6	H	H	H	H	H	M	H	M	H	H	M	H	L

5. COURSE ASSESSMENT METHODS

DIRECT:

5. Continuous Assessment Test: T1, T2 (Theory & Practical Components): Closed Book
6. Open Book Test.
7. Cooperative Learning Report, Assignment, Group Presentation, Group Discussion, project Report, Field Visit Report, Poster Presentation, Seminar, Quiz (written).
8. Pre-Semester & End Semester Theory Examination

INDIRECT:

2. Course end survey (Feedback)

Name of the Course Coordinator: Prof. V. Bhuvaneshwari

CORE PRACTICAL III: ADVANCED ASP .NET PROGRAMMING LAB

SEMESTER: II
CREDITS: 4

CODE: P21CS2P3
HOURS/WEEK: 3

1. COURSE OUTCOMES

On successful completion of this course, students will be able to:

CO #	Course Outcomes	Level	Ex. No.
CO1	Design the Website using ASP .NET MVC Controls	K6	1-2
CO2	Establish the database connectivity for ASP .NET MVC web applications	K6	3-4
CO3	Design the web applications using RazorView and SignalIR Controls	K6	5-6
CO4	Develop the web application using AJAX Controls	K6	7
CO5	Design the webservices using XML	K6	8
CO6	Design the web applications using Azure Cloud and Xamarin	K6	9 - 10

2. SYLLABUS

1. Create a simple web Applications for students mark entry using ASP .NET MVC.
2. Create a Web Applications for library stock maintenance using Controllers, Actions, Filters, Selector and Views of ASP .NET MVC.
3. Create a web application for Employee details of company using Data models of ASP. NET MVC
4. Design a web applications for Employee details of company using Database connectivity of ASP .NET MVC
5. Create a web Applications for students mark entry using RazorView of ASP .NET MVC
6. Develop a simple chatbot for college admissions using SignalIR.
7. Develop a web application for college admission process using ajax controls
8. Develop an XML web service for college website.
9. Create a WebAPI for the college website and publish in Azure.
10. Develop a mobile app for college admission process using Xamarin

3. SPECIFIC LEARNING OUTCOMES

S.NO	EXERCISE	LEVEL
1	Create a simple web Applications for students mark entry using ASP .NET MVC.	K5
2	Create a Web Applications for library stock maintenance using Controllers, Actions, Filters, Selector and Views of ASP .NET MVC.	K5
3	Create a web application for Employee details of company using Data models of ASP. NET MVC	K5
4	Design a web applications for Employee details of company using Database connectivity of ASP .NET MVC	K6
5	Create a web Applications for students mark entry using RazorView of ASP .NET MVC	K6
6	Develop a simple chatbot for college admissions using SignalIR.	K6
7	Develop a web application for college admission process using ajax controls	K6
8	Develop an XML web service for college website.	K6
9	Create a WebAPI for the college website and publish in Azure.	K6
10	Develop a mobile app for college admission process using Xamarin	K6

4. MAPPING SCHEME (CO, PO & PSO)

P21CS2P3	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PSO1	PSO2	PSO3	PSO4
CO1	H	M	H	H	M	L	H	-	H	H	M	M	H
CO2	M	M	H	M	M	-	H	M	H	H	H	M	H
CO3	H	M	M	M	M	L	H	-	H	H	M	M	H
CO4	H	M	L	L	-	-	-	-	M	H	M	M	L
CO5	M	M	H	H	H	M	M	M	H	M	H	M	H
CO6	M	M	H	H	M	M	M	H	H	H	H	M	H

5. COURSE ASSESSMENT METHODS

DIRECT

1. Continuous Assessment Test: T1, T2 (Theory & Practical Components): Closed Book
2. Open Book Test.
3. Cooperative Learning Report, Assignment, Group Presentation, Group Discussion, project Report, Field Visit Report, Poster Presentation, Seminar, Quiz (written).
4. Pre-Semester & End Semester Theory Examination

INDIRECT

1. Course end survey (Feedback)

Name of the Course Coordinator: Dr. B. Karthikeyan

CORE PRACTICAL IV: REAL TIME WEB APP DEVELOPMENT LAB

SEMESTER: II
CREDITS: 4

CODE: P21CS2P4
HOURS/WEEK: 3

1. COURSE OUTCOMES

On successful completion of this course, students will be able to:

CO #	Course Outcomes	Level	Ex. No
CO1	Develop web application in Node JS, by understand the steps in designing a web page	K5	1-2
CO2	Design a webpage using server controls and manages the properties of server controls	K5	3
CO3	Develop an express .js code to create basic controls and model view controls	K5	4-6
CO4	Design web applications using controls of Angular js	K6	7-8
CO5	Develop the web application with ngModel and ngContorller	K6	9
CO6	Understand and implement the CRUD operations	K6	10

2. SYLLABUS

1. Create students web application for welcoming the user using Node.js
2. Create students profile using Node.Js
3. Implement the MVC pattern
4. Develop Hello world Express.js application
5. Build a registration form using Express.js
6. Develop a web application for Library that can perform the CRUD operations.
7. Build single page Coffee shop application using Angular JS.
8. Make a Shopping list using Angular JS which can add and delete items
9. Create a Angular JS application to implement the ngModel and ngController features.
10. Develop a web application for restaurant that can perform the CRUD operations using MEAN.

3. SPECIFIC LEARNING OUTCOMES

S.NO	EXERCISE	LEARNING OUTCOME	LEVEL
1	Create students web application for welcoming the user using Node.js	<ul style="list-style-type: none">• To build the code in node.js• To execute the code in command prompt.• To view the student web application form in the browser	K5
2	Create students profile using Node.js	<ul style="list-style-type: none">• To develop the student profile program with specific operations and save the file with the extension .js.• To execute the node.js program in the command prompt and display the student profile in the browser.	K5
3	Implement the MVC pattern	<ul style="list-style-type: none">• To recall element access with access specifications.• To create the code with specific functionality• To implement the Model View Controller to develop various controls• To execute the program in the browser	K5

4	Develop Hello world Express.js application	<ul style="list-style-type: none"> • To recall the steps to develop the program • To classify the modules in a program. • To develop Hello world program in Express.js and save it in the .js extension. • To execute the program in command prompt and visualize the result in the browser. 	K6
5	Build a registration form using Express.js	<ul style="list-style-type: none"> • To define and initialize variables. • To demonstrate element access • To inspect the operation of methods. • To develop the registration program with appropriate controls in Express.js. • To execute the program in command prompt and view the registration form in the browser. 	K6
6	Develop a web application for Library that can perform the CRUD operations.	<ul style="list-style-type: none"> • To construct the Program for Library management • To classify the various types of mangoose db operations. • To experiment with specific, CREATE, READ, UPDATE and DELETE operations • To make use these operations and visualize the result in the browser. 	K6
7	Build single page Coffee shop application using Angular JS.	<ul style="list-style-type: none"> • To define a Code. • To illustrate the operation of Angular Js. • To construct the program for coffee shop application using various controls. • To examine the code in command prompt. • To evaluate the result in web browser 	K6
8	Make a Shopping list using Angular JS which can add and delete items	<ul style="list-style-type: none"> • To develop the code for shopping list using Angular Js • To construct a code for adding and deleting an item • To examine the code and evaluate the result 	K6
9	Create a Angular JS application to implement the ngModel and ngController features.	<ul style="list-style-type: none"> • To create a construct an Angular Js application • To apply ngModel and ngController features. • To examine the code and evaluate the result 	K6
10	Develop a web application for restaurant that can perform the CRUD operations using MEAN.	<ul style="list-style-type: none"> • To construct the Program for Restaurant management • To classify the various types of mangoose db operations. • To experiment with specific CREATE, READ, UPDATE and DELETE operations using MEAN • To make use these operations and visualize the result in the browser. 	K6

Web links:

1. https://www.w3schools.com/nodejs/nodejs_mongodb_create_db.asp
2. <https://angularjs.org>
3. <https://github.com/gothinkster/node-express-realworld-example-app>
4. <https://github.com/gothinkster/angularjs-realworld-example-app>

4. MAPPING SCHEME (CO, PO & PSO)

P21CS2P4	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PSO1	PSO2	PSO3	PSO4
CO1	H	M	H	H	M	L	H	-	H	H	M	M	H
CO2	M	M	H	M	M	-	H	M	H	H	H	M	H
CO3	H	M	M	M	M	L	H	-	H	H	M	M	H
CO4	H	M	L	L	-	-	-	-	M	H	M	M	L
CO5	M	M	H	H	H	M	M	M	H	M	H	M	H
CO6	M	M	H	H	M	M	M	H	H	H	H	M	H

5. COURSE ASSESSMENT METHODS

DIRECT

5. Continuous Assessment Test: T1, T2 (Theory & Practical Components): Closed Book
6. Open Book Test.
7. Cooperative Learning Report, Assignment, Group Presentation, Group Discussion, project Report, Field Visit Report, Poster Presentation, Seminar, Quiz (written).
8. Pre-Semester & End Semester Theory Examination

INDIRECT

2. Course end survey (Feedback)

Name of the Course Coordinator: Dr. M. P. Anuradha

CORE VII: IMAGE AND VIDEO ANALYSIS

SEMESTER: III
CREDITS: 4

CODE: P21CS307
HOURS/WEEK: 5

1. COURSE OUTCOMES

On successful completion of this course, students will be able to:

CO #	Course Outcomes	Level	Unit
CO1	Apply the concepts of Image Transformation, Enhancement, segmentation, compression on the Monochrome and color images.	K3	I
CO2	Develop the algorithms for Image Enhancement, Segmentation, Compression, Video restoration, video enhancement and video motion estimation in MATLAB	K5	II
CO3	Develop a feature extraction algorithms for different types of images	K5	III
CO4	Develop the algorithms for Image Enhancement, Segmentation for Colour Images	K5	IV
CO5	Examine the video restoration, enhancement and motion estimation of videos	K5	V
CO6	Formulate the research problems on Computer Vision	K6	V

2. A. SYLLABUS

Unit I : Basics of Image Processing

Introduction and motivation to computer vision – Digital image representation and file formats – Basic image processing operations – Basics of MATLAB – Scripts, functions, graphics & visualization – Basic Image operations using image processing toolbox – Image manipulation using image processing toolbox – Light, Color, and Electromagnetic Spectrum – Image Acquisition – Image Digitization.

