ESTD : 1946

THE NATIONAL INSTITUTE OF ENGINEERING

MYSORE – 8

(Autonomous Institution under VTU)

B.E - CSE

Scheme of V - VI Semester

Department of Computer Science and Engineering

The National Institute of Engineering

Scheme of Teaching & Examination (2022 Scheme)

Department: Computer Science and Engineering (BE in CS&E)

B.E. 2023 Admitted Batch

V Semester

Sl.No	Type of Course	Course Code	Course Title	Teaching Department (TD)	Question Paper setting Board (PSB)	Teaching Hrs/Week				Examination				Credits
						L	T	P	S	Duration in Hours	CIE Marks	SEE Marks	Total Marks	
1	HSMS	BCS501	Software Engineering & Project Management	CS	CS	3	0	0		3	50	50	100	3
2	IPCC	BCS502	Computer Networks	CS	CS	3	0	2		3	50	50	100	4
3	PCC	BCS503	Automata Theory and Computation	CS	CS	3	2	0		3	50	50	100	4
4	PCCL	BCSL504	Full Stack Development Lab	CS	CS	0	0	2		2	50	50	100	1
5	PCC	BCS505	Full Stack Development	CS	CS	3	0	0		3	50	50	100	3
6	PEC	BCS516X	Professional Elective Course (Industry suggested course) - Group I	CS	CS	3	0	0		3	50	50	100	3
7	PROJ	BCS586	Minor Project	CS	CS	0	0	2		-	50	-	50	1
8	AEC	BRMCS557	Research Methodology and IPR	CS	CS	2	0	0		2	50	50	100	2
9	MC	BESK508	Environmental Studies	Civil Engg	Civil	1	0	0		-	50	-	50	1
10	MC	BNSK559	National Service Scheme (NSS)	NSS Coordinator		0	0	2		_	100	_	100	0
		BPEK559	Physical Education (PE) (Sports & Athletics)	PED										
		BYOK559	Yoga	Yoga Teacher										
		BMUK559	Music	Music Teacher										
Total											550	350	900	22

Professional Elective Course - Group I			
BCS516A	Computer Graphics and Visualization		
BCS516B	Artificial Intelligence		
BCS516C	Unix System Programming		

Course Code: BCS501 **Course: Software Engineering & Project Management**

Credits: 3 **L:T:P - 3:0:0**

SEE: 50% **CIE: 50%**

SEE Hours: 3 **Max. Marks: 100**

Prerequisites if any	**NIL**
Learning objectives	1. Learn fundamental software engineering processes and models. 2. Learn to apply analysis and modeling techniques for real-world software. 3. Learn to validate software using effective testing strategies 4. Learn to employ suitable software project estimation models.

On successful completion of the course, the student will be able to:

COs	**Course Outcomes**	**Bloom's level**
CO1	Explain the principles of software engineering, process models and agile practices for structured software development.	Understand
CO2	Apply suitable requirement analysis and modeling techniques to design software solutions for real-world applications.	Apply
CO3	Analyze software quality by employing appropriate testing strategies and project estimation methods for software development and people management.	Analyze
CO4	Formulate, present the role-play and outcomes as a team to simulate real-world project scenarios.	Evaluate

Mapping with POs and PSOs

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12		PSO1	PSO2
CO1	3	3	2	2	2	-	-	-	2	3	-	3		1	-
CO2	3	2	3	3	2	-	-	-	2	2	-	3		2	-
CO3	2	2	2	3	2	-	-	-	2	2	-	3		3	-
CO4	2	3	3	2	2	2	-	3	3	3	3	3		1	2

Mapping Strength: Strong-3 Medium-2 Low -1

Nos.	Module	No. of Lecture Hours	No. of Tutorial Hours	No. of Practical Hours
	Module -1			
1.1	Introduction: The Nature of Software - Software application domains	**2**	**-**	**-**
1.2	Software Process : Software Engineering A layered Technology Process Frame work Generic Process model	**3**	**-**	**-**
1.3	Process Models: Incremental Process Models Evolutionary Process models: Prototyping, Spiral model	**3**	**-**	**-**
	Module – 2			
2.1	Agile View of Process: Agility, Agile Process Agile Process Model: Extreme programming (XP) Scrum	**3**	**-**	**-**
2.2	Requirement Engineering: Introduction to requirement engineering Requirement Engineering Tasks Initiating Requirement Engineering Process Developing USE-CASE	**4**	**-**	**-**
	Module – 3			
3.1	Building The Analysis Model: Requirement Analysis , Data Modeling concept, Analysis Modeling Approach Scenario Based Modeling Flow Based Modeling Behavioral Modeling	**4**	**-**	**-**
3.2	Design Engineering: Design Process Design Quality Design Concepts	**4**	**-**	**-**
3.3	Creating an Architectural Design: Software Architecture Data Design Architectural Styles and Patterns	**3**	**-**	**-**
	Module - 4			
4.1	Testing Strategies: A Strategic Approach to Software Testing, Test Strategies for Conventional Software Unit Testing and Integration Testing Validation testing	**3**	**-**	**-**
4.2	Testing Tactics: Software Testing Fundamentals Black Box & White Box Testing Basis Path Testing ,Black Box Testing	**4**	**-**	**-**
	Module – 5			
5.1	Project Management: Project Management Spectrum People, Product Process, Project	**3**	**-**	**-**
5.2	Software Project Estimation: Decomposition Techniques Software Project Estimation: Decomposition Techniques Empirical Estimation Models	**3**	**-**	**-**
5.3	Report writing	**1**	**-**	
	Total No. of Lecture Hours	**40**		
	Total No. of Tutorial Hours		**0**	
	Total No. of Practical Hours			**0**

Text Book :

1. **Software Engineering: A Practitioners Approach – Roger S. Pressman, 7th Edition, McGraw-Hill 2010**

Reference Books:

1. **Software Engineering: Ian Somerville, 10th Edition, Pearson Education, 2016.**
2. **Software Engineering Theory and Practice: Shari Lawrence Pfleeger, Joanne M. Atlee, 3rd Edition, Pearson Education, 2006.**
3. **Software Engineering Principles and Practice: Waman S Jawadekar, Tata McGraw Hill, 2004**

Online Resources:

1. **https://www.digimat.in/nptel/courses/video/106101061/L01.html**
2. **https://www.digimat.in/nptel/courses/video/106105182/L01.html**
3. **https://www.coursera.org/learn/software-processes-and-agile-practices**

Course Code: BCS502 **Course Name: Computer Networks**

Credits:4 **L:T: P- 3:0:2**

SEE:50%Marks **CIE:50%Marks**

SEE Hours: 3 **Max.Marks:100**

Prerequisites if any	Basic concepts of communication, Digital electronics and computers.
Learning objectives	1.To Understand the fundamentals of computer networks by studying networking models and the services provided at the network layer. 2. To analyze the mechanisms of IPv4 and IPv6 addressing and apply subnetting concepts for efficient network design 3. To analyze the services and operations of protocols in the Network, Transport, and Application layers. 4. Implement networking concepts and protocol functionalities in simulated environments using Cisco Packet Tracer.

Course Outcomes:

On the successful completion of the course, the student will be able to

COs	**Course Outcomes**	**Bloom's level**
CO1	Explain the fundamental principles of Computer Networks, Networking models and the operations of core protocols.	Understand
CO2	Apply IPv4/IPv6 addressing, forwarding methods, routing protocols to enable efficient data transmission across networks.	Apply
CO3	Apply transport layer protocols and mechanisms to ensure reliable and optimal communication.	Apply
CO4	Analyze application-layer protocols and socket programming concepts for standard client-server communication.	Analyze
CO5	Implement networking concepts and protocol functionalities in simulated environments using Cisco Packet Tracer.	Evaluate

Mapping of COs with Pos and PSOs:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	1	2	2					2	1	3	2	2
CO2	3	3	3	2	2					2	1	3	3	3
CO3	3	3	2	2	2					2	1	3	3	3
CO4	3	3	2	2	2					2	1	3	3	3
CO5	3	3	3	2	3	1	1	1	2	2	1	3	3	3

Mapping Strength: Strong–3 Medium – 2 Low–1

Course Structure

Sl. No.	Modules	No. of Lecture Hours	No. of Tutorial Hours	No. of Practical Hours
	Module– 1:Introduction to Computer Networks and Network Layer			
1.1	Networks	1	-	-
1.2	Network Types	1		
1.3	**Networks Models:** TCP/IP Protocol Suite, The OSI Model.	1		
1.4	**Network Layer:** Network layer services: Packetizing, Routing and forwarding, other services	1	-	-
1.5	**Packet switching:** Datagram approach, Virtual-Circuit approach	1	-	-
1.6	Network layer performance	1	-	-
	Module– 2: Network Layer-Part I			
2.1	IPv4 addresses : Address space, Class full addressing	1	-	-
2.2	Class less addressing	2	-	-
2.3	DHCP, NAT	1	-	-
2.4	Forwarding of IP Packets: Forwarding Based on Destination Address	1		
2.5	**Network Layer Protocols:** Internet Protocol,(IP): IPV4 Datagram format	1	-	-
2.6	Fragmentation Options,	1	-	-
2.7	Security of IPV4 datagram, ICMP Checksum	1	-	-
2.8	**Next Generation IP:** IPV6Addressing	1	-	-
2.9	The IPV6 Protocol	1	-	-
	Module– 3:Network Layer-Part II			
3.1	**Unicast Routing:** Routing Algorithms: Distance Vector Routing,	1	-	-
3.2	Link state Routing	1		
3.3	Unicast Routing Protocols: Routing Information Protocol(RIP)	1	-	-
3.4	OSPF	1	-	-
3.5	Border Gateway Protocol(BGP):operation of External BGP(e BGP), Operation of Internal BGP (e BGP)	2	-	-
3.6	**Multicast Routing:** Introduction, Unicasting, Multicasting, broadcasting	1	-	-
3.7	MOSPF	1	-	-
	Module– 4: Transport Layer			
4.1	**Transport Layer:** Transport Layer Services: process-to-process communication, ICANN Ranges, Encapsulation and Decapsulation, Multiplexing and Demultiplexing, Flow control and Error Control	1	-	-
4.2	Connectionless and connection-oriented service, Transport Layer protocols: Introduction, services, port numbers	1	-	-
4.3	User Datagram Protocol: User Datagram and UDP services	1	-	-
4.4	Transmission Control Protocol: TCP services, TCP features, Segment			
4.5	TCP connection, Windows in TCP, Flow control(in brief)			
4.6	Error control :checksum, acknowledgement, generating acknowledgement, Retransmission- Retransmission after RTO, Retransmission after 3 duplicates, Out of order segments			
4.7	TCP congestion control: congestion window, congestion detection, Congestion policies, fast recovery	1	-	-

	Module –5:Application Layer			
5.1	**Application Layer:** Application-Layer Paradigms, Application Programming Interface (Socket and socket address)	2	-	-
5.2	**Standard Client Server Protocols:** World Wide Web	1	-	-
5.3	HTTP	1		
5.4	FTP	1	-	-
5.5	Electronic Mail	1		
5.6	TELNET	1		
5.7	Secure Shell(SSH)-components, Domain Name System (DNS)	1	-	-
	Total No. of Hours	**40Hours**		

	List of Experiments			
1	Study of networking devices like Hub, Repeater ,Bridge, Switch, Router, Gateway, Access point, MODEM and NIC			1
2	Write a program to implement the following Error Detection Techniques a. Cyclic Redundancy Check(CRC) b. Check sum			1
3	a) Introduction to Cisco packet tracer. b) Create basic topologies and assign IP address, subnet mask, gateway IP address and test connectivity using PING command.			1
4	Perform: a) An Initial Switch Configuration. (Host name, Console password, vty password, Privileged EXEC mode password, Privileged EXEC mode secret, IP address on VLAN1 interface, Default gateway) b) An Initial Router Configuration. (Configure the router host name, configure the passwords, configure the banner messages, verify the router configuration)			1
5	Configure and implement DHCP service in a Local Area Network. also Configure Cisco Router as a DHCP sever.			1
6	a. Implement the Static Routing and Default Routing. b. Configure WEP on a Wireless Router.			1
7	Configure Dynamic Routing using RIP.			1
8	c. Using TCP/IP sockets, write a client–server program to make the client send the file name and to make the server send back the contents of the requested file if present.			1
9	Configure and implement DNS service.			1
10	Defining and using Access Control Lists.			1
	Total No. of Lecture Hours	**40**	-	-
		Total No. of Tutorial Hours	**00**	**-**
			Total No. Of Practical Hours	**10**

Textbook:

1. **BehrouzForouzan," Data Communications and Networking",TataMcGraw-Hill,5th Edition,**

   2013.

Module1:

Chapter1:1.2,1.3 Chapter2:2.2,2.3

Chapter18:18.2,18.3,18.4(18.4.1,18.4.2,18.4.3,18.4.4,18.4.5)

Module2:

Chapter 18: 18.5(18.5.1**)** Chapter 19: 19.1.1,19.1.2,19.1.3,19.1.4,19.2.3

Chapter 22: 22.1,22.2

Module 3:

Chapter20:20.2(20.2.1,20.2.2),20.3(20.3.2,20.3.3,20.3.4)
Chapter21:21.1,21.3.2

Module 4:

Chapter23:23.1.1,

Chapter24:24.1(24.1.1,24.1.2),24.2(24.2.1,24.2.2),24.3(24.3.1,24.3.2, 24.3.3,24.3.4,24.3.6,24.3.7,24.3.8,24.3.9)

Module 5:

Chapter25:25.1.2,25.2.1

Chapter26:26.1(26.1.1,26.1.2),26.2,26.3(26.3.1),26.4,26.5(26.5.1),26.6.