Unit II : Image Operations, Transformations and Histograms

Arithmetic Operations: Fundamentals and Applications – Logic Operations: Fundamentals and Applications – Mapping and Affine Transformations – Geometric Operations Using MATLAB – Other Geometric Operations and Applications – Gray-level (Point) Transformations – Other Linear and Non-linear Point Transformations – Computing and interpreting Image Histograms- Histogram Equalization – Histogram Modification Techniques.

Unit III : Image Filtering, Edge Detection and Image Segmentation

Convolution and Correlation – Image Smoothing (Low-pass Filters) – Image Sharpening (High-pass Filters) – Image Restoration: Noise and Noise Models - Noise Reduction Using Spatial-domain & Frequency Domain Techniques - Image Deblurring Techniques – Edge Detection: First-order and Second-order Derivative Edge Detection – Canny Edge Detector, Edge Linking and Boundary Detection Intensity-based Segmentation – Region-based Segmentation and Watershed Segmentation

Unit IV : Color Images, Compression and Feature Extraction, Representation

Color Models – Pseudocolor Image Processing – Full-color Image Processing – Basic Concepts of Image Compression – Lossless and Lossy Compression Techniques – Image Compression Standards and Quality Measures – Feature Vectors and Vector Spaces – Binary Object Features- Boundary Descriptors – Histogram-based (Statistical) and Texture Features.

Unit V : Video Processing: sampling and motion estimation

Monochrome Analog Video and Color Video – Digital Video Basics – Digital Video Formats and Standards – Video Processing in MATLAB – Video Sampling and Basics - Standards Conversion – Fundamentals of Motion Estimation and Motion Compensation – General Methodologies in Motion Estimation – Motion Estimation Algorithms – Video Enhancement and Noise Reduction.

B. TOPICS FOR SELF STUDY

- Computer Vision using Machine Learning
- Computer Vision using Deep Learning
- Image Classification Model using OpenCV
- Image Classification using Watson

C. TEXT BOOKS

1. Oge Marques, *“Practical Image and Video Processing Using MATLAB”*, Wiley Publications, 2011.

D. REFERENCES

1. Mark S. Nixon and Alberto S. Aguado, *“Feature Extraction & Image Processing for Computer Vision”*, 3ed, Academic Press, 2012.
2. Gonzalez, R.C. Woods, *“Digital Image Processing”*, 4th Edition, PHI publications, 2015.

E. WEB LINKS

- <https://www.udacity.com/course/introduction-to-computer-vision--ud810>
- <https://www.coursera.org/professional-certificates/applied-artificial-intelligence-ibm-watson-ai>
- https://onlinecourses.nptel.ac.in/noc20_cs88/preview
- <https://cloudxlab.com/course/99/computer-vision>

3. SPECIFIC LEARNING OUTCOMES

Unit	Topic	Topic Learning Outcomes	Level
By the end of each topic of a unit, students will be able to			
I	Basics of Image Processing		
1.1	Introduction and motivation to computer vision	Explain the components of Human Vision Systems	K2
		Demonstrate the basic concepts of Computer vision	K2
1.2	Digital image representation and file formats	Categorize the different types for Digital Image Representation	K4
1.3	Basic image processing operations	Explain the different types of Image Processing operations	K5
1.4	Basics of MATLAB – Scripts, functions, graphics & visualization	Design the Matlab programming for different types of problems	K6
1.5	Basic Image operations and Image manipulation using image processing toolbox (IPT)	Develop the programs for Image processing algorithms using IPT	K6
1.6	Light, Color, and Electromagnetic Spectrum	Summarize the basic concepts of Optics	K2
1.7	Image Acquisition and Image Digitization	Examine the different types sensors for acquisition of images	K4
		Explain the concepts of Image Digitization	K5
II	Image Operations, Transformations and Histograms		
2.1	Arithmetic Operations: Fundamentals and Applications	Identify the type of arithmetic operations on images	K2
		Develop the program for logical operations using Matlab functions	K6
2.2	Logic Operations: Fundamentals and Applications	Categorize the logical Operations	K3
		Develop the program for logical operations using Matlab functions	K6

2.3	Geometric Operations: – Mapping and Affine Transformations	Explain the different types of mapping (Transformations)	K5
2.4	Geometric Operations Using MATLAB	List the matlab functions of Geometric Operations Design the Matlab application for Geometric Operations	K4 K6
2.5	Other Geometric Operations and Applications	Discuss about the features of wrapping and other non-linear transformations	K6
2.6	Gray-level (Point) Transformations – Other Linear and Non-linear Point Transformations	Demonstrate the Point Transformation on grayscale images	K2
		Assess the performance of the Linear transformations on different images	K5
		Compare the non-linear transformations over linear transformations.	K4
2.7	Histogram Processing: Computing and interpreting Image Histograms	Construct the histogram for the different images	K6
		Interpret the histograms of different images	K5
2.8	Histogram Equalization	Elaborate the steps involved in the Histogram Equalization	K6
2.9	Histogram Modification Techniques	Explain the procedure of Direct histogram specifications	K5
		List the other histogram modification techniques	K5
III	Image Filtering, Edge Detection and Image Segmentation		
3.1	Neighbourhood Processing: Convolution and Correlation – Image Smoothing (Low-pass Filters) – Image Sharpening (High-pass Filters)	Formulate the convolution and correlation operations for filter design	K6
		Design the Low-pass filters for the images	K6
		Design the High-pass filters for the images	K6
		Examine the filtering techniques on different images	K5
3.2	Image Restoration: Noise and Noise Models - Noise Reduction Using Spatial-domain & Frequency Domain Techniques	Explain the Noise Model	K5
		Develop the Noise filters for the spatial domain	K6
		Develop the Noise -pass filters for the frequency Domain	K6
		Construct the filtering techniques using matlab	K6
3.3	Image Deblurring Techniques	Develop the algorithms for Deblurring	K6
3.4	Edge Detection: First-order and Second-order Derivative Edge Detection	Distinguish the First-order and Second-order Derivative Edge Detection	K5
		List the Matlab functions for the edge detection methods	K4
		Compare the different types of Edge Detection methods	K5
3.5	Canny Edge Detector	Explain the working flow of Canny Edge Detector	K5
3.6	Edge Linking and Boundary Detection	Examine the Edge Linking and Boundary Detection methods for different images	K5
3.7	Image Segmentation:	Demonstrate the concepts of Image segmentation	K2

	Intensity-based Segmentation	Categorize the methods of Intensity based segmentations	K4
3.8	Region-based Segmentation	Discuss the features of Region-based Segmentation methods	K6
3.9	Watershed Segmentation	Compare the Watershed Segmentation with other methods	K5
IV	Color Images, Compression and Feature Extraction, Representation		
4.1	Color Image Processing: Color Models	Discuss the concepts of colour models List the matlab functions for colour image processing	K6 K4
4.2	Pseudocolor Image Processing	Explain the characteristics of pseudocolor processing	K5
4.3	Full-color Image Processing	Compare the full color image processing over pseudocolor processing	K5
4.4.	Image Compression : Basic Concepts of Image Compression	Outline concepts of Image Compressions	K2
4.5	Lossless and Lossy Compression Techniques	Analyse the different methods of Lossless Compression	K4
		Classify the different methods of Lossy Compression	K4
4.6	Image Compression Standards and Quality Measures	Estimate the compression methods using Quality Measures	K5
4.7	Feature Extraction: Feature Vectors and Vector Spaces	Summarize the concepts of feature extraction	K2
		Illustrate the feature extraction using Matlab functions	K2
4.8	Binary Object Features	Compile the binary object features	K6
4.9	Boundary Descriptors	Construct the Boundary descriptors for the images	K6
4.10	Histogram-based (Statistical) and Texture Features	Compare the Histogram-based (Statistical) and Texture Features	K5
V	Video Processing: sampling and motion estimation		
5.1	Video Fundamentals: Monochrome Analog Video and Color Video	Outline the concepts of Video fundamentals	K2
		Distinguish the Monochrome and colour video	K4
5.2	Digital Video Formats and Standards	Analyse the different types of video formats and standards	K4
5.3	Digital Video Basics	Explain how the analog video converted to digital video	K5
5.4	Video Processing in MATLAB	Develop the matlab program for video processing	K6
5.5	Video Sampling Rate & Standards Conversion: Video Sampling and Basics	Examine the sampling rate conversion	K5
5.6	Standards Conversion	Elaborate the main practical aspects involved in converting a video sequence from one format to another.	K6
		Analyze steps are involved in PAL to NTSC (and vice versa) standard conversion	K4

5.7	Fundamentals of Motion Estimation and Motion Compensation – General Methodologies in Motion Estimation – Motion Estimation Algorithms	Demonstrate the motion estimation (ME) and its properties Design techniques and algorithms can be used to estimate motion within a video sequence	K2 K6
5.8	Video Enhancement and Noise Reduction.	Develop the techniques can be used to filter a video sequence Formulate the research problems related to Video processing List the matlab functions to be used for video processing	K6 K6 K2

4. MAPPING SCHEME (CO, PO & PSO)

P21CS307	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PSO1	PSO2	PSO3	PSO4
CO1	H	H	H	H	H		M		H	H	H	H	
CO2	H	H	H	H	M		M		H	H	M	H	M
CO3	H	H	H	H	M		M		H	H	M	H	M
CO4	H	H	H	H	M		M		H	H	M	H	M
CO5	H	H	H	H	M		M		H	H	M	H	M
CO6	H	H	H	H	H	M	H	M	H	H	M	H	L

5. COURSE ASSESSMENT METHODS

DIRECT

1. Continuous Assessment Test: T1, T2 (Theory & Practical Components): Closed Book
2. Open Book Test.
3. Cooperative Learning Report, Assignment, Group Presentation, Group Discussion, project Report, Field Visit Report, Poster Presentation, Seminar, Quiz (written).
4. Pre-Semester & End Semester Theory Examination

INDIRECT

1. Course end survey (Feedback)

Name of the Course Coordinator: Dr. P. S. Eliahim Jeevaraj

CORE VIII: MACHINE LEARNING

SEMESTER: III
CREDITS: 5

CODE: P21CS308
HOURS/WEEK: 5

1. COURSE OUTCOMES

On successful completion of this course, students will be able to:

CO #	Course Outcomes	Level	Unit Covered
CO1	Elaborate the basics of building machine learning systems, to implement the perceptron learning model and the convergence.	K6	I
CO2	Demonstrate the sci-kit learning method and SVM for solving linear and non-linear problems.	K4	II
CO3	Apply the Dimensionality Reduction and model evaluation for different data set	K5	III
CO4	Compare the different cluster analysis algorithm	K4	IV
CO5	Develop the neural network for the various dataset to classify them	K6	IV
CO6	Design the web application using Scikit FLASH	K6	V

2. A. SYLLABUS

UNIT – 1 Machine Learning Algorithm - Training

Three different types of Machine Learning Algorithms-Predictions with supervised learning-Predicting with Class labels-Regression for predicting continuous outcomes-Introduction to basic terminology and notations used in Machine Learning

Building Machine Learning Systems: Preprocessing, Training, Selection and Evaluation Of Model-Installing Python Package, Artificial Neurons: A glimpse into the early history of Machine Learning-Implementation of Perceptron learning algorithm in Python-Adaptive linear neurons and the Convergence of Learning-Implementation of Adaptive linear neuron in Python.

UNIT – 2 Machine Learning Classifier

Training a perceptron via Scikit-Learn-Modeling class probabilities through logistic regression-Support vector machine for maximum margin classification-Solving Non-Linear problems using a kernel SVM

Learning Decision Tree-A Lazy learning Algorithms – K-Nearest Neighbors -Data Preprocessing: Dealing with missed and Categorical data-Data Preprocessing: Dealing with Data set Testing and Training-Selection of Meaningful features-Selection of meaningful sparse solutions with L1 regularization

UNIT – 3 Dimensionality Reduction and Model Evaluation

Unsupervised Dimensionality reduction using P.C.A [Principal Component Analysis]-Supervised data compression via Linear Discriminant analysis-Using Kernel principal component analysis for N-L-M-Streamlining workflows with pipeline-Using K-Fold cross-validation to assess model performance

Debugging algorithms with learning and validation-Different performance evaluation metrics-Implementing simple majority vote classifier-Tuning the ensemble classifier-Weak Learners Vs Adaptive Boosting.

UNIT – 4 Cluster Analysis

Grouping objects by similarity using K-Means-Organizing Clusters as a hierarchical tree. -Training Artificial Neural Network for image recognition-Modeling Complex functions with Artificial Neural Network-Classifying Hand Written Digits

Training an Artificial Neural Network-Developing initiation for Back Propagation-Debugging Neural Network with Gradient Checking-Convergence in Neural Network-Architecture of Neural Network

UNIT – 5 Embedding a Machine Learning Model into a Web Application

Serializing fitted Scikit-learn estimators-Setting up SQL-LITE database for data storage- Developing a Web application with FLASK-First FLASK Web application-Form Validation and Rendering-Turning the movie classifier into a web application-Deploying the web application to a public server- Updating the movie review Classifier

B. TOPICS FOR SELF STUDY

- Data Input and Preprocessing with Tensorflow
- Machine Learning Model Building
- Prediction with Tensorflow
- Monitoring and evaluating models using Tensorboard

C. TEXT BOOKS

1. Sebastian Raschka, Python Machine Learning, First Edition, [PACKT] , 2015.
2. Luis Pedro Coelho, Willi Richert, Building Machine Learning System with Python, Second Edition [PACKT], 2015
3. Gavin Hakeking, Mastering Machine Learning with Sci-kit Learn, Second Edition [PACKT], 2017

D. REFERENCE BOOKS

1. Joel Grus, Data Science from Scratch, First Edition, O'Reilly, 2015
2. Gavin hakeking, Mastering machine learning with scikit-learn, First Edition, [PACKT] , 2014

E. WEB LINKS

- <https://nptel.ac.in/noc/courses/noc20/SEM1/noc20-cs44/>
- <https://www.coursera.org/learn/machine-learning>
- <https://developers.google.com/machine-learning/crash-course>
- <https://www.udemy.com/topic/machine-learning/>

3. SPECIFIC LEARNING OUTCOMES

Unit	Course Content	Learning Outcomes	Level
I	Machine Learning Algorithm – Training		
1.1	Types of Machine Learning Algorithms	Description of basic types of Machine learning algorithms.	K1
1.2	Making Predictions with supervised learning	Demonstrate the Supervised learning method to make predictions.	K2
1.3	Predictions of Class Labels	Apply the principles of Predicting Class labels and their meanings.	K3
1.4	Continuous outcome prediction using Regression Learning	Experiment with the outcome of Regression Learning method.	K3
1.5	Terminologies and Notations used in Machine Learning	Label the basic symbols and terms used in Machine Learning	K1
1.6	Building Machine Learning Systems	Categorize the Machine Learning system	K4
1.7	Installation of Python Package	Apply the installation tasks of a Python Package.	K3
1.8	Implementing Perceptron model in Python	Building Perceptron Model using Python.	K6
1.9	Convergence of Learning	Forumulate the Convergence process in Learning	K6
1.10	Implementation of Adaptive linear neuron in Python	Interpret the implementation of Adaptive Linear neuron in Python	K5

II	Machine Learning Classifier		
2.1	Training a perceptron via Scikit-Learn	Make use of Scikit-Learn in training of a Perceptron	K3
2.2	Probabilistic Modeling through Logistic Regression	Evaluate Probabilistic Modeling through Logistic Regression	K5
2.3	Support vector machine for maximum margin classification	Analysis of margin classification using Support Vector Machine	K4
2.4	Solving Non-Linear problems using a kernel SVM	Experiment the problem of solving Non-Linear Problems using SVM.	K3
2.5	Learning Decision Tree	Classification of Learning of Decision Tree	K2
2.6	K-Nearest Neighbor – A Lazy learning Algorithms	Examine the functions of KNN Algorithm	K4
2.7	Data Pre processing: Dealing with missed and Categorical data	Organize the ways to handling missed and Categorical data	K3
2.8	Data Preprocessing: Dealing with Data set Testing and Training	Contrast training and testing the datasets	K4
2.9	Selection of Meaningful features	Interpret the selection of Features	K5
2.10	Selection of meaningful sparse solutions with L1 regularization	Identify L1 regularization in sparse solutions.	K3
III	Dimensionality Reduction and Model Evaluation		
3.1	Unsupervised Dimensionality reduction using Principal Component Analysis	Applying Unsupervised Dimensionality reduction using PCA	K3
3.2	Supervised data compression via Linear Discriminant analysis	Compare and contrast LDA and Supervised data Compression	K4
3.3	Using Kernel principal component analysis for N-L-M	Define the effects of Kernel PCA over N-L-M	K1
3.4	Stream Lining workflows with pipeline	Demonstrate the Stream lining with the pipeline.	K2
3.5	Using K-Fold cross-validation to assess model performance	Modeling performance using K-Fold Validation	K3
3.6	Debugging algorithms with learning and validation	Extend debugging in learning and validation	K2
3.7	Different performance evaluation metrics	List the performance evaluation metrics	K1
3.8	Implementing simple majority vote classifier	Compile the working principle of simple majority vote classifier	K6
3.9	Tuning the ensemble classifier	Measure classifier tuning method	K5
3.10	Weak Learners Vs Adaptive Boosting	Compare and contrast Weak learners and Adaptive Boosting	K5
IV	Cluster Analysis		
4.1	Grouping objects by similarity using K-Means	Relate the grouping of objects by K-Means	K1
4.2	Organizing Clusters as a hierarchical tree.	Relating the organization of clusters as a hierarchical tree.	K4
4.3	Training Artificial Neural Network for image recognition	Experiment with Neural network to recognize an image	K3
4.4	Modeling Complex functions with Artificial Neural Network	Demonstrate the complex function modelling with ANN	K2
4.5	Classifying Hand Written Digits	Organize the Handwritten symbols using classification	K3

4.6	Training an Artificial Neural Network	Recall on training ANN	K1
4.7	Developing initiation for Back Propagation	Categorize the initiation process in Back propagation	K4
4.8	Debugging Neural Network with Gradient Checking	Utilize Gradient check in NN for debugging.	K3
4.9	Convergence in Neural Network	Spell the convergence process in NN	K1
4.10	Architecture of Neural Network	Define the Architecture of NN	K1
V	Embedding a Machine Learning Model into a Web Application		
5.1	Serializing fitted Scikit-learn estimators	Organizing the Scikit-learn estimators	K3
5.2	Setting up SQL-LITE database for data storage	Illustrate the SQL-LITE database storage set up	K2
5.3	Developing a Web application with FLASK	Build an Web application with FLASK	K3
5.4	First FLASK Web application	Development of First FLASK Application	K3
5.5	Form Validation and Rendering	Demmonstrate the process of Validation and Rendering in GORMS	K2
5.6	Turning the movie classifier into a web application	Experiment with the movie classifier in web application	K2
5.7	Deploying the web application to a public server	Justify the web application deployment	K5
5.8	Updating the movie review Classifier	Examine the updation in movie review classifier	K4

4. MAPPING SCHEME (CO, PO & PSO)

P21CS308	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PSO1	PSO2	PSO3	PSO4
CO1	H	H	H	H	H		M		H	H	H	H	
CO2	H	H	H	H	M		M		H	H	M	M	M
CO3	H	H	H	M	M		M		H	M	M	H	M
CO4	H	H	H	M	M		M		H	H	M	H	M
CO5	H	H	M	H	M		M		H	L	M	H	M
CO6	H	M	H	H	H	M	H	M	H	H	M	M	M