Reference Book:

1. LarryPetersonandBruceSDavis"ComputerNetworks:ASystemApproach5th Edition,Elsevier-2014
2. ComputerNetworks,AndrewS.Tanenbaum,PearsonEducation,4thEdition, 2002.
3. DataandComputerCommunication,WilliamStallings,PearsonEducation,8th Edition,2007

OnlineResources:

https://archive.nptel.ac.in/courses/106/105/10610518

Code: BCS503	**Course: Automata Theory and Computation**
Credits: 4	**L:T:P - 3:2:0**
SEE: 50 Marks	**CIE: 50 Marks**
SEE Hours: 3	**Max. Marks:100**

Prerequisites if any	Any Programming language, Discrete Mathematical structures
Learning objectives	1. To give an overview of the theoretical foundations of computer science from the perspective of formal languages and illustrate finite state machines to solve problems in computing 2. To familiarize Regular grammars, context frees grammar and also to explain the hierarchy of problems arising in the computer sciences.

Course Outcomes:

On the successful completion of the course, the student will be able to

COs	**Course Outcomes**	**Bloom's level**
CO1	Explain the basic concepts of formal languages of finite automata techniques	Understand
CO2	Apply Finite Automata techniques to recognize the given Formal Languages	Apply
CO3	Analyze computational problems to determine the appropriate automata for their recognition	Analyze
CO4	Evaluate the suitability of Finite Automata Models for solving computational problems	Evaluate

Mapping with POs and PSOs:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12		PSO1	PSO2
CO1	3	1	2	3	-	-	-	-	2	2	-	2		1	1
CO2	3	2	2	2	3	-	-	-	3	3	-	2		3	3
CO3	3	3	3	3	-	-	-	-	3	3	-	2		3	3
CO4	3	3	2	3	-	-	-	-	3	3	-	2		3	3

Mapping Strength: Strong– 3 Medium – 2 Low – 1

Course Structure

Sl.No	Modules	No. of Lecture Hours	No. of Tutorial Hours	No. of Practical Hours
Module – 1: Introduction to Automata Theory				
1.1	Central Concepts of Automata Theory	1		
1.2	Finite Automata - Deterministic Finite Automata	2	1	
1.3	Non Deterministic Finite Automata, NFA to DFA	2		
1.4	Finite Automata with Epsilon Transitions	1		
1.5	Elimination of Epsilon Transitions	2	1	
Module – 2: Regular Expressions and Languages				
2.1	Regular Expressions	2		
2.2	Finite Automata and Regular Expressions – DFA to Regular Expression – Kleen’s closure	2	1	
2.3	Finite Automata to Regular Expression – State elimination	2		
2.4	Regular Expression to Finite automata	1	1	
2.5	Applications of Regular Expressions	1		
Module – 3: Properties of Regular Languages, Context Free Grammars				
3.1	Proving Languages not to be regular, Closure Properties of regular languages	2		
3.2	Equivalence and Minimization of Automata	2	1	
3.3	Context Free Grammars	2	1	
3.4	Leftmost, rightmost Derivations, Parse Trees, Ambiguity in Grammars	2		
Module – 4: Pushdown Automata				
4.1	Pushdown Automata, Languages of PDA	2	1	
4.2	Equivalence of PDA and CFG – Grammars to PDA	2		
4.3	Elimination of Epsilon, Unit Productions, Useless productions	2	1	
4.4	Chomsky Normal Form	2	1	
Module – 5: Turing Machines				
5.1	Introduction to Turing Machines	2		
5.2	Notation, Instantaneous description and transitions of Turning Machine	2	1	
5.3	Post’s Correspondence Problem	2		
5.4	Introduction to Class P and NP problems	2		
Total No. of Lecture Hours		**40**	**-**	**-**
Total No. of Tutorial Hours			**10**	**-**
Total No. of Practical Hours				**00**

Textbook:

1. John E Hopcroft, Rajeev Motwani, Jeffrey D Ullman, Introduction to Automata Theory, Languages, and Computation, 3rd Edition, Pearson Education

Reference Book:

1. Sipser, Michael. Introduction to the Theory of Computation. 3rd ed. Cengage Learning, 2012. ISBN: 9781133187790.
2. Peter Linz, An Introduction to Formal Languages and Automata, 6/e

Code: BCSL504 **Course: Full stack development lab**

Credits: 1 **L:T:P - 0:0:2**

SEE: 50 **CIE: 50**

Practical Hours:2 hrs/week **Max.Marks: 100**

Course Outcomes:

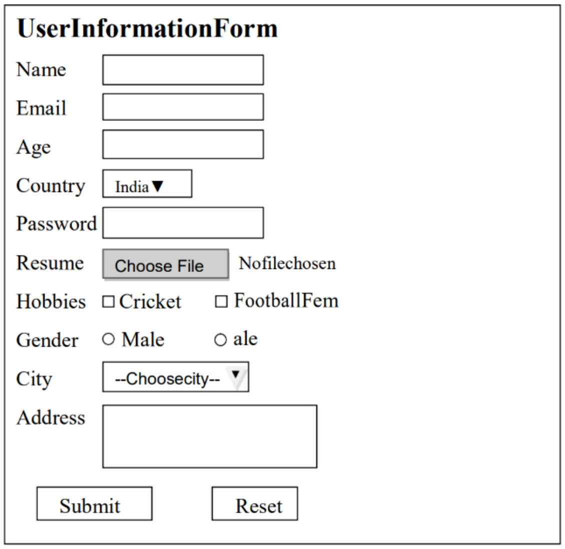
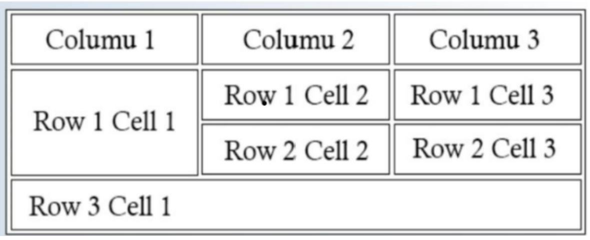
On the successful completion of the course, the student will be able to

COs	**Course Outcomes**	**Bloom's level**
CO1	Explain the roles of HTML/CSS, JavaScript/React, Node/Express, and databases in full-stack development.	Understand
CO2	Develop full-stack applications with responsive UI, interactive components, REST APIs, and database CRUD operations.	Apply
CO3	Analyze and debug data flow across client, server, and database to identify and resolve issues.	Analyze
CO4	Evaluate application design for usability, performance and security and recommend improvements.	Evaluate

Mapping with POs and PSOs

COs	P O1	PO 2	PO 3	PO 4	P O5	PO 6	PO 7	P O8	P O 9	PO 10	PO 11	PO 12		PS O1	PS O2
CO1	3	2	-	-	2	-	-	-	-	2	-	1		3	2
CO2	3	3	3	-	3	-	-	-	2	2	-	2		3	3
CO3	3	3	3	2	2	-	-	-	2	3	-	3		3	3
CO4	2	3	3	3	2	2	-	-	2	3	-	3		3	3

Mapping Strength: Strong– 3 Medium – 2 Low – 1

<table>
<tr><th>Sl.No</th><th>Lab Experiments</th><th>Course outcome</th></tr>
<tr><td>1</td><td>Create the below registration form using forms in HTML.

UserInformationForm
Name
Email
Age
Country India▼
Password
Resume Choose File Nofilechosen
Hobbies □Cricket □ FootballFem
Gender ○ Male ○ ale
City --Choosecity-- ▼
Address
Submit Reset</td><td>CO1</td></tr>
<tr><td>2</td><td>Design a webpage using HTML that meets the following requirements:
1. Create a table as shown below.
2. The first row of the table should include the headers: <b>"Column 1"</b>, <b>"Column 2"</b>and <b>"Column 3"</b>.
3. Add two additional rows with sample data under these headers.
4. Use <b>inline CSS</b> to:
a. Add a <b>border</b> to the table and its cells.
b. <b>Center-align</b> all the text within the table.

<table><tr><td>Columu 1</td><td>Columu 2</td><td>Columu 3</td></tr><tr><td rowspan="2">Row 1 Cell 1</td><td>Row 1 Cell 2</td><td>Row 1 Cell 3</td></tr><tr><td>Row 2 Cell 2</td><td>Row 2 Cell 3</td></tr><tr><td colspan="3">Row 3 Cell 1</td></tr></table></td><td>CO1</td></tr>
<tr><td>3</td><td>Create a visually appealing webpage that showcases the CSS box model by applying styling to a<div> element, incorporating padding, margins and borders. Additionally, include an image and a hyperlink within the page to enhance the layout.</td><td>CO1</td></tr>
</table>

4	Write a JavaScript program that demonstrates how to handle events from different HTML elements. The program should: 1. Trigger an alert when the user clicks anywhere on the body of the page. 2. Change the text of a button when it is clicked and display a success message below the button. 3. Dynamically update a paragraph with the text entered in a text box as the user types. 4. Display a password strength message (weak, medium, or strong) based on the length of the entered password, updating the message as the user types. 5. Ensure that the program uses appropriate event handlers for each element and explains the purpose of each event handler in the program.	CO2
5	Create a JavaScript program where a random number between 1 and 100 is generated and the user is prompted to guess the number. Provide feedback on whether the guess is too high, too low, or correct, and allow the user to keep guessing until they get it right. After a correct guess, ask if they want to play again using confirm(); if yes, restart the game, otherwise, display "Thanks for playing!" and end the program.	CO2
6	Write a JavaScript program to create an array of student objects, convert their marks to CGPA using the map method and display a list of students with a CGPA of 9 or higher. Also, show the total count of students with CGPA 9 and above using the filter and reduce methods.	CO2
7	1. Create an HTML page with a list of items, a few buttons and some paragraphs. 2. Write JavaScript to do the following: a) Change the background color of an element by its id. b) Change the text color of all list items (<li>) to blue. c) Change the text of all buttons to "Clicked!" when clicked. d) Increase the font size of all paragraphs. e) Select the first element with a specific class and change its text.	CO2
8	Demonstrate all the methods of mounting and updating phase of react class Component by changing the city name from “Mysore” to “Bangalore” after 2 seconds and display the same.	CO3
9	Illustrate react program to implement error boundaries to find 1. Division by zero error. 2. Array index of out of bound.	CO3
10	Implement a React functional component that: 1. Accepts an age prop and displays it inside a heading (<h1>). 2. Includes an input field and a button that allows the user to update the age in the heading using a state variable (utilizing useState&useEffect).	CO3
11	Write a react program to create a list of departments and iterate through all the elements of the list and return an unordered list with each department as a list item using a function component.	CO3

12	Write a react function component to create a counter that is incremented by one on button click and display a message 'End of the count ' when count reaches 10.	CO3
13	Write a Node.js program to read input from the console and store it in a file using file server module.	CO4
14	Apply the concept of custom modules in Node.js. 1. Create a custom module fileOperations.js with functions to read a file, write to a file and append data to a file. 2. Use this module in another file app.js to demonstrate its functionality by performing file operations.	CO4
15	Illustrate how to add an event listener, emit an event to trigger the listener and then remove the event listener in node.js.	CO4
16	Explain various debugging and logging methods in Node.js with example to demonstrates the following: 1. Measuring execution time of code blocks. 2. Grouping related log messages and displaying them hierarchically. 3. Using assertions to conditionally display error messages. 4. Displaying data in a tabular format. 5. Logging debugging details for analysis.	CO4
17	Write a Node.js program to: 1. Create an HTTP server that listens on port 8000. 2. Respond with “Welcome to Nodejs” when the root URL (/) is accessed. 3. Return a 404 status with a custom message for all other routes.	CO4
18	Demonstrate the following MongoDB operations 1. Write a script to create a new database named CollegeDB and a collection named students. 2. Insert three documents into the students collection with fields name, age, sem, usn and grade. 3. Insert multiple documents into a collection named courses with fields courseName, courseCode, and credits. 4. Write queries to retrieve all documents from the students collection. 5. Retrieve all documents where the age is greater than 18. 6. Write a query to find all students with a grade of "A". 7. Write a query to find all courses with credits greater than or equal to 3.	CO5
19	Demonstrate the following operations using Mongo shell script 1. Switch to the WorkDB database. 2. Create a teachers collection. 3. Insert documents with fields name, subject and experience. 4. Use the Mongo shell to export the students collection to a JSON file. 5. Import the JSON file back into a new collection named studentsBackup.	CO5
20	Write a MongoDB aggregation query for the following to analyze an employees collection with an example for each:Group the employees by their department and calculate the total number of employees in each department. The average salary of employees in each department. 1. Sort the aggregated results by the total number of employees in descending order. 2. Example Data in the employees Collection.	CO5

Code: BCS505 **Course: Full Stack Development**

Credits: 3 **L:T:P - 3:0:0**

SEE: 50 **CIE: 50**

SEE Hours: 3 **Max. Marks: 100**

Prerequisites if any	Computer networks
Learning objectives	1. Create web page using HTML & CSS 2. Develop familiarity with the JavaScript language, realize concepts commonly used in dynamic language programming, such as introspection, higher-order functions, closures, familiar with common libraries and tools that are used in web application development. 3. To Create React Components, lifecycle of components, rendering list and portal and perform some simple tests, and error handling. 4. To create Node.js modules, events and database access and interact with databases using MongoDB .

Course Outcomes: *On the successful completion of the course, the student will be able to*

COs	**Course Outcomes**	**Bloom's level**
CO1	Explain the principles of client-side, server-side, and database technologies in full-stack development	Understand
CO2	Apply HTML, CSS, JavaScript/React, Node.js/Express, and database concepts to design interactive web applications	Apply
CO3	Analyze application architecture, data flow, and integration across front-end, back-end, and database layers.	Analyze
CO4	Evaluate web applications for performance, scalability, security, and usability, and recommend improvements.	Evaluate

Mapping with POs and PSOs:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12		PSO1	PSO2
CO1	3	2	-	-	2	-	-	-	-	2	-	1		3	2
CO2	3	3	3	-	3	-	-	-	2	2	-	2		3	3
CO3	3	3	3	2	2	-	-	-	2	3	-	3		3	3
CO4	2	3	3	3	2	2	-	-	2	3	-	3		3	3

Mapping Strength: Strong– 3 Medium – 2 Low – 1

Course Structures

	Modules	No. of Lecture Hours	No. of Tutorial Hours	No. of Practical Hours
	Modules -1:HTML &CSS			
1.1	Introduction to HTML, Basic syntax, Standard HTML Document Structure, Images,	1	-	-
1.2	Paragraphs, Line Breaks, Heading Tags, Font Styles, Sizes, Color, Character Entity	1	-	-
1.3	links, Lists, tables, form tags, Font properties, text decoration, List property,	2	-	-
1.4	CSS Selectors, CSS Colors and background, CSS Box Model, CSS Margins, Padding, Borders. Background images, span and div tags, Grid and flex.	2	-	-
	Module 2: JavaScript			
2.1	Introduction to JavaScript, General syntactic characteristics	1	-	-
2.2	Primitives, operations, expressions	1	-	-
2.3	Screen output and keyboard input, Control statements,	1	-	-
2.4	Object creation &modification, arrays, array methods, Functions,	2	-	-
2.5	JavaScript and XHTML Documents: The JavaScript execution environment, TheDocument Object Model, Element access in JavaScript, Events and event handling,	2	-	-
2.6	Handling events from the Body elements, Button elements, Text box and Password elements.	2	-	-
	Module – 3: ReactJS			
3.1	Introduction, Templating using JSX	1	-	-
3.2	Components, State, Props,Hooks	2	-	-
3.3	Lifecycle of Components	2	-	-
3.4	Rendering List and Portals	2	-	-
3.5	Error Handling, Routers	2	-	-
	Module4: NodeJS			
4.1	Node.js overview, Node.js – basics and setup.	1	-	-
4.2	Node.js console, Node.js command utilities	1	-	-
4.3	Node.js modules, Node.js concepts	2	-	-

4.4	Node.js events, Node.js database access.	2		-
4.5	Introduction to Express.js • Creating routes (GET, POST, PUT, DELETE) • Handling request and response objects • Middleware in Express (built-in, custom, third-party)	2		
Module 5: MongoDB				
5.1	MongoDB basics: Documents, collections, database query language, mongo shell.	2	-	-
5.2	MongoDB CRUD operations (create, read, projection, update, delete aggregate)	2	-	-
5.3	reading from MongoDB, writing from MongoDB	2	-	-
5.4	MongoDB with PHP, MongoDB with NodeJS	2	-	-
	Total No. of Lecture Hours	**40**		-
	Total No. of Tutorial Hour		**00**	-
	Total No. of Practical Hours			**00**

Textbook:

1. Programming the World Wide Web, Robert W. Sebesta, 4th Edition, Pearson Education, 2008.
2. Full-Stack React Projects: Learn MERN Stack Development by Building Modern Web Apps Using MongoDB, Express, React, and Node.js, 2nd Edition.

Reference Book:

1. Open-Source Web Development with Lamp, James Lee and Brent Ware, Pearson Education, 2009.
2. Internet and World Wide Web: How to Program -Harvey M. Deitel, Paul J. Deitel, 4thedition, Pearson, 2009.
3. The Web Programming Building Internet Applications- Chris Bates, 3rd Edition, Wiley India, 2006

Online Resources:

1. MongoDB Notes for Professionals book Tutorial for Beginners in PDF by GoalKicker.com
2. https://docs.google.com/viewer?url=https://www.computer-pdf.com/pdf/0840-mongodb-notes-for-professionals-book.pdf
3. https://www.javatpoint.com/reactjs-tutorial

Code: BCS516A **Course: Computer Graphics and Visualization**

Credits: 3 **L:T:P - 3:0:0**

SEE: 50 Marks **CIE: 50 Marks**

SEE Hours: 3 **Max. Marks: 100**

Prerequisites if any	Computer concepts and programming
Learning objectives	● To understand Graphic system and use open GL APIs ● To perform geometric transformations and understand different viewing projections

Course Outcomes:

On the successful completion of the course, the student will be able to

COs	Course Outcomes	Bloom's level
CO1	Explain the basic components in computer graphic system	Understand
CO2	Illustrate the OpenGL APIs and components of interactive programming	Apply
CO3	Explain and Illustrate basics of geometric objects and transformations	Understand
CO4	Explain and demonstrate geometric transformations in homogeneous coordinates using OpenGL	Analyze
CO5	Analyze and compare parallel and perspective projections with relevant examples	Analyze

Mapping with Pos and PSOs:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12		PSO1	PSO2
CO1	1	2	-	-	-	-	-	-	-	-	-	-		-	-
CO2	1	2	-	-	3	-	-	-	-	-	-	-		-	-
CO3	2	2	-	-	2	-	-	-	-	-	-	-		-	-
CO4	3	2	-	-	3	-	-	-	-	-	-	-		2	2
CO5	2	2	2	-	3	-	-	-	-	-	-	-		2	2

Mapping Strength: Strong– 3 Medium – 2 Low – 1

Course Structure

Sl.	Module Name	No. of Lecture Hours	No. of Tutorial Hours	No. of Practical Hours
Module – 1: Introduction				
1.1	Applications of computer graphics	1	-	-
1.2	A graphics system	1	-	-
1.3	Images: Physical and synthetic	1	-	-
1.4	Imaging Systems	1	-	-
1.5	The synthetic camera model	1	-	-
1.6	The programmer’s interface, Graphics architectures	1	-	-
1.7	Programmable Pipelines, Performance characteristics	1	-	-
Module – 2: Graphics Programming				
2.1	The Sierpinski Gasket	1	-	-
2.2	Programming Two Dimensional Applications	1	-	-
2.3	The OpenGL API, Primitives and attributes	1	-	-
2.4	Color; Viewing	1	-	-
2.5	Control functions	1	-	-
2.6	The Gasket program	1	-	-
2.7	Polygons and recursion	1	-	-
2.8	Adding interactions, Menus	1	-	-
Module – 3: Geometric Objects and Transformations-I				
3.1	Scalars, Points, and Vectors	2	-	-
3.2	Three-dimensional Primitives	1	-	-
3.3	Coordinate Systems and Frames	1	-	-
3.4	Frames in OpenGL	1	-	-
3.5	Matrix and vector classes	1	-	-
3.6	Affine Transformations	1	-	-
3.7	Rotation	1	-	-

Module – 4:Geometric Objects and Transformations-II				
4.1	Transformation in Homogeneous Coordinates	2	-	-
4.2	Concatenation of Transformations	3	-	-
4.3	Transformation Matrices in OpenGL	3	-	-
Module – 5: Viewing				
5.1	Classical and computer viewing;;;; ;	1	-	-
5.2	Viewing with a Computer; Parallel Projections	1	-	-
5.3	Perspective Projections; Perspective Projections with OpenGL	1	-	-
5.4	Perspective Projection Matrices	1	-	-
5.5	Hidden-surface removal	2	-	-
5.6	Displaying Meshes	3	-	-
Total No. of Lecture Hours		**40**	**-**	**-**
Total No. of Tutorial Hours			**00**	**-**
Total No. of Practical Hours				**00**

Textbook:

1. Edward Angel: **Interactive Computer Graphics A Top-Down Approach with OpenGL**, 6th Edition, Pearson Education, 2012

Reference Book:

1. Donald Hearn and Pauline Baker: Computer Graphics- OpenGL Version, 3rd Edition, Pearson Education, 2004.
2. F.S. Hill Jr.: Computer Graphics Using OpenGL, 2nd Edition, Pearson education, 2001.
3. James D Foley, Andries Van Dam, Steven K Feiner, John F Hughes, Computer Graphics, Addison-Wesley 1997

Course Code: BCS516B **Course: Artificial Intelligence**

Credits:3 **L:T:P:**3:0:0

CIE: 50 Marks **SEE:**50 Marks

SEEHours:3 **Max.Marks:**100

Pre requisites if any	None
Learning objectives	1. To understand historical context of AI and its fundamental principles. 2. To acquire proficiency in applying basic AI principles to solve problems effectively. 3. To familiarize oneself with the methodologies of inference, perception, knowledge representation, and learning in AI.

Course Outcomes:
On the successful completion of the course, the student will be able to

COs	**Course Outcomes**	**Bloom's level**
CO1	Describe fundamental AI principles to address problems, knowledge representation and learning, intelligent agents and their interaction with the AI environment.	Understand
CO2	Apply problem-solving agents, search strategies and various reasoning methods to solve a given problem.	Apply
CO3	Analyze different AI approaches to distinguish their effectiveness in solving knowledge-based and uncertain problem scenarios.	Analyze
CO4	Evaluate search and inference algorithms within the context of problem-solving	Evaluate

Mapping with Pos and PSOs:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12		PSO1	PSO2
CO1	3	2	1	-	1	-	-	-	-	-	-	3		1	1
CO2	3	3	2	2	2	1	1	2	2	2	2	3		2	2
CO3	3	3	2	2	2	2	1	2	2	2	2	3		2	2
CO4	3	3	2	2	2	2	1	2	2	2	2	3		2	2

Mapping Strength: Strong–3 Medium–2 Low–1

Course Structure

Sl. no	Modules	No. of Lecture Hours	No. of Tutorial Hours	No. of Practical Hours
Module–1: Introduction to AI				
1.1	**Introduction:** What is AI? Foundations of AI.	2	-	-
1.2	History of Artificial Intelligence	1	-	-
1.3	The State of the Art	1	-	-
1.4	**Intelligent Agents:** Agents and environment	1	-	-
1.5	Concept of Rationality	1	-	-
1.6	The nature of environments	1	-	-
1.7	The structure of agents	1	-	-
Module–2: Problem solving based on searching				
2.1	Problems solving Agents, Example problems	2	-	-
2.2	Searching for solutions	1	-	-
2.3	Uniformed Search strategies–Uniform cost search	1	-	-
2.4	Breadth First Search, Depth First Search	1	-	-
2.5	Depth Limited Search	1	-	-
2.6	Iterative Deepening Depth First	2	-	-
Module–3: Heuristic Search Strategies				
3.1	Heuristic functions	1	-	-
3.2	Greedy best first search	1	-	-
3.3	A*algorithm	2	-	-
3.4	Local Search & Optimization: Hill Climbing	1	-	-
3.5	Genetic Algorithms	2	-	-
Module–4: Logical Agents, First Order Logic, Inference in First Order Logic				
4.1	**Logical Agents:** Knowledge–based agents	1	-	-
4.2	The Wumpus world, Logic	1	-	-
4.3	Proposition allogic	1	-	-
4.4	Reasoning patterns in Propositional Logic	2	-	-
4.5	**First Order Logic:** Representation Revisited, Syntax and Semantics of First Order logic	2	-	-
4.6	Using First Order logic.	1	-	-
4.7	**Inference in First Order Logic:** Propositional Versus First Order Inference	1	-	-
4.8	Unification, Forward Chaining	1	-	-
4.9	Backward Chaining, Resolution.	1	-	-
Module–5: Quantifying Uncertainty				
5.1	Acting under Uncertainty	1	-	-
5.2	Probability Notation	1	-	-
5.3	Inference using Full Joint Distributions	1	-	-
5.4	Independence	1	-	-
5.5	Baye's Rule and its use	1	-	-
5.6	Wumpus World Revisited	1	-	-
Total No. of Lecture Hours		**40**	-	-
Total No. of Tutorial Hours			**00**	-
Total No. of Practical Hours				**00**

Textbook:

1. StuartJ.RussellandPeterNorvig,ArtificialIntelligence,3rdEdition,Pearson,2015

Reference Book:

1. ElaineRich,KevinKnight,ArtificialIntelligence,3rdedition,TataMcGrawHill,2013
2. GeorgeFLugar,ArtificialIntelligenceStructureandstrategiesforcomplex,Pearson Education, 5th Edition, 2011

Online Resources:

1. Artificial Intelligence: https://nptel.ac.in/courses/106105077

Code: BCS516C	**Course**: **UNIX System Programming**
Credits: 3	**L:T:P - 3:0:0**
SEE: 50 Marks	**CIE: 50 Marks**
SEE Hours: 03 hour	**Max. Marks: 100**

Prerequisites if any	NIL
Learning objectives	Develop the ability to writing efficient and robust UNIX applications, utilizing system calls, libraries, and tools effectively to develop scalable and portable software solutions for diverse computing environments.