5. COURSE ASSESSMENT METHODS

DIRECT

1. Continuous Assessment Test: T1, T2 (Theory & Practical Components): Closed Book
2. Open Book Test.
3. Cooperative Learning Report, Assignment, Group Presentation, Group Discussion, project Report, Field Visit Report, Poster Presentation, Seminar, Quiz (written).
4. Pre-Semester & End Semester Theory Examination

INDIRECT

1. Course end survey (Feedback)

Name of the Course Coordinator: Dr. B. Satheesh Kumar

CORE IX: COGNITIVE COMPUTING**SEMESTER: III**
CREDITS: 4**CODE: P21CS309**
HOURS/WEEK: 5

1. COURSE OUTCOMES

On successful completion of this course, students will be able to:

CO. #	Course Outcomes	Level	Unit
CO1	Design and evaluate the cognitive system	K6	I
CO2	Develop the cognitive system using NLP	K6	II
CO3	Evaluate the role of big data in cognitive systems	K5	III
CO4	Design the business implications of cognitive computing	K4	IV
CO5	Build the application of cognitive computing for precision medicine	K6	V
CO6	Develop the applications of cognitive in Healthcare	K6	V

2. A. SYLLABUS**UNIT I : Foundation of Cognitive Computing**

Foundation of Cognitive Computing: cognitive computing as a new generation, the uses of cognitive systems, system cognitive, gaining insights from data, Artificial Intelligence as the foundation of cognitive computing, understanding cognition. Design Principles for Cognitive Systems: Components of a cognitive system, building the corpus, bringing data into cognitive system, machine learning, hypotheses generation and scoring, presentation and visualization services

UNIT II: Natural Language Processing

Natural Language Processing in support of a Cognitive System: Role of NLP in a cognitive system, semantic web, Applying Natural language technologies to Business problems. Representing knowledge in Taxonomies and Ontologies: Representing knowledge, Defining Taxonomies and Ontologies, knowledge representation, models for knowledge representation, implementation considerations.

UNIT III: Big Data and Cognitive Computing

Relationship between Big Data and Cognitive Computing : Dealing with human-generated data, defining big data, architectural foundation, analytical data warehouses, Hadoop, data in motion and streaming data, integration of big data with traditional data. Applying Advanced Analytics to cognitive computing: Advanced analytics is on a path to cognitive computing, Key capabilities in advanced analytics, Using advanced analytics to create value, Impact of open source tools on advanced analytics.

UNIT IV: Building a Cognitive Application

The Business Implications of Cognitive Computing : Preparing for change ,advantages of new disruptive models , knowledge meaning to business, difference with a cognitive systems approach , meshing data together differently, using business knowledge to plan for the future , answering business questions in new ways , building business specific solutions , making cognitive computing a reality , cognitive application changing the market. The process of building a cognitive application: Emerging cognitive platform, defining the objective, defining the domain, understanding the intended users and their attributes, questions and exploring insights, training and testing

UNIT V: Application of Cognitive Computing

Building a cognitive health care application: Foundations of cognitive computing for healthcare, constituents in healthcare ecosystem, learning from patterns in healthcare Data, Building on a foundation of big data analytics, cognitive applications across the health care eco system, starting with a cognitive application for healthcare, using cognitive applications to improve health and wellness, using a cognitive application to enhance the electronic medical record Using cognitive application to improve clinical teaching.

B. TOPICS FOR SELF STUDY

- Deep Learning for Cognitive Computing
- Knowledge based AI : Cognitive Computing
- Cognitive computing for cloud

C. TEXT BOOKS

1. Judith Hurwitz, Marcia Kaufman, Adrian Bowles, “Cognitive Computing and Big Data Analytics”, Wiley Publications, 2015

D. REFERENCE BOOKS

1. Kai Hwang, Min Chen, Big-Data Analytics for Cloud, IoT and Cognitive Computing [1 ed.], Wiley, 2017

E. WEB LINKS

- <https://cognitiveclass.ai/>
- <https://www.edx.org/course/implementation-strategies-ai-and-cognitive-computing>
- <https://www.udemy.com/course/ibm-watson-for-artificial-intelligence-cognitive-computing/>

3. SPECIFIC LEARNING OUTCOMES

Unit	Course Content	Learning Outcomes	Level
I	Foundation of Cognitive Computing		
1.1	Foundation of Cognitive Computing: cognitive computing as a new generation	Explain the features of Cognitive Computing	K5
1.2	Uses of cognitive systems	List the Uses of cognitive systems	K4
1.3	System cognitive, Gaining insights from data, Artificial Intelligence as the foundation of cognitive computing, understanding cognition	Explain the role of AI in cognitive computing Understand the cognition concepts to build the system	K5 K4
1.4	Design Principles for Cognitive Systems: Components of a cognitive system, building the corpus, bringing data into cognitive system	Design the cognitive systems Integrate the data into cognitive system	K6 K5
1.5	Design Principles for Cognitive Systems: Machine learning, hypotheses generation and scoring, presentation and visualization services	Design the cognitive system using Machine Learning Assess the system by visualization	K6 K6
II	Natural Language Processing		
2.1	Natural Language Processing in support of a Cognitive System: Role of NLP in a cognitive system, semantic web, Applying Natural language technologies to Business problems.	Build the cognitive system using NLP Design the cognitive system for business problems	K6 K6
2.2	Representing knowledge in Taxonomies and Ontologies: Representing knowledge, Defining Taxonomies and Ontologies	Understand the ontology of web Explain the knowledge taxonomy	K5 K4
2.3	knowledge representation, models for knowledge representation, implementation considerations.	Construct the models for knowledge representation	K6
III	Big Data and Cognitive Computing		
3.1	Relationship between Big Data and Cognitive Computing : Dealing with	Explain the Big Data Architecture in Cognitive System	K4 K6

	human-generated data, defining big data, architectural foundation	Design the Big data based cognitive system	
3.2	Analytical data warehouses, Hadoop, data in motion and streaming data, integration of big data with traditional data.	Analyze the data in cognitive system using Hadoop	K5
3.3	Applying Advanced Analytics to cognitive computing: Advanced analytics is on a path to cognitive computing, Key capabilities in advanced analytics	Develop the analysis tool for analysis of data in cognitive system	K6
3.4	Using advanced analytics to create value, Impact of open source tools on advanced analytics.	List the open source tools for advanced analytics	K4
IV	Building a Cognitive Application		
4.1	The Business Implications of Cognitive Computing: Preparing for change, advantages of new disruptive models, knowledge meaning to business, difference with a cognitive systems approach, meshing data together differently.	Design the business implications of cognitive computing Develop the disruptive models for cognitive systems Mesh the data from different sources for cognitive systems.	K6 K6 K5
4.2	Using business knowledge to plan for the future, answering business questions in new ways, building business specific solutions, making cognitive computing a reality, cognitive application changing the market	Develop the business knowledge system Assess the market trend using cognitive systems	K6 K6
4.3	The process of building a cognitive application: Emerging cognitive platform, defining the objective, defining the domain	Build the cognitive applications Defines the cognitive platform	K6 K3
4.4	Understanding the intended users and their attributes, questions and exploring insights, training and testing	Understand the users of cognitive system	K3
V	Application of Cognitive Computing		
5.1	Building a cognitive health care application: Foundations of cognitive computing for healthcare	Develop the cognitive healthcare applications	K6
5.2	Constituents in healthcare ecosystem, learning from patterns in healthcare Data	Build the ecosystem for Health care data	K6
5.3	Building on a foundation of big data analytics, cognitive applications across the health care eco system, starting with a cognitive application for healthcare	Analyze the healthcare big data Design cognitive system for healthcare applications	K5 K6
5.4	Using cognitive applications to improve health and wellness, using a cognitive application to enhance the electronic medical record Using cognitive application to improve clinical teaching.	Explain the cognitive applications for healthcare	K4

4. MAPPING SCHEME (CO, PO & PSO)

P21CS309	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PSO1	PSO2	PSO3	PSO4
CO1	H		L		L								
CO2	H	M	M	M	H	H							
CO3	H	H	H	H	M	L	M		H	M	H	L	
CO4	H	H	M	L	H			H	M	H	M	M	H
CO5	H			H	H		H	H	H	H	H	M	H
CO6	H		H	H	H		H	H	H	H	H	H	H

5. COURSE ASSESSMENT METHODS

DIRECT:

1. Continuous Assessment Test: T1, T2 (Theory & Practical Components): Closed Book
2. Open Book Test.
3. Assignment, Group Presentation, Group Discussion, Seminar, Quiz (written).
4. Pre-Semester & End Semester Theory Examination

INDIRECT:

1. Course end survey (Feedback)

Name of the Course Coordinator: Dr. P.S. Eliahim Jeevaraj

ELECTIVE 4a: DESIGN AND IMPLEMENTATION OF DATA WAREHOUSE

SEMESTER: III
CREDITS: 4

CODE: P21CS3:4
HOURS/WEEK: 5

1. COURSE OUTCOMES

On successful completion of this course, students will be able to:

CO #	Course Outcomes	Level	Unit Covered
CO1	Explain the fundamental concepts of data warehouse	K2	I
CO2	Illustrate the model of data warehouse architecture	K4	II
CO3	Impart the concepts of analytical processing operation	K4	III
CO4	Determine data warehouse design concepts	K3	IV
CO5	Create queries to get the desired results from data warehouse	K5	V
CO6	Evaluate Extraction, Transformation and Loading techniques in order to get information from the internal and external sources	K6	V

2. A. SYLLABUS

Unit I : Data warehouse fundamental concepts

Hierarchies – Measures – OLAP operations – Data warehouses – Data warehouse architecture: Back-End tier, Data Warehouse tier – OLAP tier, Front-end tier – Variations of the architectures – Data warehouse Design – Microsoft SQL Server Tools – Pentaho Business Analytics.