Course Outcomes:

On the successful completion of the course, the student will be able to

COs	**Course Outcomes**	**Bloom's level**
CO1	Describe Unix Architecture, File system and use of Basic Commands	Understand
CO2	Build an application/service over a Unix system.	Apply
CO3	Categorize, compare and make use of Unix System Calls	Analyze
CO4	Analyze Shell Programming and to write Shell Scripts	Analyze

Mapping with POs and PSOs:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12		PSO 1	PSO 2
CO1	3	-	-	-	2	-	-	-	-	-	-	-		-	-
CO2	3	-	-	-	2	-	-	2	-	-	-	-		2	3
CO3	2	2	2	-	2	-	-	-	-	-	-	-		3	2
CO4	2	2	2	-	3	-	-	-	-	-	-	-		3	2

Mapping Strength: Strong– 3 Medium – 2 Low – 1

Course Structure

Sl.No	Modules	No. of Lecture Hours	No. of Tutorial Hours	No. of Practical Hours
Module – 1				
1.1	Unix Components/Architecture. Meaning of Internal and external commands.	1		
1.2	Features of Unix. The UNIX Environment and UNIX Structure	1		
1.3	Posix and Single Unix specification. General features of Unix commands/ command structure. Command arguments and options. Basic Unix commands such as echo, printf, ls, who, date, passwd, cal, Combining commands.	1		
1.4	The type command: knowing the type of a command and locating it. The root login. Becoming the super user: su command.	1		
1.5	Naming files. Basic file types/categories. Organization of files. Hidden files. Standard directories. Parent child relationship.	1		
1.6	The home directory and the HOME variable. Reaching required files- the PATH variable, manipulating the PATH, Relative and absolute pathnames. Directory commands – pwd, cd, mkdir, rmdir commands.	2		
1.7	The dot (.) and double dots (..) notations to represent present and parent directories and their usage in relative path names. File related commands – cat, mv, rm, cp, wc and od commands.	1		
Module – 2				
2.1	File attributes and permissions	2		
2.2	The shells interpretive cycle	2		
2.3	Connecting commands	2		
2.4	Shell Programming	2		
Module – 3				
3.1	UNIX File APIs	1		
3.2	UNIX Processes and Process Control	2		
3.3	The Environment of a UNIX Process	1		
3.4	Process Control	2		

Module – 4				
4.1	Changing User IDs and Group IDs, Interpreter Files, system Function, Process Accounting, User Identification, Process Times, I/O Redirection.	3		
4.2	Overview of IPC Methods	2		
4.3	Shared Memory	3		
Module – 5				
5.1	Signals: The UNIX Kernel Support for Signals, signal, Signal Mask, sigaction, The SIGCHLD Signal and the waitpid Function, The sigsetjmp and siglongjmp Functions, Kill, Alarm, Interval Timers, POSIX.lb Timers.	4		
5.2	Daemon Processes: Introduction, Daemon Characteristics, Coding Rules, Error Logging, Client-Server Model.	3		
	Total No. of Lecture Hours	**40**	**-**	**-**
		Total No. of Tutorial Hours	**00**	**-**
			Total No. of Practical Hours	**00**

Text books:

1. Sumitabha Das., Unix Concepts and Applications., 4thEdition., Tata McGraw Hill (Chapter 1,2 ,3,4,5,6,8,13,14)

2. W. Richard Stevens: Advanced Programming in the UNIX Environment, 2nd Edition, Pearson Education, 2005 (Chapter 3,7,8,10,13,15)

3. Unix System Programming Using C++ - Terrence Chan, PHI, 1999. (Chapter 7,8,9,10)

Reference books:

1. M.G. Venkatesh Murthy: UNIX & Shell Programming, Pearson Education.

2. Richard Blum , Christine Bresnahan : Linux Command Line and Shell Scripting Bible, 2ndEdition, Wiley,2014.

Code:BCS586 **Course :Minor Project**

Credits:1 **L:T:P : 0:0:2**

CIE:50 Marks **Max. Marks: 50**

Pre-requisites if any	Form a Team, define Project objectives, literature survey, identify resources, define Methodology, documentation.
Learning objectives	● Clearly state the goals and objectives of the project. ● Determine the scope and expected outcomes. ● Collaborate with classmates for the project teamwork. ● Assign roles and responsibilities based on each team member's strengths. ● Identify potential risks and challenges that might arise during the project. ● Maintain detailed records of your project, design, and development process. ● Prepare to present your findings and results clearly and comprehensively.

Course outcomes:

On the successful completion of the course, the student will be able to

COs	**Course Outcomes**	**Bloom'slevel**
CO1	Formulate and analyze a real-world problem to define project requirements and objectives.	Analyze
CO2	Apply appropriate computing techniques, tools, and technologies to design and implement a workable solution.	Apply
CO3	Develop and test a functional prototype/system by working collaboratively in a team.	Evaluate
CO4	Document and present the project outcomes effectively with consideration of ethical and societal aspects.	Create

Mappingwith POs andPSOs:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12		PSO1	PSO2
CO1	3	3	2	2	3	1	1	1	2	2	2	2		2	1
CO2	3	2	3	2	3	1	1	2	2	2	2	2		3	3
CO3	3	2	3	2	3	2	1	1	3	3	2	2		2	3
CO4	3	1	1	1	3	2	2	3	2	3	2	2		2	2

Mapping Strength: Strong–3 Medium–2 Low–1

Course Code: BRMCS557 **Course name: Research Methodology and IPR**

Credits: 02 **L:T:P - 2:0:0**

SEE: 50 Marks **CIE: 50 Marks**

SEE Hours: 2 **Max. Marks:100**

Prerequisites if any	NIL
Learning objectives	1. To gain knowledge on research methodology and explain the technique of formulating a research problem. 2. To understand various research designs and different types of data collections. 3. To understand various sampling designs and its characteristics. 4. To acquire the knowledge on report writing and various concepts of IPR.

Course Outcomes:

On the successful completion of the course, the student will be able to

COs	**Course Outcomes**	**Bloom's level**
CO1	Discuss different aspects of research Methodology	Understand
CO2	Apply various research design and methods of data collection	Apply
CO3	Analyze various sampling design and its characteristics	Analyze
CO4	Evaluate research designs for various real time problems	Evaluate

Mapping with POs and PSOs:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12		PSO1	PSO2
CO1	3	3	3	2	2	-	3	2	-	-	2	3		3	2
CO2	3	2	3	2	2	2	-	2	2	-	2	3		3	-
CO3	3	2	3	2	3	2	2	-	2	-	2	3		3	-
CO4	3	3	2	2	2	2	3	3	2	2	2	3		2	1

Mapping Strength: Strong– 3 Medium – 2 Low – 1

Course Structure

Sl No	Modules	No. of Lecture Hours	No. of Tutorial Hours	No. of Practical Hours
Module – 1: Research Methodology: An Introduction, Defining the research problem				
1.1	Meaning of Research, Objectives of Research Motivation in Research	1	-	-
1.2	Types of Research, Research Approaches, Research Methods versus Methodology, Research and Scientific Method.	2	-	-
1.3	Research Process.	3	-	-
1.4	Criteria of good research, what is a Research Problem?	1	-	-
1.5	Selecting the Problem, Necessity of Defining the Problem, Technique Involved in Defining a Problem.	2	-	-
Module – 2: Research Design, Methods of data collection				
2.1	Meaning of Research, Design, Need for Research Design	1	-	-
2.2	Features of a Good Design, Important Concepts Relating to Research Design, Different Research Designs.	2	-	-
2.3	Collection of Primary Data, Observation Method, Interview Method Collection of Data through Questionnaires.	3	-	-
2.4	Collection of Data through Schedules, Difference between Questionnaires and Schedules ,Some Other Methods of Data Collection, Collection of Secondary Data, Selection of Appropriate Method for Data Collection	3	-	-
Module – 3:Report writing and Introduction to Intellectual Property				
3.1	Significance of Report Writing, Different Steps in Writing Reports,Layout of the Research Report, Types of Reports, Precautions for Writing Research Reports, Conclusions	1	-	-
3.2	Role of IP in the Economic and Cultural Development of the Society, IP Governance, Conditions for Obtaining a Patent Protection, To Patent or Not to Patent an Invention, Rights Associated with Patents, Enforcement of Patent Rights, Inventions Eligible for Patenting, Process of Patenting	4	-	-
3.3	Classes of Copyrights, Criteria for Copyright Ownership of Copyright, Copyrights of the Author, Copyright Infringements, Copyright Registration, Copyright Symbol Validity of Copyright.	2	-	-
	Total No. of Lecture Hours	**25**	-	-
		Total No. of Tutorial Hours	**00**	-
			Total No. of Practical Hours	**00**

Textbook:

1. Kothari, C.R., (2014), Research Methodology, New Age International second revised edition
2. Prof. Rupinder Tewari, Ms. Mamta Bhardwaj (2021) Intellectual Property A Primer for Academia.

Reference Book:

1. Ranjit Kumar, (2011). Research Methodology a step by step guide for beginners, Sage Publications
2. Chawla, Deepak &Sondhi, Neena (2011). Research methodology: Concepts and Cases, Vikas Publishing House Pvt. Ltd. Delhi.

Online Resources:

1. https://onlinecourses.swayam2.ac.in/cec23_ge07/preview
2. https://onlinecourses.nptel.ac.in/noc22_ge08/preview

The National Institute of Engineering

Scheme of Teaching & Examination (2022 Scheme)

Department: Computer Science and Engineering (BE in CS&E)

B.E. 2023 Admitted Batch

VI Semester

Sl.No	**Type of Course**	**Course Code**	**Course Title**	**Teaching Department (TD)**	**Question Paper setting Board (PSB)**	**Teaching Hrs/Week**				**Examination**				**Credits**
						L	**T**	**P**	**S**	**Duration in Hours**	**CIE Marks**	**SEE Marks**	**Total Marks**	
1	IPCC	BCS601	Internet of Things	CS	CS	3	0	2		3	50	50	100	**4**
2	PCC	BCS602	Machine Learning	CS	CS	3	0	0		3	50	50	100	**3**
3	PEC	BCS613x	Professional Elective Course - Group II	CS	CS	3	0	0		3	50	50	100	**3**
4	OEC	BCS654x	Open Elective Course - Group I	CS	CS	3	0	0		3	50	50	100	**3**
5	PCC	BCS605	Cloud Computing	CS	CS	3	2	0		3	50	50	100	**4**
6	PCC	BCS606	Distributed Systems	CS	CS	3	0	0		3	50	50	100	**3**
7	PCCL	BCSL607	Machine Learning lab	CS	CS	0	0	2		2	50	50	100	**1**
8	AEC/ SDC	BCS657X	Ability Enhancement Course / Skill Development Course V	CS	CS	If the course is a Theory					50	-	50	**1**
						1	0	0		1				
						OR								
						If the course is a Laboratory								
						0	0	2		2				
9	MC	BNSK658	National Service Scheme (NSS)	NSS Coordinator		0	0	2		_	100	_	100	**0**
		BPEK658	Physical Education (PE) (Sports & Athletics)	PED										
		BYOK658	Yoga	Yoga Teacher										
		BMUK658	Music	Music Teacher										
10	MC	BIKK259	Indian Knowledge Systems	Humanities		1	0	0	0	0	50	-	50	**0**
										Total	**500**	**350**	**850**	**22**

Professional Elective Course - Group II

BCS613A	Blockchain Technology	BCS613D	Advanced Java
BCS613B	Computer Vision	BCS613E	Entrepreneurship and Enterprise Resource Planning
BCS613C	Compiler Design		

Open Elective Course - Group I

BCS654A	Introduction to Java Programming	BCS654D	Introduction to AI
BCS654B	Introduction to Web Technologies	BCS654E	Introduction to Data Science
BCS654C	Mobile Application Development	BCS654F	Introduction to Blockchain Technology

Ability Enhancement Course / Skill Enhancement Course-V

BCSL657A	Progressive App Development	BCS657D	Devops
BCSL657B	Tosca – Automated Software Testing		
BCS657C	Agile Development		

Code: BCS601 **Course: Internet of Things**

Credits: 4 **L:T:P - 3:0:2**

SEE: 100 Marks **CIE: 50 Marks**

SEE Hours: 3 **Max. Marks: 100**

Prerequisites if any	Fundamentals of Networking and Basic Programming Languages (C, C++ &Python)
Learning objectives	● To understand the fundamentals of the Internet of Things and its building blocks along with their characteristics ● To analyze the role of data, data analytics, and IoT endpoints in IT and OT ● To assess the impact of IoT in real world

Course Outcomes:

COs	**Course Outcomes**	**Bloom's level**
CO1	Describe the IoT architecture, characteristics, and the role, types, and selection criteria of sensors and actuators in IoT systems.	Understand
CO2	Implement Arduino programming concepts to interface sensors and actuators for functional IoT applications.	Apply
CO3	Compare and analyze various IoT communication and network protocols for efficient device interoperability.	Analyze
CO4	Evaluate IoT data analytics techniques, challenges, and security measures to assess and improve real-world IoT applications across domains such as home automation, smart cities, agriculture, healthcare, and defense.	Evaluate
CO5	Develop IoT-based prototypes using Arduino by integrating sensors, actuators, and cloud services for real-time monitoring and control applications	Create

On the successful completion of the course, the student will be able to

Mapping with POs and PSOs:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	3	1	1	1		1	2	1	1	2	1	1
CO2	3	3	3	1	3	1		2	2	1	1	2	1	1
CO3	3	3	3	1	3	1	1	2	2	2	1	2	1	2
CO4	3	3	3	1	3	1	1	2	2	2	2	2	2	2
CO5	3	3	3	2	3	1	1	2	2	3	2	1	2	2

Mapping Strength: Strong– 3 Medium – 2 Low – 1

Course Structure

Sl.No	Modules	No. of Lecture Hours	No. of Tutorial Hours	No. of Practical Hours
Module – 1: Introduction to Internet of Things and Sensors				
1.1	Introduction to IoT: Definition, Characteristics of IoT	1	-	-
1.2	IoT Architecture	1		
1.3	Sensors, Types of Sensors	3		
1.4	Characteristics and Selection of Sensors	3	-	-
Module – 2: Actuators and controllers				
2.1	Actuators and Types of Actuators	2	-	-
2.2	Characteristics and Selection of Actuators	2		
2.3	Introduction to Arduino: Architecture and GPIO pin structure.	2		
2.4	Fundamentals of Arduino programming	2	-	-
Module – 3: Communication Technology				
3.1	M2M, Difference Between IoT and M2M	3	-	-
3.2	Communication protocols: LoRaWAN, WIFI	2	-	-
3.3	Data and Network Protocols: MQTT, COAP& HTTPS	3		
Module – 4: Data and Analytics for IoT				
4.1	An introduction to data analytics for IoT.	1	-	-
4.2	Structured versus unstructured data, IoT data analytics overview.	2		
4.3	IoT data analytics challenges, Edge streaming analytics.	2	-	-
4.4	Securing IoT: OT security, common challenges in OT security.	3		
Module – 5:IoTCase Studies				
5.1	Case studies: Home Automation	1	-	-
5.2	Smart Cities	1	-	-

5.3	Agriculture	2		
5.4	Healthcare	2		
5.5	Defense	2		
IoT Lab Programs				
1	Introduction to working IDE and its practice			1
2	Sense the available networks using Arduino			1
3	Measure the distance using ultrasonic sensors and make an LED Blink using Arduino			1
4	Detect the vibration of an object using Arduino			1
5	Connect with the available Wi-Fi using Arduino			1
6	Temperature notification using Arduino			1
7	LDR to vary light intensity of LED using Arduino			1
8	Creation of Things Speak Account			1
Total No. of Lecture Hours		**40**	**-**	**-**
Total No. of Tutorial Hours			**00**	**-**
Total No. of Practical Hours				**10**

Textbook:

1. ArshdeepBahga, Vijay Madisetti, “Internet of Things- A Hands-On Approach”, Universities press, 2014.
2. David Hanes, Gonzalo Salgueiro, Rob Barton " IoT Fundamentals: Networking Technologies, Protocols and Use Cases for the Internet of Things", 2019

Reference Book:

1. Srinivasa K G, “Internet of Things”, CENGAGE Leaning India, 2017
2. Raj Kamal, Internet of Things: Architecture and Design Principles, McGraw Hill Education, 2017

Code: BCS602 **Course: Machine Learning**

Credits:3 **L:T:P-3:0:0**

SEE: 50Marks **CIE:50 Marks**

SEEHours:3 **Max.Marks:100**

Prerequisites ifany	Basic knowledge of programming (preferably Python), linear algebra, probability, and statistics.
Learning objectives	1.To provide understanding of ML fundamentals, types, processes, and applications. 2.To provide knowledge of dataset preparation, statistical analysis, and feature engineering. 3.To provide exposure to similarity-based learning models like KNN and Nearest Centroid. 4.To provide knowledge of regression, decision tree, and Bayesian models for prediction. 5.To provide understanding of clustering methods and reinforcement learning for problem-solving.