Unit II : Conceptual Data warehouse design

Conceptual modeling of data warehouses – Balanced Hierarchies and Unbalanced hierarchies – Generalized Hierarchies – Parallel hierarchies – Alternative – hierarchies – Parallel hierarchies – Non strict hierarchies – Facts with Multiple – granularities – Many to many dimensions – Querying the Northwind Cube Using the OLAP Operations.

Unit III : Logical Data warehouse Design

Logical Modeling of data warehouses – Relational data warehouse design – Relational implementation of the conceptual model – Time Dimensions – Logical representation of hierarchies – Advanced modeling aspects – Slowly changing dimensions – SQL/OLAP operations : Data Cube – Rollup, Cube and Grouping sets – Window Functions.

Unit IV : Querying Data warehouses

Tuples and Data sets, Basic Queries – Slicing, Navigation – Cross join, Sub-queries – Calculated Members and Named sets – Relative Navigation – Time Series Functions – Filtering – Sorting – Top and Bottom analysis – Aggregation functions.

Unit V : Physical Data warehouse design and Extraction, Transformation and Loading

Physical modeling of data warehouses – Materialized views – Data Cube Maintenance – PipeSort Algorithms – Cube Size Estimation – Evaluation of Star queries – Data warehouse partitioning – Business Process Modeling Notation – Conceptual ETL Design using BPMN – Conceptual design of the Northwind ETL Process.

B. TOPICS FOR SELF STUDY

- Design and Build a Data Warehouse for Business Intelligence Implementation
- Data integration workflows and analytical queries
- Materialized views, Dashboards and Visualizations
- BI tools

C. TEXT BOOK

1. Alejandro Vaisman, Esteban Zimanyi, “*Data Warehouse Systems*”, Springer – 2014, ISBN 978-3-642-54654-9

D. REFERENCE BOOK

1. William H. Inmon, “*Building the Data Warehouse*”, Wiley Publishing Inc, 2005, 4th Edition, ISBN-13: 978-0-7645-9944-6

E. WEB LINKS

- <https://www.coursera.org/learn/data-warehouse-bi-building>
- <https://www.coursera.org/specializations/data-warehousing>
- <https://www.edx.org/learn/data-warehouse>

3. SPECIFIC LEARNING OUTCOMES

Unit	Topics	Learning Outcomes	Level
I	Data warehouse fundamental concepts		
1.1	Hierarchies	Recognizing the data warehouse Hierarchies	K1
1.2	Measures	Identifying the specific measures Of Data warehouse Concepts	K1
1.3	OLAP operations	Classifying the various OLAP Operations	K2
1.4	Data warehouses	Summarizing the Data warehouse Concepts	K2
1.5	Data warehouse architecture	Demonstrating the Data warehouse Architecture	K2
1.6	Back-End tier	Categorizing the Data warehouse Tiers	K2
1.7	OLAP tier		
1.8	Data Warehouse tier		
1.9	Front-end tier		
1.10	Variations of the architectures	Relating the variations of the architecture	K2
1.11	Data warehouse Design	Explain the Data warehouse Design	K2
1.12	Parallel hierarchies	Discuss the various tools of MS-SQL Server	K2
1.13	Pentaho Business Analytics	Extend the Business Analytics with Pentaho Model	K2
II	Conceptual Data warehouse design		
2.1	Conceptual modelling of data	Describing the conceptual modelling of data	K1
2.2	Balanced Hierarchies and Unbalanced hierarchies	Contrasting Balanced hierarchies with Unbalanced hierarchies	K2
2.3	Generalized Hierarchies	Relating the types of Hierarchies	K2
2.4	Alternative hierarchies		
2.5	Parallel hierarchies		
2.6	Non strict hierarchies		
2.7	Facts with Multiple granularities	Determining the facts with multiple granularities	K3
2.8	Many to many dimensions	Discussing many to many Dimensions	K2
2.9	Querying the North wind Cube Using the OLAP Operations	Applying the queries using OLAP Operations	K3
III	Logical Data warehouse Design		
3.1	Logical Modelling of data warehouses	Extend the Logical Model in Data warehouse	K2
3.2	Relational data warehouse design	Constructing Relational data warehouse design	K3
3.3	Relational implementation of the conceptual model	Structuring the conceptual model in relational database design	K4
3.4	Time Dimensions	Correlating the time dimensions	K4

3.5	Logical representation of hierarchies	Prioritizing the hierarchies based on the logical representation	K4
3.6	Advanced modelling aspects	Illustrating the Advanced Modelling Aspects	K4
3.7	Slowly changing dimensions	Accessing the Slowly Changing Dimensions	K5
3.8	SQL/OLAP operations	Validating SQL/OLAP Operations	K5
3.9	Data Cube	Designing the Data Cube	K4
3.10	Rollup, Cube and Grouping sets	Estimating Rollup, Cube and Grouping Sets	K4
3.11	Window Functions	Contrasting the various Window Functions	K4
IV	Querying Data warehouses		
4.1	Tuples and Data sets, Basic Queries	Defining the tuples datasets and basic queries	K1
4.2	Slicing, Navigation	Indicating Slicing and Navigation concept	K2
4.3	Cross join, Sub-queries	Examine the cross join and sub queries	K3
4.4	Calculated Members and Named sets	Operating Calculated Members and Named sets	K3
4.5	Relative Navigation	Executing Relative navigational Function	K3
4.6	Time Series Functions	Classifying the various time series Functions	K4
4.7	Filtering	Compare Filtering with Sorting	K4
4.8	Sorting		
4.9	Top and Bottom analysis	Construct the top and bottom analysis	K3
4.10	Aggregation functions	Integrate the aggregate Functions	K4
V	Physical Data warehouse design and Extraction, Transformation and Loading		
5.1	Physical modelling of data warehouses	Describing Physical modelling of data warehouse	K2
5.2	Materialized views	Computing the materialized views	K3
5.3	Data Cube Maintenance	Summarizing the concept of data cube maintenance	K2
5.4	Pipe Sort Algorithms	Determine the pipe sort algorithms	K3
5.5	Cube Size Estimation	Estimating the cube size	K4
5.6	Evaluation of Star queries	Checking the evaluation of queries	K5
5.7	Data warehouse partitioning	Composing the partition of data warehouse	K6
5.8	Business Process Modelling Notation	Simulating the Business Process Modelling Notation	K6
5.9	Conceptual ETL Design using BPMN	Formulating Conceptual ETL Design using BPMN	K6
5.10	Conceptual design of the Northwind ETL Process.	Adapting the conceptual design with ETL Process	K6

4. MAPPING SCHEME (CO, PO & PSO)

P21CS4:1	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PSO1	PSO2	PSO3	PSO4
CO1	H	M	M	L	-	-	-	-	-	H	-	L	-
CO2	H	L	H	L	-	-	-	-	M	-	H	-	-
CO3	M	M	M	H	M	L	H	M	L	H	H	M	H
CO4	M	H	M	L	M	-	-	-	-	H	M	-	L
CO5	H	M	H	M	H	H	M	L	M	M	H	M	L
CO6	M	L	L	M	L	M	H	H	H	H	H	H	M

5. COURSE ASSESSMENT METHODS

DIRECT

1. Continuous Assessment Test: T1, T2 (Theory & Practical Components): Closed Book
2. Open Book Test.
3. Cooperative Learning Report, Assignment, Group Presentation, Group Discussion, project Report, Field Visit Report, Poster Presentation, Seminar, Quiz (written).
4. Pre-Semester & End Semester Theory Examination

INDIRECT

1. Course end survey (Feedback)

Name of the Course Coordinator: Dr. M. P. Anuradha

CORE PRACTICAL VI: IMAGE AND VIDEO ANALYTICS LAB

SEMESTER: III
CREDITS: 3

CODE: P21CS3P5
HOURS/WEEK: 5

1. COURSE OUTCOMES

On successful completion of this course, students will be able to:

CO #	Course Outcomes	Level
CO1	Practice fundamental concepts of images and image Transformation	K4
CO2	Analyze the transformation, filtering, segmentation and edge detection methods	K5
CO3	Construct the Matlab applications for image compression techniques	K6
CO4	Develop the matlab programs for motions and objects from videos	K5
CO5	Design the applications for colour image processing operations	K6
CO6	Formulate the research problems on Computer Vision	K6

2. SYLLABUS

1. Develop an application for Image Transformation
2. Develop an application for Image Enhancement
3. Develop an application for Image Restoration
4. Develop an application for Edge Detection
5. Develop an application for Image Segmentation
6. Develop an application for Image Compression
7. Develop an application for Color Image Processing
8. Develop an application for Video enhancement
9. Develop an application for Video Restoration
10. Develop an application for Motion detection and Segmentation in videos

3. SPECIFIC LEARNING OUTCOMES

Topic	Topic Learning Outcomes	Level
Image Transformation	Apply the image transformation on grayscale images	K4
Image Enhancement	Design the applications for image enhancement techniques	K6
Image Restoration	Develop the algorithms for Image filters	K6
Edge Detection	Construct the edge detection techniques for color and grayscale images	K6
Image Segmentation	Design the applications for image segmentation methods	K6
Image Compression	Develop the algorithms for Image compression	K6
Color Image Processing	Construct the methods for colour image processing operations	K6
Video enhancement	Design the applications for Video enhancement techniques	K6
Video Restoration	Develop the algorithms for Video filters	K6
Motion detection and Segmentation in videos	Construct the Motion detection techniques for Videos	K6

4. MAPPING SCHEME (CO, PO & PSO)

P21CS3P5	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PSO1	PSO2	PSO3	PSO4
CO1	H	H	H	H	H		M		H	H	H	H	
CO2	H	H	H	H	M		M		H	H	M	H	M
CO3	H	H	H	H	M		M		H	H	M	H	M
CO4	H	H	H	H	M		M		H	H	M	H	M
CO5	H	H	H	H	M		M		H	H	M	H	M
CO6	H	H	H	H	H	M	H	M	H	H	M	H	L