Course Outcomes:

On the successful completion of the course, the student will be able to

COs	**Course Outcomes**	**Bloom's level**
CO1	Describe core concepts, processes, and applications of machine learning.	Understand
CO2	Apply supervised, unsupervised and reinforcement learning techniques to solve problems.	Apply
CO3	Examine datasets and learning algorithms using statistical and similarity-based methods.	Analyze
CO4	Assess machine learning models for accuracy, interpretability, and suitability.	Evaluate

Mapping with POs and PSOs:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	**3**	**2**	-	-	**2**	-	-	-	-	**1**	-	**2**	**3**	**3**
CO2	**3**	**3**	**3**	-	**3**	-	-	-	-	**2**	**2**	-	**3**	**3**
CO3	**3**	**3**	**2**	**3**	**2**	-	-	-	-	**2**	-	**3**	**3**	**3**
CO4	**3**	**3**	**3**	**2**	-	-	-	-	**3**	-	**3**	**3**	**3**	**3**

Mapping Strength: Strong– 3 Medium–2 Low –1

Course Structure

Sl.No	Modules	No. of Lecture Hours	No. of Tutorial Hours	No. of Practical Hours
Module – 1				
1.1	**Introduction:** Need for Machine Learning, Machine Learning Explained	1	-	-
1.2	Machine Learning in Relation to other Fields	1	-	-
1.3	Types of Machine Learning	3	-	-
1.4	Challenges of Machine Learning, Machine Learning Process	1	-	-
1.5	Machine Learning Applications	1	-	-
Module–2				
2.1	**Understanding Data:** Introduction	2	-	-
2.2	Big Data Analysis Framework	1	-	-
2.3	Descriptive Statistics	1	-	-
2.4	Bivariate Data and Multivariate Data	1	-	-
2.5	Multivariate Statistics	1	-	-
2.6	Essential Mathematics for Multivariate Data (Only Linear Systems and Gaussian Elimination for Multivariate Data, Matrix Decompositions	2	-	-
2.7	Feature Engineering and Dimensionality Reduction Techniques (only Introduction)	1	-	-
Module–3				
3.1	**Basic Learning Theory:** Design of Learning System,	1	-	-
3.2	Introduction to Concept of Learning	1	-	-
3.3	Find-S Algorithm	1		
3.4	**Similarity-based Learning:** Nearest-Neighbor Learning	1	-	-
3.5	Weighted K-Nearest-Neighbor Algorithm	1	-	-
3.6	Nearest Centroid Classifier	1	-	-
Module – 4				
4.1	**Regression Analysis:** Introduction to Regression, Introduction to Linear Regression	1	-	-
4.2	Multiple Linear Regression	1	-	-
4.3	Polynomial Regression	1	-	-
4.4	Logistic Regression	1	-	-
4.5	**Decision Tree Learning:** Introduction to Decision Tree Learning Model	1	-	-
4.6	Decision Tree Induction Algorithms (Only ID3 Tree construction)	2	-	-
4.7	**Bayesian Learning:** Introduction to Probability-based Learning, Fundamentals of Bayes Theorem	1	-	-
4.8	Classification Using Bayes Model (Only Naïve Bayes Algorithm)	2	-	-
Module–5				
5.1	**Clustering Algorithms:** Introduction to Clustering Approaches	1	-	-
5.2	Hierarchical Clustering Algorithms (Only Single Linkage or MIN Algorithm and Complete Linkage or MAX or Clique)	2	-	-
5.3	Partitional Clustering Algorithm; K-Means Algorithm	2	-	-
5.4	**Reinforcement Learning:** Overview of Reinforcement Learning, Scope of Reinforcement Learning	1	-	-

5.5	Reinforcement Learning as Machine Learning, Components of Reinforcement Learning	1	-	-
5.6	Q-Learning	1	-	-
	Total No. of Lecture Hours	**40**	-	-
		Total No. of Tutorial Hours	**00**	-
			Total No .of Practical Hours	**00**

Text Book

1. S Sridhar, M Vijayalakshmi, “Machine Learning”, OXFORD University Press 2021, First Edition.

Module 1: 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7

Module 2: 2.1, 2.3, 2.4, 2.6, 2.7, 2.8.1, 2.8.2, 2.10(only introduction)

Module 3: 3.3, 3.4, 4.2, 4.3, 4.4

Module 4: 5.1, 5.3, 5.5, 5.6, 5.7, 6.1, 6.2.1, 8.1, 8.2, 8.3.1

Module 5: 13.1, 13.3.1, 13.3.2, 13.4, 14.1, 14.2, 14.3, 14.4, 14.9

Reference Books

1. Murty, M. N., and V. S. Ananthanarayana. Machine Learning: Theory and Practice, Universities Press,

2024.

2. T. M. Mitchell, “Machine Learning”, McGraw Hill, 1997.

3. Burkov, Andriy. The hundred-page machine learning book. Vol. 1. Quebec City, QC, Canada: Andriy

Burkov, 2019.

Web links and Video Lectures (e-Resources):

1. https://www.universitiespress.com/resources?id=9789393330697
2. https://www.drssridhar.com/?page_id=1053
3. Machine Learning Tutorials: https://www.geeksforgeeks.org/machine-learning/
4. Machine Learning Tutorials: https://www.tutorialspoint.com/machine_learning/index.htm
5. Python for Machine Learning: https://www.w3schools.com/python/python_ml_getting_started.asp
6. Introduction to Machine Learning: https://onlinecourses.nptel.ac.in/noc22_cs29/preview

Code: BCS613A **Course: Blockchain Technology**

Credits: 3 **L:T:P 3:0:0**

SEE: 50 Marks **CIE: 50 Marks**

SEE Hours: 3 **Max. Marks:100**

Prerequisites if any	NIL
Learning objectives	● Understand blockchain architecture and use Hashing. ● Describe different consensus algorithms. ● Design decentralized applications using blockchain.

Course Outcomes:

On the successful completion of the course, the student will be able to

Cos	**Course Outcomes**	**Bloom's level**
CO1	Explain the foundational concepts of blockchain technology, including its architecture, consensus mechanisms, and applications in distributed systems and supply chain management.	Understand
CO2	Apply blockchain decentralization and cryptographic techniques to analyze decentralized ecosystems, Smart Contracts, and digital signatures.	Analyze
CO3	Explain the structure and functioning of Bitcoin transactions, blocks, and networks, and Analyze the deployment and application of smart contracts within blockchain systems.	Analyze
CO4	Illustrate the architecture and functioning of the Ethereum blockchain, including transactions, smart contracts, EVM, and consensus.	Analyze
CO5	Analyze the architecture and protocols of enterprise blockchain platforms, including Hyperledger Fabric, Sawtooth Lake, and Corda.	Analyze

Mapping with POs and PSOs:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3					2			2	2		2	2	2
CO2	3	3		3	2	2	2		2	2		2	2	2
CO3	3	3		2	2	2	2		2	2		2	2	2
CO4	3	3		2	2	2	2		2	2		2	2	2
CO5	3					2	2		2	2		2	2	2

Mapping Strength: Strong– 3 Medium – 2 Low – 1

Course Structure

Sl. No.	Module Name	No. of Lecture Hours	No. of Tutorial Hours	No. of Practical Hours
Module – 1: Introduction to Blockchain				
1.1	Introduction, Concepts of Blockchain, History, Definition of Blockchain	1	-	-
1.2	Fundamentals of Blockchain, Characteristics of Blockchain, Consensus in Trust-Building Exercise	1	-	-
1.3	Public, Private, and Hybrid Blockchains, DLT, DLT Applications and databases	2	-	-
1.4	Architecture of Blockchain, Transactions	1	-	-
1.5	Blocks, chaining blocks, transactions, Value proposition of Blockchain Technology	2	-	-
Module – 2: Decentralized System &Hash Functions				
2.1	Introduction, Distributed Decentralized Databases, Decentralized Enterprise	1	-	-
2.2	Decentralization	1	-	-
2.3	Disintermediation	1	-	-
2.4	Hash Functions: Hashing	1	-	-
2.5	MAC	1	-	-
2.6	SHA-1, SHA-256	1	-	-
2.7	Distributed Hash Tables	2	-	-
2.8	Hashing and Data Structures, Hashing in Blockchain Mining	1	-	-
Module – 3: Consensus				
3.1	Introduction, Consensus Approach, Consensus Algorithms-POW	1	-	-
3.2	POS, POA, POET	2	-	-
3.3	POB, Byzantine Agreement Methods, PBFT, DBFT	2	-	-
3.4	Symmetric Cryptography, Asymmetric Cryptography	2	-	-

Module – 4: Bitcoins & Decentralized Applications				
4.1	Introduction, Working of Bitcoin, Merkle Trees, Bitcoin Block Structure	1	-	-
4.2	Bitcoin Address	1	-	-
4.3	Bitcoin Transactions	1	-	-
4.4	Bitcoin Network, Bitcoin Wallets, Bitcoin Payments, Bitcoin Clients, Bitcoin Supply	2	-	-
4.5	Decentralized Applications- Introduction, Today’s Web Applications Requirement	1	-	-
4.6	Mining in Blockchain Bitcoin, Blocks Validation and identification, Bitcoins Creation	1	-	-
4.7	Mining Hardware, Mining Software, Running Miner Software, Executing several miners Reasons for Bitcoin Mining	2	-	-
Module 5: Ethereum and Smart Contract				
5.1	Introduction , Ethereum, History, Ethereum Virtual Machine	1	-	-
5.2	Working of Ethereum, Ethereum Clients, Ethereum Key Pairs	1	-	-
5.3	Ethereum Address, Ethereum Wallets	1	-	-
5.4	Ethereum Transactions, Ethereum Languages	1	-	-
5.5	Ethereum Development Tools	1	-	-
5.6	Smart Contracts- Introduction, SMART Contract	1	-	-
5.7	Benefits of SMART CONTRACT, Absolute and immutable, Contractual Confidentiality , Law Implementation & settlement, characteristics, Hyperledger fabric Architecture	1	-	-
5.8	Supply Chain Management	1	-	-
	Total No. of Lecture Hours	**40**		
		Total No. of Tutorial Hours	**00**	
			Total No. of Practical Hours	**00**

Textbook:

1. Mastering Blockchain- Distributed Ledger Technology, Decentralization, and Smart Contracts Explained,
   2nd Edition by Imran Bashir · 2018
2. Blockchain Technology: Concepts and Applications, by Kumar Saurabh & Ashutosh Saxena,
   WILEY Emerging Technology Series, First Edition, 2020

Reference Book:

1. Blockchain from concepts to execution, DebajaniMohanty, Second revised edition, BPB Publication, 2021
2. A Practical Guide To Blockchain And Its Applications , PARIKSHIT JAIN , Blooms Bury, Edition 2019

Online Resources:

1. https://onlinecourses.nptel.ac.in/noc22_cs44/ - Blockchain and its Applications
2. https://www.youtube.com/watch?v=qOVAbKKSH10, Blockchain Technology Explained (2 Hour Course)

Code: BCS613B **Course: Computer Vision**

Credits: 3 **L:T:P - 3:0:0**

SEE: 50 Marks **CIE: 50 Marks**

SEE Hours: 3 **Max. Marks: 100**

Prerequisites if any	Linear algebra, calculus, and probability
Learning objectives	• Equip students with the skills to understand and apply key concepts in image formation, processing, and feature detection techniques. • Enable students to implement advanced segmentation methods and perform precise feature-based alignment and calibration for real-world applications.