5. COURSE ASSESSMENT METHODS

DIRECT

1. Continuous Assessment Test: T1, T2 (Theory & Practical Components): Closed Book
2. Open Book Test.
3. Cooperative Learning Report, Assignment, Group Presentation, Group Discussion, project Report, Field Visit Report, Poster Presentation, Seminar, Quiz (written).
4. Pre-Semester & End Semester Theory Examination

INDIRECT

1. Course end survey (Feedback)

Name of the Course Coordinator: Dr. P. S. Eliahim Jeevaraj

MACHINE LEARNING LAB

SEMESTER: II
CREDITS : 3

Code: P21CS3P6
HOURS/WEEK :5

1. COURSE OUTCOMES

On successful completion of this course, students will be able to:

CO #	Course Outcomes	Level	Ex. No.
CO1	Apply the Pre-processing on the given dataset	K5	1
CO2	Develop and evaluate the regression model using Datasets	K6	2,3,4,
CO3	Develop and evaluate the Support Vector Machine using Datasets	K6	5
CO4	Develop and evaluate the Decision Tree Classifier and Regressor using Datasets	K6	6,7
CO5	Develop and evaluate the Clustering model using Datasets	K6	8, 9
CO6	Develop and evaluate the Neural Network and Perceptron model using Datasets	K6	10

2. SYLLABUS

- [Data preprocessing]: Create dataset in CSV file with 10 rows and 5 columns (name, age, height, weight, BP). Load the CSV file and do the following preprocessing to the data. Apply data range normalization, predict the missing value (if any) and select the useful features automatically. Finally print the preprocessed data.
- [Linear Regression]: Generate random numbers for input features and target. Divide the data into training and testing set. Print the training and testing data. Create a linear regression model for the dataset and run the model for 100 iterations. Print the evaluation results and learned model weights.
- [Multiple Linear Regression]: Load the boston house price prediction dataset. Divide the data into training and testing set. Print the training and testing data. Create a multiple linear regression model for the dataset and run the model for 500 iterations. Print the evaluation results and learned model weights.
- [Logistic Regression]: Load the Iris classification dataset. Divide the data into training and testing set. Print the training and testing data. Create a logistic regression model for the dataset and run the model for 500 iterations. Print the evaluation results and learned model weights.
- [Support Vector Machine]: Load the Digits classification dataset. Divide the data into training and testing set. Print the training and testing data. Create a Linear SVM model for the dataset and run the model for 500 iterations. Apply logistic regression to the dataset and print the results. Print the evaluation results and learned model weights. Compare the results with results of logistics regression in terms of speed and accuracy.
- [Decision Tree Classifier]: Load the Wine classification dataset. Divide the data into training and testing set. Print the training and testing data. Create a Decision Tree model for the dataset and run the model for 500 iterations. Print the evaluation results and learned model weights. Apply SVM to the dataset and print the results. Compare the results with results of Support Vector machines in terms of speed and accuracy.
- [Decision Tree Regressor]: Load the Diabetes prediction dataset. Divide the data into training and testing set. Print the training and testing data. Create a Decision Tree Regression model for the dataset and run the model for 500 iterations. Apply Linear Regression to the dataset and print the results. Print the evaluation results and learned model weights. Compare the results with results of Linear regression in terms of speed and accuracy.
- [K-Means Clustering]: Populate clustering dataset with 2 features using sklearn built-in functions. Divide the data into training and testing set. Print the training and testing data. Apply K-Means for the dataset and run the model for 500 iterations. Visualize the clustering results using 2D plots.
- [Spectral Clustering]: Populate clustering dataset with 2 features using sklearn built-in functions. Divide the data into training and testing set. Print the training and testing data. Apply spectral clustering for the dataset and run the model for 500 iterations. Visualize the clustering results using 2D plots. Visually compare the results with plots of the K-means in terms of speed and clustering validity.
- [Simple Neural Networks]: Load the Digits classification dataset. Divide the data into training and testing set. Print the training and testing data. Create a Neural Network with 2 layers and 100

nodes each for the dataset and run the model for 500 iterations. Apply logistic regression, SVM, decision tree to the dataset and print the results. Print the evaluation results and learned model weights. Compare the results in terms of speed and accuracy.

3. SPECIFIC LEARNING OUTCOMES

S.No.	Exercise List	Level
1	Apply the pre-processing on data set	K6
2	Develop a linear regression model for user-built data set	K6
3	Design a multiple regression model for Boston House Price Data set	K6
4	Develop a logistic regression model for Iris Dataset	K6
5	Design a Support Vector Machine for Digit Dataset	K6
6	Develop a Decision Tree Classifier for Wine Dataset	K6
7	Construct a Decision tree regressor for Diabetics Dataset	K6
8	Design a K-means clustering for dataset	K6
9	Develop a Spectral Clustering for given Dataset	K6
10	Design a Neural Network for Digit Dataset	K6

4. MAPPING SCHEME (CO, PO & PSO)

P21CS3P6	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PSO1	PSO2	PSO3	PSO4
CO1	H	M	H		H					H		M	
CO2	H	M	M	M	H	H							
CO3	H	H	H	H	M	L	M		H	M	H	L	
CO4	H	H	M	L	H	H		H	M	H	M	M	H
CO5	H	H		H	H		H	H	H	H	H	M	H
CO6	H		H	H	H		H	H	H	H	H	H	H

5. COURSE ASSESSMENT METHODS

DIRECT

1. Continuous Assessment Test: T1, T2 (Theory & Practical Components): Closed Book
2. Open Book Test.
3. Cooperative Learning Report, Assignment, Group Presentation, Group Discussion, project Report, Field Visit Report, Poster Presentation, Seminar, Quiz (written).
4. Pre-Semester & End Semester Theory Examination

INDIRECT

1. Course end survey (Feedback)

Name of the Course Coordinator: Dr. B. Satheesh Kumar

CORE X: NEURAL NETWORKS AND DEEP LEARNING

SEMESTER: IV
CREDITS: 4

CODE: P21CS410
HOURS/WEEK: 5

1. COURSE OUTCOMES

On successful completion of this course, students will be able to:

CO #	Course Outcomes	Level	Unit Covered
CO1	Develop a simple Neural Network for the given problem	K4	I
CO2	Perform training and evaluation using various metrics	K5	II
CO3	Design NEAT network	K4	III
CO4	Design and develop convolutional neural network	K5	IV
CO5	Develop the algorithms for pruning the CNN	K5	V
CO6	Design and develop recurrent neural network	K6	V

2. A. SYLLABUS

Unit I : Neural Network Part-I

Background: Typical NN, NN Structure, XOR example for NN input and output, Training, Car – Database example – Neurons and Layers – Types of Neurons – Activation Functions – Rectified Linear Units Activation Functions – Logic Programming with NN – Self Organizing Maps – Neighbourhood Functions – Hopfield Network – Training a Hopfield Network.

Unit II : Neural Network Part-II

Boltzmann Machine and Boltzmann Machine Probability – Boltzmann Machine for Traveling – Salesman Problem – Single Output Feed Forward NN – RBF Networks – Data Normalization – Training and Evaluation: Binary Classification, Sensitivity, Specificity, ROC curves – Multi-class Classification: Log loss, Multi class log loss, Means square error – Training with Simulated Annealing – Backpropagation Training: Gradients, Calculating output node delta and error functions.

Unit III : Neural Network Part-III

Backpropagation Training: Calculating interior node delta and derivatives of activation functions – Choices for Backpropagation: Batch, Online and Stochastic Gradient Descent Training – Resilient Propagation Training: Arguments and data structures – Resilient Propagation Training: Steps in the training process – Levenberg–Marquardt Training: Algorithm – Levenberg–Marquardt Training: Calculation of the Hessian – Levenberg–Marquardt Training: With multiple outputs and Overview of LMA process – NEAT Networks: Concepts – NEAT Networks: NEAT Mutation – NEAT Crossover.

Unit IV : Deep Learning Part-I

Deep Learning Features or Components – Deep Learning Tools – Comparison of NN and DBNN – Deep Belief Neural Networks architecture – Implementing DBNN with Restricted Boltzmann Machine – Training DBNN: Steps, Layer wise sampling, Computing positive and negative gradients, Updating weights and bias – Deep Belief Classification Example – Convolutional Neural Networks: LeNET-5, Convolutional layer – Convolutional Neural Networks: Max-Pool layer and Dense layer.

Unit V : Deep Learning Part-II

Pruning: connections, neurons, algorithm for pruning – Model Selection: Grid Search model selection – Model Selection: Random Search model selection, Other model selections – L1 Regularization – L2 Regularization – Dropout Regularization – Simple Recurrent Neural Networks: Context Neuron, Context Layer – Architecture of SRN: Elman SRN, Jordan SRN; – Training SRN: Backpropagation through time – Architecting Neural Network: Training parameters, Hyper Parameters.

B. TOPIC FOR SELF STUDY

- GoogleNet
- ResNet, Optimisers: Momentum Optimiser
- Momentum and Nesterov Accelerated Gradient (NAG) Optimiser
- Adagrad Optimiser

C. TEXT BOOK(S)

1. Jeff Heaton, “*Artificial Intelligence for Humans*”, Volume 3: Neural Networks and Deep Learning, Heaton Research Inc, 2015

D. REFERENCES BOOK(S)

1. Kevin Gurney, “*An Introduction to Neural Networks*”, UCL Press, 1997
2. Ian Goodfellow and Yoshua Bengio and Aaron Courville, Deep Learning, MIT Press, 2016, (<http://www.deeplearningbook.org>)

E. WEB LINKS

- <https://nptel.ac.in/courses/106/105/106105215/>)
- <https://www.coursera.org/learn/neural-networks-deep-learning>
- <https://www.deeplearning.ai/program/deep-learning-specialization/>

3. SPECIFIC LEARNING OUTCOMES

Unit	Topics	Unit Learning Outcomes	Level
I	Neural Network - I		
1.1	Typical NN, NN Structure, XOR example for NN input and output, Training, Car – Database example – Neurons and Layers	Defines the Typical Neural Networks with input and output examples along with Neurons and Layers	K3
1.2	Types of Neurons – Activation Functions – Rectified Linear Units Activation Functions – Logic Programming with NN	Explains the neuron types and the activation functions	K3
1.3	Self Organizing Maps – Neighbourhood Functions – Hopfield Network – Training a Hopfield Network.	Identifying the Neural maps and describes about the Hopfield Networks	K4
II	Neural Network - II		
2.1	Boltzmann Machine and Boltzmann Machine Probability – Boltzmann Machine for Traveling – Salesman Problem – Single Output Feed Forward NN – RBF Networks – Data Normalization	i) Detailed explanation and the Applications of th Boltzmann machine and the travelling sales man problem ii) Utilizing the Data Normalization and its various phases.	K3
2.2	Training and Evaluation: Binary Classification, Sensitivity, Specificity, ROC curves	Describes the evaluation and classification methods of NN	K3
2.3	Multi-class Classification: Log loss, Multi class log loss, Means square error – Training with Simulated Annealing	Dealing with the classification types especially log loss and the simulated annealing	K3
2.4	Backpropagation Training: Gradients ,Calculating output node delta and error functions.	Describes about the Back propagation method	K3
III	Neural Network - III		
3.1	Backpropagation Training: Calculating interior node delta and derivatives of activation functions	Explains the activation functions through the back propagation training.	K4
3.2	Choices for Backpropagation: Batch, Online and Stochastic Gradient Descent Training	Defines the various types of Back propagation techniques	K3
3.3	Resilient Propagation Training: Arguments and data structures, Resilient Propagation Training: Steps in the training process	Illustrating the various Resilient propagation training methods	K4
3.4	Marquardt Training: Algorithm, Levenberg, Marquardt Training: Calculation of the Hessian, Levenberg, Marquardt Training: With multiple outputs and Overview of LMA process	Detailed description of the Marquardt training and the algorithm with various examples & Applications.	K3

3.5	NEAT Networks: Concepts – NEAT Networks: NEAT Mutation – NEAT Crossover	Various perspective of the NEAT method in detail.	K3
IV Deep Learning -I			
4.1	Deep Learning Features or Components, Deep Learning Tools, Comparison of NN and DBNN, Deep Belief Neural Networks architecture, Implementing DBNN with Restricted Boltzmann Machine	Thorough look of Deep learning algorithm and the various levels, architecture of Deep Learning.	K4
4.2	Training DBNN: Steps, Layer wise sampling, Computing positive and negative gradients, Updating weights and bias, Deep Belief Classification Example	Next level of DBNN with classification examples	K3
4.3	Convolutional Neural Networks: LeNET-5, Convolutional layer, Convolutional Neural Networks: Max-Pool layer and Dense layer.	Describes the convolutional NN and its layer.	K4
V Deep Learning -II			
5.1	Pruning: Connections, neurons, algorithm for pruning, Model Selection: Grid Search model selection	Detailed explanation of pruning and its methodologies	K3
5.2	Model Selection: Random Search model selection, Other model selections – L1 Regularization – L2 Regularization – Dropout Regularization	Evaluating the different kinds of Model selection	K4
5.3	Simple Recurrent Neural Networks: Context Neuron, Context Layer – Architecture of SRN: Elman SRN, Jordon SRN;	Explains the Architecture of Simple Recurrent Neural Networks.	K4
5.4	Training SRN: Back propagation through time – Architecting Neural Network: Training parameters, Hyper Parameters.	Details of the Simple Recurrent Neural Network training.	K4

4. MAPPING SCHEME (CO, PO & PSO)

P21CS410	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PSO1	PSO2	PSO3	PSO4
CO1	H	H	H	H	H		M		H	H	H	H	
CO2	H	H	H	H	M		M		H	H	M	H	M
CO3	H	H	H	H	M		M		H	H	M	H	M
CO4	H	H	H	H	M		M		H	H	M	H	M
CO5	H	H	H	H	M		M		H	H	M	H	M
CO6	H	H	H	H	H	M	H	M	H	H	M	H	L

5. COURSE ASSESSMENT METHODS

DIRECT

1. Continuous Assessment Test: T1, T2 (Theory & Practical Components): Closed Book
2. Open Book Test.
3. Cooperative Learning Report, Assignment, Group Presentation, Group Discussion, project Report, Field Visit Report, Poster Presentation, Seminar, Quiz (written).
4. Pre-Semester & End Semester Theory Examination

INDIRECT

1. Course end survey (Feedback)

Name of the Course Coordinator: Dr. M. Newbegin

CORE XI: NETWORK SECURITY

SEMESTER: IV
CREDITS: 4

CODE: P21CS411
HOURS/WEEK: 5

1. COURSE OUTCOMES

On successful completion of this course, students will be able to:

CO #	Course Outcomes	Level	Unit Covered
CO1	Classify various of attacks	K2	I
CO2	Examine a variety of algorithm	K4	II
CO3	Apply the concepts of Encryption and decryption messages using block chippers and sign.	K3	III
CO4	Create digital signature using various algorithms.	K6	IV
CO5	Classify web security, intruders, viruses and firewalls	K4	V
CO6	Analyze Various Security Services	K5	V

2. A. SYLLABUS

Unit I : Symmetric Ciphers

Introduction: Overview of security concepts - Overview of Symmetric cryptography - General model for Symmetric encryption -Examine a variety of algorithm[Monoalphabetic, Polyalphabetic] - Look briefly a different approach known as Steganography - Principles of modern Symmetric Cipher(DES) - DES example and strength - Overview of general structure of AES, Transformation function - AES key expansion,AES Example, Implementation - Analyse the security of encryption Scheme(Dpuble DES, Triple DES) - Compare the content ECB,CBC, CFB, OFB and counter Model of operation.

Unit II : Asymmetric Ciphers and Public key Crpteograh, Data Integrity

Key concept to Prime numbers, Fermat's Theorem - Euclers Theorem, Testing of primality Chinis Remainder Theorem - Overview of the basic principle of public key Cryptography - RSA algorithm - Diffie Hell Mon key exchange(PKCS) - Eligamal cryptography System - Application of nCryptographic HASH function(Message authentication, Digital Signature) - Two (b simple Hash function Bit by Bit XOR, SHA Hash function based Cipher block Chaning) - SHA

Unit III :Message Authentication and Digital Signature

Introduction to the requirement of Message authentication - Digital signature - HMAC,MAC based of Block cipher DAA, CMAC - Authentication Encryption CCM and GCM - Key wrapping and unwrapping algorithm - Pseudo Random number generation using Hash functions - PRNG on MAC function, PRNG based on Hash and MAC - Digital Signature requirements, Elgamal DS - Schnorr Digital Signature NIST Signature(DSA approach) - Elliptic Curve DSA - RSA-PSS Digital Signature algorithm.

Unit IV : Key management and Distribution user authentication

Symmetric Key Distribution using Encryption – Symmetric Key distribution using asymmetric key encryption – Distribution of Public Keys – X.509 certificates – PKI-PKIX management function, Protocols – Remote user authentication principle, Remote user authentication using symmetric encryption – Kerberos- version 4 – Kerberos version 5 – Federated Identity Management - Remote user authentication using asymmetric encryption, Personal Identity Verification.

Unit V : Network and Internet Security

Network access model, Extensible authentication Protocol - IEEE802.1X port based network access control - Cloud computing - Cloud secutiry,Risks and counter measures - Cloud security as a service - Transport level security-web security, SSL - TLS, HTTP - SSH , wireless security Electronic mail security, Pretty good privacy - IP security overview, policy.

B. TOPICS FOR SELF STUDY

- Cryptoanalysis
- Memory Trade off Attack

- Differential Cryptoanalysis
- Linear Cryptoanalysis

C. TEXT BOOK(S)

1. *“Cryptography and network security”*: William Stallings, Pearson Edition 6th Edition,2016.

D. REFERENCE BOOK(S)

1. *“Cryptography-Made-Simple”*:Nigel.P. Smart.

E. WEB LINKS

- <https://alison.com/courses/network-and-security>
- <https://www.coursera.org/courses?query=network%20security>
- <https://www.edx.org/learn/network-security>

3. SPECIFIC LEARNING OUTCOMES:

Unit	Subtopics	Learning Outcomes	Level
I	Symmetric Ciphers		
1.1	Overview of security concepts	Understand the concepts of security	[K1]
1.2	Examine a variety of algorithm	Examine the variety of algorithms like Monoalphabetic, Polyalphabetic	[K4]
1.3	Analyze the security of encryption Scheme	Analyze the security of encryption scheme such as Dpublic DES, Triple DES	[K4]
II	Asymmetric Ciphers and Public key Cryptography, Data Integrity		
2.1	Key concept to Prime numbers	Recall key concepts of Prime Numbers and Define Various theormos	[K1]
2.2	Overview of the basic principle of public key Cryptography	Categorize the public key cryptography algorithms based on their basic principles	[K4]
2.3	Application of Cryptographic HASH function	Evaluate the applications of cryptographic hash functions	[K5]
III	Message Authentication and Digital Signature		
3.1	Introduction to the requirement of Message authentication	Explain the requirements of Message authentication	[K5]
3.2	Key wrapping and unwrapping algorithm	Classify Key wrapping and unwrapping algorithms	[K2]
3.3	Digital Signature requirements	Elaborate the concepts of Digital Signature Requirements	[K6]
IV	Key management and Distribution user authentication		

4.1	Symmetric Key Distribution	Explain Symmetric Key Distributions	[K5]
4.2	Distribution of Public Keys	Classify Distribution of public keys	[K3]
4.3	Remote User Authentication	Analyze Remote User Authentication using symmetric and asymmetric encryptions	[K4]
V	Network and Internet Security		
5.1	Network access model, Extensible authentication Protocol	Explain the Network access Model and Extensible authentication protocol	[K4]
5.2	Cloud computing	Evaluate Cloud security, Risks and counter measures	[K5]
5.3	Cloud security, Transport level security, web security, wireless security and IP security	Elaborate the concepts of Cloud security, Transport level security, web security, wireless security and IP security	[K6]