Course Outcomes:

On the successful completion of the course, the student will be able to

Cos	**Course Outcomes**	**Bloom's level**
CO1	Describe image formation principles and basic image processing techniques.	Understanding
CO2	Explain feature detection and match features in images	Understanding
CO3	Apply different image segmentation methods	Apply
CO4	Explain 2D and 3D alignment along with calibration techniques.	Understanding

Mapping with POs and PSOs:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12		PSO1	PSO2
CO1	2	2	-	-	2	-	-	-	-	-	-	-		-	-
CO2	3	2	2	3	2	-	-	-	-	-	-	2		3	3
CO3	3	3	3	3	3	-	-	-	-	-	-	2		-	2
CO4	3	3	3	-	3	-	-	-	-	-	-	-		-	-

Mapping Strength: Strong– 3 Medium – 2 Low – 1

Course Structure

Sl.no	Modules	No. of Lecture Hours	No. of Tutorial Hours	No. of Practical Hours
Module – 1: Introduction				
1.1	Definition of Computer Vision, Image Formation: Geometric primitives and transformations- Geometric primitives	2	-	-
1.2	2D transformations, 3D transformations	2	-	-
1.3	Photometric image formation – Lighting, Reflectance and shading	2	-	-
1.4	The Digital Camera – Sampling and aliasing	2	-	-
Module – 2: Image Processing				
2.1	Point Operators – Pixel transforms, Color transforms, Compositing and matting	2	-	-
2.2	Histogram equalization, Tonal Adjustment	1	-	-
2.3	Linear Filtering – Separable filtering, Examples of linear filtering	2	-	-
2.4	Band pass and steerable filters	1	-	-
2.5	More neighborhood operators- Nonlinear filtering, Morphology, Distance transforms, Connected components	2	-	-
Module – 3: Feature detection and matching				
3.1	Points and patches – Feature detectors, Feature descriptors, Feature matching, Feature tracking, applications	3	-	-
3.2	Performance-driven animation; Edges – Edge detection, Edge linking, Application: Edge editing and enhancement	3	-	-
3.3	Lines – Successive approximation, Hough transforms, Vanishing points	2	-	-
Module – 4: Segmentation				
4.1	Active Contours – Snakes, Dynamic snakes and Condensation, Scissors, Level Sets, Application: Contour tracking and rotoscoping	2	-	-
4.2	Split and merge – Watershed, Region splitting (divisive clustering), Region merging (agglomerative clustering)	2	-	-
4.3	Graph based segmentation, Probabilistic aggregation	2	-	-
4.4	Mean shift and mode finding – K-means and mixtures of Gaussians, Mean Shift, Normalized cuts; Graph cuts and energy-based methods.	2	-	-
Module – 5: Feature- based alignment				
5.1	2D and 3D feature-based alignment – 2D alignment using least squares, Application-Panography	2	-	-
5.2	Iterative algorithms, Robust least squares and RANSAC	2	-	-
5.3	3D alignment; Pose estimation – Linear algorithms, Iterative algorithms, Application: Augmented reality	2	-	-
5.4	Geometric intrinsic calibration - Calibration patterns, Vanishing Points.	2	-	-
	Total No. of Lecture Hours	**40**	-	-
	Total No. of Tutorial Hours		**00**	-
	Total No. of Practical Hours			**00**

Textbook:

1. Computer Vision: Algorithms and Applications (CVAA), Richard Szeliski. Springer, 2010.
2. Image Processing, Analysis, and Machine Vision, Sonka, Hlavac, and Boyle. Thomson

Online Resources:

1. NPTEL Course Link: https://onlinecourses.nptel.ac.in/noc24_ee38/preview
2. MIT OpenCourseWare

   Course Link: https://ocw.mit.edu/courses/6-801-machine-vision-fall-2020/pages/lecture-notes/

Code: BCS613C	**Course: Compiler Design**
Credits: 3	**L:T:P - 3:0:0**
SEE: 50 Marks	**CIE: 50 Marks**
SEE Hours: 3	**Max. Marks: 100**

Prerequisites if any	Data Structures and Automata Theory
Learning objectives	1. Specify and analyze the lexical, syntactic and semantic structures of advanced language features. 2. Separate the lexical, syntactic and semantic analysis into meaningful phases for a compiler to undertake language translation. 3. Gain appropriate knowledge build a compiler that converts from a non-trivial high level language to machine code.

Course Outcomes:

On the successful completion of the course, the student will be able to

COs	Course Outcomes	Bloom's level
CO1	Describe the concepts and phases of compiler	Understand
CO2	Apply compiler design techniques to solve problems related to tokenization, parsing, storage and code generation	Apply
CO3	Analyze syntax directed translation and intermediate code forms for their suitability for target architectures	Analyze
CO4	Evaluate compiler techniques in terms of performance, complexity, and grammar coverage	Evaluate

Mapping with POs and PSOs:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12		PSO1	PSO2
CO1	3	2	-	-	-	-	-	-	-	-	-	2		-	-
CO2	3	3	2	3	3	-	-	-	3	-	2	2		3	2
CO3	3	3	2	3	2	2	2	-	2	-	1	2		3	2
CO4	3	2	3	3	2	2	2	-	3	-	1	2		3	2

Mapping Strength: Strong– 3 Medium – 2 Low – 1

Course Structure

Sl.no	Modules	No. of Lecture Hours	No. of Tutorial Hours	No. of Practical Hours
Module – 1: Lexical Analysis and Top Down Parsing				
1.1	Compilers, Compiler-construction tools	1	-	-
1.2	The phases of a compiler	1	-	-
1.3	The role of the lexical analyzer, Input buffering	1	-	-
1.4	Specification of tokens, Recognition of tokens	1	-	-
1.5	The role of the parser, Top-down parsing	4	-	-
Module – 2: Bottom Up Parsing				
2.1	Bottom up parsing, Types of Bottom up parsing	1	-	-
2.2	Operator-precedence parsing	1	-	-
2.3	LR parsers	2	-	-
2.4	SLR parsers	2	-	-
2.5	LALR parsers	2	-	-
Module – 3: Syntax-Directed Translation				
3.1	Syntax-directed definitions, Construction of syntax trees	1	--	-
3.2	Bottom-up evaluation of S-attributed definitions	1	-	-
3.3	L-attributed definitions	1	-	-
3.4	Intermediate languages	1	-	-
3.5	Declarations, Assignment statements	1	-	-
3.6	Boolean expressions, Case statements	1	-	-
3.7	For and While Statements	1	-	-
3.8	Back Patching, Procedure calls	1	-	-
Module – 4: Run-Time Environments				

4.1	Source language issues	1	-	-
4.2	Storage organization	1	-	-
4.3	Storage-allocation strategies	1	-	-
4.4	Access to nonlocal names, parameter passing	2	-	-
4.5	Symbol tables	1	-	-
4.6	Dynamic storage allocation techniques	2	-	-
Module – 5: Code generation				
5.1	Issues in the design of a code generator,	1	-	-
5.2	The target machine, Run-time storage management	1	-	-
5.3	Basic blocks and flow graphs,	2	-	-
5.4	The dag representation of basic blocks,	1	-	-
5.5	Code-generator generators	1	-	-
5.6	A Simple code generator- Register allocation and assignment	2	-	-
	Total No. of Lecture Hours	**40**	-	-
		Total No. of Tutorial Hours	**00**	-
			Total No. of Practical Hours	**00**

Textbook:

1. Aho, Sethi& Ullman, "Compilers: Principles, Techniques and Tools", Pearson Education, 1986

Reference Book:

1. Principle of Compiler Design, A.V.Aho and J.D. Ullman, Addition – Wesley , 1977
2. J.P. Bennet, "Introduction to Compiler Techniques", Second Edition, McGraw-Hill,2003

Online Resources:

1. https://onlinecourses.nptel.ac.in/noc21_cs07/preview
2. https://online.stanford.edu/courses/soe-ycscs1-compilers

Code: BCS613D **Course: Advanced Java**

Credits: 3 credits **L:T:P – 3:0:0**

SEE:50 Marks **CIE: 50 Marks**

SEE Hours: 3 hours **Max. Marks:100**

Prerequisites if any	basics of java
Learning objectives	To gain knowledge of java framework and develop applications using servlets and JSP To enable interaction with database using JDBC

Course Outcomes:

On the successful completion of the course, the student will be able to

Cos	**Course outcomes**	**Bloom's level**
CO1	Explain the fundamentals of collection framework	Understand
CO2	Demonstrate the fundamental concepts of String operations and Swing applications	Apply
CO3	Design and develop web applications using Java servlets and JSP	Apply
CO4	Apply database interaction through Java database Connectivity	Analyze

Mapping with Pos and PSOs:

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12		PSO1	PSO2
CO1	2	2	3		3	-	-	-	-	-	-	-		-	-
CO2	2	2	3		3	-	-	-	-	-	-	-		2	2
CO3	2	2	3		3	-	-	-	-	-	-	-		3	2
CO4	2	2	3		3									2	2

Mapping Strength: Strong– 3 Medium – 2 Low – 1

Course Structure

Sl.No	Modules	No. of Lecture Hours	No. of Tutorial Hours	No. of Practical Hours
Module – 1: Module Name				
1.1	**The collections and Framework**: Collections Overview, The Collection Interfaces	1	-	-
1.2	The Collection Classes, Accessing a collection Via an Iterator	2	-	-
1.3	Storing User Defined Classes in Collections	1	-	-
1.4	The Random Access Interface, Working With Maps, Comparators	2	-	-
1.5	The Collection Algorithms, Arrays	1	-	-
1.6	The legacy Classes and Interfaces, Parting Thoughts on Collections.	1	-	-
Module – 2: Module Name				
2.1	**String Handling** :The String Constructors, String Length	1	-	-
2.2	Special String Operations	2	-	-
2.3	Character Extraction ,String Comparison, Searching Strings	2	-	-
2.4	Modifying a String, Data Conversion Using valueOf( ),	1	-	-
2.5	Changing the Case of Characters Within a String, joining strings	1	-	-
2.6	Additional String Methods, StringBuffer , StringBuilder	1	-	-
Module – 3: Module Name				
3.1	**Introducing Swing:** The Origin of Swing, Swing Is Built on AWT, Two Key Swing Features.	2	-	-
3.2	The MVC Connection, Components and Containers	1	-	-
3.3	The Swing Packages	1	-	-
3.4	A Simple Swing Application, Event Handling, Painting in Swing	1	-	-
3.5	**Exploring Swing :** JLabel and ImageIcon, JTextField	1	-	-
3.6	The Swing Buttons-JButton, JToggleButton, Check Boxes, Radio Buttons	2	-	-
Module – 4: Module Name				
4.1	**Introducing servlets**: Background, The Life Cycle of a Servlet, Using Tomcat for Servlet Development.	1	-	-
4.2	A simple Servlet, The Servlet API.	1	-	-
4.3	The Jakarta. Servlet Package, Reading Servlet Parameter.TheJakarta.servlet.http package	2	-	-
4.4	Handling HTTP Requests and Responses; Using Cookies; Session Tracking	1	-	-
4.5	Java Server Pages (JSP); JSP tags	1	-	-
4.6	Variables and Objects, Methods	1	-	-
4.7	Control statements, Loops, Request String, Parsing other information	1	-	-

4.8	User sessions, Cookies, Session Objects	1	-	-
Module – 5: Module Name				
5.1	**JDBC Objects**: The Concept of JDBC	1	-	-
5.2	JDBC Driver Types; JDBC Packages	1	-	-
5.3	A Brief Overview of the JDBC process, Database Connection	2	-	-
5.4	Associating the JDBC/ODBC Bridge with the Database; Statement Objects	1	-	-
5.5	ResultSet, Transaction Processing	1	-	-
5.6	Metadata, Data types; Exceptions.	1	-	-
			-	-
	Total No. of Lecture Hours	**40**	-	-
		Total No. of Tutorial Hours	**00**	-
			Total No. of Practical Hours	**00**

Textbook:

1. Herbert schildt :Java the Complete Reference, 12th Edition,Tata McGraw-Hill III
2. Jim Keogh: The complete reference J2EE , Tata McGraw-Hill 2007

Reference Book:

1. Y. Daniel Liang:Introduction to Java Programming 7th edition pearsoneduction ,2007 Stephanie
   Bodoff:THE J2EE Tutorial 2nd edition pearsoneduction ,2004
2. Uttam K. Roy: Advanced Java Programming ,oxford university press 2015

Online Resources:

1. https://onlinecourses.nptel.ac.in/noc24_cs43/preview
2. https://onlinecourses.swayam2.ac.in/aic20_sp13/preview

Code: BCS613E **Course: Entrepreneurship and Enterprise Resource Planning**

Credits: 3 **L:T:P - 3:0:0**

SEE: 50 Marks **CIE: 50 Marks**

SEE Hours: 3 **Max. Marks:100**

Prerequisites if any	None
Learning objectives	1. Explain the importance of action and practice in entrepreneurship 2. To understand the basic concept of ERP systems 3. To study the steps and activities in the ERP life cycle 4. To develop a process driven thinking towards business processes

Course Outcomes:

On the successful completion of the course, the student will be able to

COs	**Course Outcomes**	**Bloom's level**
CO1	Explain the basics of Entrepreneurship and ERP systems	Understand
CO2	Demonstrate the working of ERP with technologies	Analyze
CO3	Analyze the strategic options for ERP identification and adoption and Design the ERP implementation strategies	Analyze
CO4	Describe the various Business Modules of an Enterprise	Understand

Mapping with POs and PSOs:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12		PSO1	PSO2
CO1	3	-	-	-	2	3	3	3	3	3	3	3		-	-
CO2	3	2	-	-	2	2	2	2	3	3	3	3		2	2
CO3	3	2	2	-	2	3	2	1	1	1	1	1		2	2
CO4	3	-	-	-	-	1	1	2	3	3	3	3		2	2