4. MAPPING SCHEME (CO, PO & PSO)

P21CS411	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PSO1	PSO2	PSO3	PSO4
CO1	H	H	H	H	H		M		H	H	H	H	
CO2	H	H	H	H	M		M		H	H	M	H	M
CO3	H	H	H	H	M		M		H	H	M	H	M
CO4	H	H	H	H	M		M		H	H	M	H	M
CO5	H	H	H	H	M		M		H	H	M	H	M
CO6	H	H	H	H	H	M	H	M	H	H	M	H	M

5. COURSE ASSESSMENT METHODS

DIRECT

1. Continuous Assessment Test: T1, T2 (Theory & Practical Components): Closed Book
2. Open Book Test.
3. Cooperative Learning Report, Assignment, Group Presentation, Group Discussion, project Report, Field Visit Report, Poster Presentation, Seminar, Quiz (written).
4. Pre-Semester & End Semester Theory Examination

INDIRECT

1. Course end survey (Feedback)

Name of the Course Coordinator: Dr. G. Sobers Smiles David

ELECTIVE 5a: INTERNET OF THINGS

SEMESTER: IV
CREDITS: 4

CODE: P21CS4:5
HOURS/WEEK: 5

1. COURSE OUTCOMES

On successful completion of this course, students will be able to:

CO #	Course Outcomes	Level	Unit
1	Evaluate the architecture of IoT and Business Models	K6	I
2	Design and Develop the tools for IoT	K4	II
3	Assess the security and privacy in IoT	K5	III
4	Design the IoT devices using hardware and Arduino	K6	IV
5	Build the wearable device for IoT computing	K6	IV
6	Analyse the application of IoT	K6	V

2. A. SYLLABUS

Unit I: Internet of Things & Architecture

Introduction - Definition of IoT – Evolution - Related Statistics- Industrial Forecasts. IoT Architecture And Business Models: Introduction - IoT Architecture - Business Models for IoT

Unit II: Communication And Development Primitives & Challenges in IoT

Communication And Development Primitives: Introduction - Communication Protocols, Standards and Policies - Design and Development Tools – Datasets. **Challenges:** Introduction- Challenges associated with IoT Architecture - Challenges associated with Entities - Technology-oriented Challenges - Challenges associated with Features - Challenges associated with Development Practices.

Unit III: Data Explosion, Security and Privacy in IoT

Data Explosion : Introduction - Big Data Definition - Underlying Factors for Data Explosion in IoT - Key Challenges associated with managing Data Explosion in IoT - Technological Support. **Security and Privacy :** Security Issues in the IoT - Security Mechanisms Overview - Key Agreement, Distribution, and Security Bootstrapping - Privacy Issues in the IoT

Unit IV: IoT in Practice

Hardware for the IoT: Classes of Constrained Devices - Hardware Platforms - TelosB - Zolertia Z1 - OpenMote - Arduino - Intel Galileo - Raspberry Pi. Software for the IoT: OpenWSN - TinyOS - FreeRTOS -TI-RTOS - RIOT - Contiki OS - Networking - Low-power Operation - Simulation - Programming Model – Features. Wearable Computing for the IoT: Shaping the Internet of Things in a Mobile-Centric World - Interaction Patterns with Smart Objects through Wearable Devices - Implementation in a Real-world IoT Testbed.

Unit V: Computing Paradigms & Security Issues in Connected & Autonomous Driving Vehicles

Introduction - Connected and Autonomous Driving Vehicle (CADV) – A Technical Background - Computing Paradigms: Artificial Intelligence (AI) - Vehicular Cloud Computing (VCC) or Edge Computing - End-to-End Networking. Security Issues and Existing Countermeasures - Challenges and Countermeasures associated with Computing Paradigms - Challenges and Countermeasures associated with Participating Entities.

B. TOPICS FOR SELF STUDY

- 5G Networks and IoT.
- IoT Security and Trust.
- IoT and Personal Data Protection.
- Artificial Intelligence and IoT.

C. TEXT BOOKS

1. Brij B. Gupta, Megha Quamara, “ Internet of Things Security Principles, Applications, Attacks, and Countermeasures” , 1st Edition, CRC Press, 2020. (Unit I,II, IIIA, V)

2. Simone Cirani, Gianluigi Ferrari, Marco Picone, Luca Veltri , “Internet of Things_ Architectures, Protocols and Standards”, Wiley, 2018. (Unit IIIB, Unit IV)

D. REFERENCES

1. Arshdeep Bahga, Vijay Madisetti, “Internet of Things – A hands-on approach”, Universities Press, 2012.
2. Honbo Zhou, “The Internet of Things in the Cloud: A Middleware Perspective”, CRC Press, 2012.
3. Dieter Uckelmann, Mark Harrison, Florian Michahelles, “Architecting the Internet of Things”, Springer, 2011.

E. Web Links

- <https://iot-analytics.com/iot-edge-computing-what-it-is-and-how-it-is-becoming-more-intelligent/>
- <https://www.coursera.org/learn/iot-wireless-cloud-computing>
- https://onlinecourses.nptel.ac.in/noc20_cs66/preview

3. SPECIFIC LEARNING OUTCOMES

Unit	Course Content	Learning Outcomes	Level
I	Internet of Things & Architecture		
1.1	Introduction - Definition of IoT – Evolution - Related Statistics- Industrial Forecasts.	Explain the evolution of IoT Explain the industrial forecasts using IoT	K5 K5
1.2	IoT Architecture And Business Models: Introduction - IoT Architecture	Explain the IoT Architecture	K5
1.3	Business Models for IoT	Evaluate the business models for IoT	K6
II	Communication And Development Primitives & Challenges in IoT		
2.1	Communication And Development Primitives: Introduction - Communication Protocols, Standards and Policies - Design and Development Tools – Datasets.	Explain the communication protocols List the development tools List the Datasets for IoT	K5 K3 K3
2.2	Challenges: Introduction- Challenges associated with IoT Architecture - Challenges associated with Entities	Assess the challenges for IoT Architecture	K6
2.3	Technology-oriented Challenges - Challenges associated with Features - Challenges associated with Development Practices.	List the technology oriented challenges	K3
III	Data Explosion, Security and Privacy in IoT		
3.1	Data Explosion : Introduction - Big Data Definition - Underlying Factors for Data Explosion in IoT - Key Challenges associated with managing Data Explosion in IoT - Technological Support.	Explain the Data Explosion List the challenges in Data Explosion in IoT	K5 K3
3.2	Security and Privacy : Security Issues in the IoT - Security Mechanisms Overview - Key Agreement, Distribution, and Security Bootstrapping	Assess security in IoT Design the security bootstrapping in IoT	K6 K6
3.3	Privacy Issues in the IoT	Evaluate the privacy issues in IoT	K6
IV	IoT in Practice		
4.1	Hardware for the IoT: Classes of Constrained Devices - Hardware Platforms - TelosB - Zolertia Z1 - OpenMote - Arduino - Intel Galileo - Raspberry Pi.	Design the IoT devices using hardware tools	K6

4.2	Software for the IoT: OpenWSN - TinyOS - FreeRTOS - TI-RTOS - RIOT - Contiki OS - Networking - Low-power Operation - Simulation - Programming Model – Features.	Develop the code for IoT devices using the software IDEs	K6
4.3	Wearable Computing for the IoT: Shaping the Internet of Things in a Mobile-Centric World - Interaction Patterns with Smart Objects through Wearable Devices - Implementation in a Real-world IoT Testbed.	Explain the wearable computing for IoT Assess the IoT data from wearable devices	K5 K6
V	Computing Paradigms & Security Issues in Connected & Autonomous Driving Vehicles		
5.1	Introduction - Connected and Autonomous Driving Vehicle (CADV) – A Technical Background	Design the CADV	K6
5.2	Computing Paradigms: Artificial Intelligence (AI) - Vehicular Cloud Computing (VCC) or Edge Computing - End-to-End Networking.	Explain the computing paradigms in CADV	K6
5.3	Security Issues and Existing Countermeasures - Challenges and Countermeasures associated with Computing Paradigms - Challenges and Countermeasures associated with Participating Entities.	Assess the security issues and challenges in CADV	K6

4. MAPPING SCHEME (CO, PO & PSO)

P21CS4:1	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PSO1	PSO2	PSO3	PSO4
CO1	H	H	H	H	H		M		H	H	H	H	
CO2	H	H	H	H	M		M		H	H	M	H	M
CO3	H	H	H	H	M		M		H	H	M	H	M
CO4	H	H	H	H	M		M		H	H	M	H	M
CO5	H	H	H	H	M		M		H	H	M	H	M
CO6	H	H	H	H	H	M	H	M	H	H	M	H	L

5. COURSE ASSESSMENT METHODS

DIRECT:

1. Continuous Assessment Test: T1, T2 (Theory & Practical Components): Closed Book
2. Assignment, Group Presentation, Poster Presentation, Seminar, Quiz (written).
3. Pre-Semester & End Semester Theory Examination

INDIRECT:

1. Course end survey (Feedback)

Name of the Course Coordinator: Dr. R. Jemimah Priyadharshini

PROJECT

SEMESTER: IV
CREDITS: 6

CODE: P21CS4PJ
HOURS/WEEK: 6

COURSE OUTCOMES

Upon completion of this course, students should be able to:

- Identify and define the problem statement
- Define and justify scope of the proposed problem
- Gather and analyze system requirements
- Propose an optimized solution among the existing solutions
- Practice software analysis and design techniques
- Develop a functional application based on the software design
- Apply coding, debugging and testing tools to enhance the quality of the software
- Construct new software system based on the theory and practice gained
- Prepare proper documentation of software project following the standard guidelines
- Develop technical reporting and oral presentation skills

Text Book(s):

1. Lynn E. Miner & Jeremy T. Miner, "*Proposal Planning and Writing*", Third Edition, Greenwood Publishing Group, 2003.

References Book(s):

1. William Navidi, "*Statistics for Engineers and Scientists*", 2nd Edition, McGraw-Hill, 2007.