Mapping Strength: Strong– 3 Medium – 2 Low – 1

Course Structure

Sl. No	Modules	No. of Lecture Hours	No. of Tutorial Hours	No. of Practical Hours
Module – 1: Entrepreneurship				
1.1	Introduction to Entrepreneurship : Meaning and concept of entrepreneurship, role of entrepreneurship in economic development,	1		
1.2	Myths about entrepreneurs, agencies in entrepreneurship management and future of entrepreneurship,	1		
1.3	Types of entrepreneurs, the skills/ traits required to be an entrepreneur, Creative and Design Thinking	2		
1.4	The entrepreneurial decision process, skill gap analysis, Importance of communication, barriers and gateways to communication	2		
1.5	Meaning and concept of E-cells, advantages to join E-cell, significance of E-cell, various activities conducted by E-cell	2		
Module – 2: Introduction to ERP				
2.1	Enterprise—An Overview, Business Processes	1		
2.2	Introduction to ERP, Basic ERP Concepts,	1		
2.3	Justifying ERP Investments	2		
2.4	Risks of ERP, Benefits of ERP	2		
Module – 3: ERP AND TECHNOLOGY				
3.1	ERP and Related Technologies, Business Intelligence (BI) and Business Analytics (BA) , E-Commerce and E-Business, ERP, Internet, and WWW	3		
3.2	Business Process Reengineering (BPR) , Data Warehousing and Data Mining, On-line Analytical Processing (OLAP),	3		
3.3	Product Life Cycle Management (PLM), Supply Chain Management (SCM), Customer Relationship Management (CRM)	3		
Module – 4: ERP IMPLEMENTATION				
4.1	Implementation Challenges, ERP Implementation (Transition) Strategies, ERP Implementation Life Cycle,	3		
4.2	Implementation Methodologies ,ERP Deployment MethodsVendors and Consultants , Employees and Employee Resistance , Contracts with Vendors, Consultants, and Employees	3		
4.3	Training and Education, Data Migration , Project Management and Monitoring, Post-Implementation Activities, Success and Failure Factors	3		

	of an ERP Implementation			
Module – 5: Business Modules				
5.1	Business Modules of an ERP Package – Financials, Manufacturing (Production)	2		
5.2	Human Resources Management, Plant Maintenance,	2		
5.3	Materials Management, Quality Management	2		
5.4	Marketing, Sales, Distribution, and Service	2		
Total No. of Lecture Hours		**40**	**-**	**-**
Total No. of Tutorial Hours			**00**	**-**
Total No. of Practical Hours				**00**

Textbook:

1. Business model generation: a handbook for visionaries, game changers, and challengers.“ by A. Osterwalder and Y. Pigneur. John Wiley & Sons, 2010
2. ERP Demystified, Alexis Leon,Tthird Edition, Tata McGraw-Hill, 2007

Code: BCS654A	**Course: Introduction to Java Programming**
Credits:3	**CIE:50 Marks**
L:T:P–3:0:0	**SEE: 50Marks**
SEE Hours:3	**Total Marks:100**

Prerequisites if any	**C programming**
Learning objectives	1. Distinguish Object-Oriented programming paradigm from Procedure- Oriented Programming 2. Use the Java programming language for various programming technologies.

Course Outcomes:

On the successful completion of the course, the student will be able to

COs	**Course Outcomes**	**Bloom's level**
CO1	Develop Java programs using Object – Oriented paradigm	Apply
CO2	Demonstrate Java code utilities in packages, interfaces and String class	Apply
CO3	Apply the concepts of Multithreading and Exception handling to develop efficient and error free codes.	Apply

Mapping with Pos and PSOs:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12		PSO1	PSO2
CO1	3	2	3	2	2	-	-	-	-	-	2	3		3	-
CO2	3	3	3	3	2	-	-	-	-	-	2	3		2	2
CO3	3	3	2	2	2	-	-	-	-	-	2	3		3	1

Mapping Strength: Strong–3 Medium–2 Low–1

Course Structure

Sl. no	Modules	No. of Lecture Hours	No. of Tutorial Hours	No. of Practical Hours
Module–1:Introduction to Object Oriented development				
1.1	What is Object Orientation? What is OO development? OO themes	1	-	-
1.2	Evidence for use fullness of OO development; OO modeling history	1	-	-
1.3	Modeling as Design Technique: Modeling; abstraction	2	-	-
1.4	The three models. Class Modeling: Object and class concepts	2	-	-
1.5	Link and associations concepts; Generalization and inheritance	2	-	-
Module–2: Java & JDK				
2.1	Java’s magic: Bytecode; Java Development Kit(JDK)	1	-	-
2.2	The Java Buzzwords, Simple Java programs	1	-	-
2.3	Data types, arrays, Control Statements	2	-	-
2.4	Classes: Classes fundamentals; Declaring objects	2	-	-
2.5	Constructors	2	-	-
2.6	This keyword, garbage collection	2	-	-
Module–3:Inheritance				
3.1	Inheritance: inheritance basics, using super	3	-	-
3.2	Creating multi-level hierarchy, method overriding	3		-
Module–4:Packages & Interfaces				
4.1	Packages	2	-	-
4.2	Access Protection in packages	2	-	-
4.3	Importing Packages	1	-	-
4.4	Interfaces: references and variables, Default Interface methods	2	-	-
4.5	Exception handling in Java-fundamentals	1	-	-
Module–5:Threads				
5.1	Multi-Threaded Programming: What are threads?	2	-	-
5.2	How to make the classes ‘threadable’	2	-	-
5.3	Extending threads; Implementing ‘runnable’	2	-	-
5.4	Thread Synchronization	2	-	-
Total No. of Lecture Hours		**40**	-	-
Total No. of Tutorial Hours			**00**	-
Total No. of Practical Hours				**00**

Text books:

1. Herbert Schildt, ***Java The Complete Reference-Eleventh Edition***,McGrawHill;11th Edition.
2. Michael Blaha, James Rumbaugh: Object-Oriented Modeling and Design with UML,2nd Edition, Pearson Education/ PHI, 2007

Reference Book:

1. Dr. R. Nageswara Rao, Core Java, An Integrated Approach, DreamtechPress,2016.
2. Mahesh Bhave and Sunil Patekar, "Programming with Java", First Edition, Pearson Education, 2008
3. Rajkumar Buyya, S Thamaraiselvi, xingchenchu, Object oriented Programming with java, Tata McGraw Hill Education Private Limited, 2009
4. Richard A Johnson, an Introduction to Java Programming and Object-Oriented Application Development, Delmar Cengage Learning, 2007.
5. Michael Blaha, James Rumbaugh: Object-Oriented Modeling and Design with UML, 2nd Edition, Pearson Education/ PHI, 2007.

Online Resources:

1. Udacity Free Course https://www.udacity.com/course/java-programming-basics--ud282
2. Introduction To Programming in Java (by Evan Jones, Adam Marcus and Eugene Wu): https://ocw.mit.edu/courses/6-092-introduction-to-programming-in-java-januaryiap-2010/Java Tutorial: https://www.w3schools.com/java/
3. Java Tutorial: https://www.javatpoint.com/java-tutorial

   JavaTutorial:https://www.geeksforgeeks.org/java/

Code: BCS654B **Course: Introduction to Web Technologies**

Credits: 3 **LTP: 3:0:0**

SEE: 50 **CIE:50**

SEE Hours: 3 **Total Marks:100**

Pre requisites if any	NIL
Learning objectives	The goal of this course is to comprehend I. Learn the language of the web. II. Develop dynamic pages using JavaScript, jQuery, AngularJS

Course Outcomes:

On the successful completion of the course, the student will be able to

COs	**Course Outcomes**	**Bloom's level**
CO1	Describe the basics of Internet.	Understand
CO2	Experiment with CSS and Java Script and also develop applications using the same.	Apply
CO3	Design dynamic web applications using Java script and HTML.	Apply
CO4	Demonstrate applications of JQuery for the given problem	Apply
CO5	Demonstrate applications of AngularJS for the given problem	Apply

Mapping with Pos and PSOs:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12		PSO1	PSO2
CO1	3	2	2	-	2	-	-	-	-	-	2	3		2	2
CO2	3	3	2	2	3	-	-	-	-	-	2	3		2	2
CO3	3	3	2	2	3	-	1	-	-	-	2	3		3	3
CO4	3	3	2	2	3	-	1	-	-	-	2	3		3	3
CO5	3	3	2	2	3	-	-	-	-	-	2	3		2	2

Mapping Strength: Strong–3 Medium–2 Low–1

Course Structure

Sl. No	Modules	No. of Lecture Hours	No. of Tutorial Hours	No. of Practical Hours
Module–1 Fundamentals and Introduction to HTML/XHTML				
1.1	Brief introduction the internet, WWW, Web Browsers, and Web Servers	1	-	-
1.2	URLs; MIME; HTTP; Security	1	-	-
1.3	The Web Programmers Toolbox.	1	-	-
1.4	Standard HTML document structure;	1	-	-
1.5	Basic text *mark*-up; Images; Hypertext Links; Lists;	2	-	-
1.6	Tables; Forms.	2	-	-
Module–2 Introduction to CSS and Java Script Introduction				
2.1	CSS Introduction, Levels of stylesheets.	1	-	-
2.2	Selector Forms.	1	-	-
2.3	Font properties ont size, Color properties, Box Model	1	-	-
2.4	Margin, Padding, Background Image	1	-	-
2.5	Basics of JavaScript: Overview of JavaScript; Object orientation and JavaScript;	1	-	-
2.6	General syntactic characteristics; Primitives, operations, and expressions;	1	-	-
2.7	Screen output and keyboard input; Control statements; Object creation and modification;	1	-	-
2.8	Arrays, Functions, Constructor.	1	-	-
Module–3 Dynamic Java Script				
3.1	The Java Script Execution Environment	1	-	-
3.2	Element Access in Java Script Events and Event Handling	1	-	-
3.3	Handling Events from Button Elements Handling Events from Text Box and Password Elemen**t**	1	-	-
3.4	Validating Form Input	1	-	-
3.5	Dynamic Documents with Java Script	1	-	-
3.6	Absolute Positioning, Relative Positioning Static positioning	1	-	-
3.7	Moving Elements Element Visibility	1	-	-
3.8	Locating the Mouse Cursor, Reacting to a Mouse Click	1	-	-
Module–4 Introduction to JQuery				
4.1	Introduction to JQuery	1	-	-
4.2	Syntax	1	-	-
4.3	selectors	2	-	-
4.4	events	2	-	-
4.5	JQuery HTML, JQuery C \SS	2	-	-

Module–5 Introduction to AngularJS				
5.1	Introduction to Angular JS, Directives	1	-	-
5.2	Module, Controller, scope	1	-	-
5.3	Filters	1	-	-
5.4	Events, Services	1	-	-
5.5	Expressions	2	-	-
5.6	Directives in DOM	1	-	-
5.7	AngularJS Forms, Examples	1	-	-
Total No. of Lecture Hours		**40**	-	-
Total No. tutorial Hours			**00**	
Total No. Of practical Hours				**00**

Textbooks:

1. ProgrammingtheWorldWideWeb,RobertW.Sebesta,PearsonEducation – SeventhEdition
2. HTML5BlackBookbyDreamtech,2ndEdition
3. Learn AngularJS in 1 Day: Complete Angular JS Guide with Examples by Krishna Rungta
4. An Introduction to JQuery and JavaScript: A Fast and Simple Way to Start Creating Web Applications by Daniel Green

ReferenceBooks:

1. WebProgrammingByChrisBates,WileyPublications,StudentEdition
2. AngularJSProgrammingByRayYao
3. RobinNixon,“LearningPHP,MySQL&JavaScriptwithJQuery,CSSandHTML5” ,O’ReillyPublications,2015. 4thEdition,
4. ZakRuvalcabaAnneBoehm,“Murach’s HTML5andCSS3”,3rdEdition,Murach’s/ShroffPublishers&DistributorsPvt Ltd, 2016.

Code: BCS654C **Course: Mobile Application Development**

Credits: 3 **L:T:P - 3:0:0**

SEE: 50 Marks **CIE: 50 Marks**

SEE Hours: 03 hour **Max. Marks: 100**

Prerequisites if any	Java
Learning objectives	• Learn to setup Android application development environment • Illustrate user interfaces for interacting with apps and triggering actions • Interpret tasks used in handling multiple activities • Identify options to save persistent application data • Appraise the role of security and performance in Android applications

Course Outcomes:

On the successful completion of the course, the student will be able to

COs	**Course Outcomes**	**Bloom's level**
CO1	Explain Android application by setting up Android development environment.	Apply
CO2	Implement adaptive, responsive user interfaces that work across a wide range of devices	Analyze
CO3	Demonstrate methods in storing, sharing and retrieving data in Android applications	Apply
CO4	Analyze performance of android applications and understand the role of permissions and security.	Analyze

Mapping with POs and PSOs:

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12		PSO 1	PSO2
CO1	3	2	3	-	3	-	-	-	1	-	1	3		2	2
CO2	3	3	3	-	3	-	-	-	1	-	2	3		2	2
CO3	3	2	3	-	3	-	-	-	1	-	2	3		2	2
CO4	3	2	3	-	2	-	-	-	1	-	2	3		2	2

Mapping Strength: Strong– 3 Medium – 2 Low – 1

Course Structure

Sl.No	**Modules**	**No. of Lecture Hours**	**No. of Tutorial Hours**	**No. of Practical Hours**
Module – 1				
1.1	Get started	1	-	-
1.2	Build your first app	2	-	-
1.3	Activities	2	-	-
1.4	Testing, debugging and using support libraries	3	-	-
Module – 2				
2.1	User Interaction	3	-	-
2.2	Delightful user experience	3	-	-
2.3	Testing your UI	2	-	-
Module – 3				
3.1	Background Tasks	3	-	-
3.2	Triggering	3	-	-
3.3	Scheduling and optimizing background tasks	2	-	-
Module – 4				
4.1	All about data	1	-	-
4.2	Preferences and Settings	2	-	-
4.3	Storing data using SQLite	2	-	-
4.4	Sharing datawith content providers	2	-	-
4.5	Loading data using Loaders	1	-	-
Module – 5				
5.1	Permissions	2	-	-
5.2	Performance and Security	3	-	-
5.3	Firebase and AdMob	2	-	-
5.4	Publish	1	-	-
Total No. of Lecture Hours		**40**	-	-
Total No. of Tutorial Hours			**00**	-
Total No. of Practical Hours				**00**

Textbook:

1. Google Developer Training, "Android Developer Fundamentals Course – Concept Reference”, Google Developer Training Team, 2017.

Reference Book:

1. Erik Hellman, “Android Programming – Pushing the Limits”, 1st Edition, Wiley IndiaPvt Ltd, 2014.

2. Dawn Griffiths and David Griffiths, “Head First Android Development”, 1st Edition,O’Reilly SPD Publishers, 2015.

3. J F DiMarzio, “Beginning Android Programming with Android Studio”, 4th Edition,Wiley India Pvt Ltd, 2016.

4. Anubhav Pradhan, Anil V Deshpande, “Composing Mobile Apps” using Android, Wiley 2014.

Online Resources:

1. https://www.gitbook.com/book/google-developer-training/android-developerfundamentals-course-concepts/details

Code: BCS654D	**Course: Introduction to AI**
Credits: 3	**L:T:P: 3:0: 0**
SEE: 50	**CIE: 50**
SEE Hours: 3	**Max. Marks: 100**

Prerequisites if any	Probability , Statistics and Linear Algebra
Learning objectives	To gain insights on different concepts and methods used in Artificial intelligence to solve real world problems.

Course Outcomes:

On the successful completion of the course, the student will be able to

COs	**Course Outcomes**	**Bloom's level**
CO1	Explain Artificial Intelligence concepts and methods.	Understand
CO2	Use knowledge representation to solve real world problems	Apply
CO3	Use neural networks to solve real world problems	Apply
CO4	Solve problems using classification and clustering techniques.	Apply

Mapping with POs and PSOs:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12		PSO1	PSO2
CO1	2	2	-	-	-	-	-	-	-	-	-	3		-	-
CO2	2	2	3	-	3	-	-	-	-	-	2	3		3	2
CO3	2	3	3	-	3	-	-	-	-	-	2	3		2	3
CO4	3	3	3	-	3	-	-	-	-	-	2	3		2	3

Mapping Strength: Strong– 3 Medium – 2 Low – 1

Course Structure

Sl. No.	Module Name	No. of Lecture Hours	No. of Tutorial Hours	No. of Practical Hours
Module – 1: Introduction				
1.1	Introduction	1		
1.2	Characteristics	1		
1.3	Exhaustive Searches	1		
1.4	Heuristic Search Techniques	1		
1.5	Interative Deepening	1		
1.6	Constant satisfaction	1		
1.7	General problem solving	1		
Module – 2: Problem reduction and Logic concepts				
2.1	Bounded look ahead strategy	1		
2.2	Alpha-Beta Pruning	1		
2.3	Propositional calculus	1		
2.4	Propositional logic	1		
2.5	Natural Deduction system	1		
2.6	Axiomatic system	1		
2.7	Semantic tableau system in propositional logic	1		
2.8	resolution refutation in propositional logic and Predicate logic	2		
Module – 3: Advanced problem-solving paradigm				
3.1	Planning- types of planning systems	2		
3.2	Linear planning using a goal stack	1		
3.3	Non –linear planning strategies	1		
3.4	Means-ends analysis	1		
3.5	Knowledge representation using semantic network	1		
3.6	Extended semantic networks for KR	1		
3.7	Knowledge representation using frames	1		

Module – 4: Uncertainty Measure				
4.1	Probability Theory	2		
4.2	Bayesian Belief Networks	3		
4.3	Machine Learning Paradigms	3		
Module – 5: Support vector Machine, case-based reasoning and learning ANN				
5.1	Single Layer and Multilayer	1		
5.2	RBF	1		
5.3	Design issues in ANN	1		
5.4	Recurrent Network	1		
5.5	Deductive learning,	2		
5.6	Clustering	2		
Total No. of Lecture Hours		**40**		
Total No. of Tutorial Hours			**00**	
Total No. of Practical Hours				**00**

Textbook:

1. Artificial Intelligence, Saroj Kaushik Cengage Learning 2014 Edition

Reference Book:

1. Artificial Intelligence: Structures and Strategies for Complex Problem Solving, George F Luger Pearson Addison Wesley 6th Ed, 2008.
2. Artificial Intelligence, E Rich, K Knight, and S B Nair Tata Mc-Graw Hill 3rd Ed, 2009.
3. Artificial Intelligence: A Modern Approach, Stuart Russell and Peter Norvig Prentice Hall 3rd, 2009

Code: BCS654E	**Course: Introduction to Data Science**
Credits:3	**L:T:P-3:0:0**
SEE:50 Marks	**CIE:50 Marks**
SEE Hours:3	**Max.Marks:50**

Prerequisites if any	Linear Algebra, Probability and Statistics
Learning objectives	1. To understand the significance of Data Science in Industry and Academia. 2. Learn basics of R language

Course Outcomes:

On the successful completion of the course, the student twill be able to

COs	**Course Outcomes**	**Bloom's level**
CO1	Describe the importance of Data Science and learn Statistical modeling, probability distributions, fitting a model, Overfitting	Understanding
CO2	Demonstrate EDA and illustrate few machine learning algorithms and implement using R language.	Apply
CO3	Apply Bayesian law and Use Machine learning algorithms as spam filters	Apply
CO4	Explore Feature Generation, Feature Selection, recommended system and Data engineering.	Apply

Mapping with POs and PSOs:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12		PSO1	PSO2
CO1	3	2	2	-	-	-	-	-	2	-	-	3		-	-
CO2	2	2	2	-	2	-	-	-	2	-	2	3		2	2
CO3	2	3	2	-	2	-	-	-	2	-	2	3		2	2
C04	3	2	-	-	3	2	-	-	2	-	2	3		2	2

Mapping Strength: Strong–3 Medium–2 Low –1

Course Structure

Sl. No	Modules	No. of Lecture Hours	No. of Tutorial Hours	No. of Practical Hours
Module –1: Introduction				
1.1	**Introduction**: What is Data Science?	1	-	-
1.2	Big Data and Data Science hype –and getting past the hype,	1	-	-
1.3	Why now?–Data fiction, Current land scape of perspectives, A Data Science Profile, Thought Experiment: Meta-Definition, OK, So What Is a Data Scientist, Really?	1	-	-
1.4	Needed Statistical Inference: Statistical Thinking in the Age of Big Data, Statistical Inference, Populations and samples, Populations and Samples of Big Data, Big Data Can Mean Big Assumptions	1	-	-
1.5	Modeling: What is a model? Statistical modeling, Probability distributions, Fitting a model, Overfitting	2	-	-
1.6	R Programs for the algorithms	1	-	-
Module–2:Exploratory Data Analysis and Data Science Process				
2.1	**Exploratory Data Analysis and the Data Science Process:** Basic tools(plots, graphs and summary statistics) of EDA	1	-	-
2.2	Philosophy of EDA	1	-	-
2.3	The Data Science Process, A Data Scientist's Role in This Process	1	-	-
2.4	Algorithms: Machine Learning Algorithms, Three Basic Algorithms: Linear Regression	2	-	-
2.5	k-Nearest Neighbors(kNN)	1	-	-
2.6	k-means Clustering	1	-	-
2.7	Comparison of these three algorithms	1	-	-
Module–3:MachineLearning Algorithm and Usage in Applications				
3.1	**Machine Learning Algorithm and Usage in Applications:** Spam Filter, Linear Regression and Spam Filter, Filtering Spam,	1	-	-
3.2	K-NN and spam Filter	2	-	-
3.3	Naïve Bayes Algorithm, Spam Filter using Naïve Bayes , Laplace Smoothing, Comparing Naïve Bayes to K-NN Motivating application.	3	-	-
3.4	Data Wrangling: APIs and other tools for scrapping the Web	2	-	-
3.5	introduction to Logical Regression and M6D case study	1		
	Module -4 : Extracting Meaning from Data			
4.1	**Extracting Meaning from Data:**T heKaggle Model	1	-	-
4.2	Example: User(customer) retention.	1	-	-
4.3	Feature Generation(brain storming, role of domain expertise, and place for imagination)	1	-	-
4.4	Feature Selection algorithms: Filters; Wrappers	1	-	-
4.5	Decision Trees;	1	-	-
4.6	Random Forests	1	-	-

	Module -5: Recommended system and Data Engineering			
5.1	Recommendation Systems: A Real-World Recommendation Engine, Some Problems with Nearest Neighbors	1		
5.2	Beyond Nearest Neighbor: Machine Learning Classification	1		
5.3	The Dimensionality Problem	1		
5.4	Singular Value Decomposition.	1		
5.5	Data Engineering, Map reduce, ,	1		
5.6	Word Frequency Problem	1		
5.7	Map Reduce Solution			
5.8	Other Examples of Map Reduce, Pregel-An Introduction.	1		
5.9	Data Visualization: Basic principles.	1		
5.10	Ideas and tools for data visualization.	1		
Total No. of Lecture Hours		**40**		
Total No. of Tutorial Hours			**00**	
Total No. of Practical Hours				**00**

Text book:

1. Doing Data Science, Cathy O’Neil and Rachel Schutt, Straight Talk from The Frontline O‘Reilly, 2014

Reference books:

1. Data Mining: Concepts and Techniques Jiawei Han, Micheline Kamber and Jian Pei Morgan Kauffman, Third Edition, 2012
2. Mining of Massive DatasetsV2.1 Jure Leskovek, AnandRajaraman and Jeffrey Ullman, Cambridge University Press, 2nd Edition, 2014

Online Resources:

1. https://onlinecourses.nptel.ac.in/noc21_cs69
2. https://www.coursera.org/programs/projects/getting-started-with-kaggle?
3. https://www.coursera.org/learn/r-programming?

Code:BCS654F **Course: Introduction to Blockchain Technology**

Credits: 3 **L:T:P 3:0:0**

SEE: 50 Marks **CIE: 50 Marks**

SEE Hours: 3 **Max. Marks: 100**

Prerequisites if any	Computer networks, Cryptography
Learning objectives	● Impart strong technical understanding of Blockchain technologies. ● Develop familiarity of current technologies ● Introduce application areas and current practices

Course Outcomes:

On the successful completion of the course, the student will be able to

Cos	**Course Outcomes**	**Bloom's level**
CO1	Describe the operational aspects of the Blockchain ecosystem	Understand
CO2	Identify the cryptographic primitives behind Blockchain	Apply
CO3	Compare the consensus algorithm used in Blockchain technology	Understand
CO4	Discuss the functional aspects of Bitcoin network, Ethereum and SMART Contract	Understand

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO 1	3	2	-	2	2	1	-	-	-	-	-	-	2	2
CO 2	3	2	-	2	2	1	-	-	-	-	-	-	2	2
CO 3	3	2	-	2	2	-	-	-	-	-	-	-	2	2
CO 4	3	2	-	2	3	-	-	-	-	-	-	-	2	2

Mapping Strength: Strong–3 Medium–2 Low –1

Course Structures

Sl. No.	Module Name	No. of Lecture Hours	No. of Tutorial Hours	No. of Practical Hours
Module – 1: Introduction to Blockchain				
1.1	History of Bitcoin and Blockchain	2	-	-
1.2	Digital Ledger Technology (DLT), Peer-to-Peer (P2P) Network,	1	-	-
1.3	Centralized, Decentralized and Distributed Networks	2		
1.4	Public Blockchain, Private Blockchain.	1		
1.5	Applications of Blockchain	2		
	Module – 2: Decentralized System & Hash Functions			
2.1	Cryptographic Hash Functions	1		
2.2	Cryptographic Nonce	1		
2.3	Transactions, Asymmetric Key Cryptography	1		
2.4	Address and Address Derivation – Private Key Storage, Ledgers	1		
2.5	Blocks, Chaining Blocks	1		
2.6	Zero Knowledge System	1		
2.7	Attacks – 51% attack, Sybill attack.	1		
2.6	Different types of SHA , SHA-256	2		
	Module-3: Consensus algorithm			
3.1	Proof of Work Consensus Algorithm	1		
3.2	Proof of Stake Consensus Algorithm, Delegated Proof of Stake (DPoS)	2		
3.3	Proof of Burn, Practical Byzantine Fault tolerance	2		
3.4	Proof of elapsed time	1		
	Module – 4: Blockchain Mining and Forking			
4.1	Permission Blockchain, Permissionless Blockchain,	1		
4.2	Forking – Soft forking, Hard Forking	2		
4.3	Cryptographic changes forking, Merkle Tree	1		
4.4	Bitcoin Mining, Mining Incentives Strategies.	2		
4.5	Bitcoin Cryptocurrencies - Double Spending problem and its avoidance in Blockchain	1		

	Module-5: Blockchain platforms			
5.1	Ethereum Platform	1		
5.2	Transactions in Ethereum – Ether wallet, Ether Accounts	1		
5.3	Ether Gas, Gas Price, Gas Limit, Ether Tokens	1		
5.4	ERC20 Ethereum stands for Tokens	1		
5.5	Mining in Ethereum and Awards	1		
5.6	Smart Contract	1		
5.7	Hyperledger Platform –Hyperledger Fabric Architecture, Membership services, Blockchain services	2		
	Total hours	**40**	**0**	**0**

TEXT B0OK

1. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller and Steven Goldfeder, "Bitcoin and Crypto currency Technologies: A Comprehensive Introduction", Princeton University Press, Edition illustrated, 2016

2. Imran Bashir, "Mastering Blockchain: Distributed ledgers, Decentralization and Smart Contracts explained". PACKT PUBLICATION, 2nd edition, 2018

REFERENCE BOOKS

1. Malcolm Campbell-Verduyn, "Bitcoin and Beyond Crypto currencies, Blockchains,
And Global Governance", publisher Routledge; 1st edition 2017.
2. Kumar Saurabh, Ashutosh Saxena ,Blockchain Technology: Concepts and Applications Kindle Edition, Wiley , 2020
3. Daniel Lincoln, "Blockchain Evolution Explained: A Beginners Guide to Understanding Blockchain Technology", Kindle Edition.
4. Andreas M. Antonopoulos , David A. Harding, " Mastering Bitcoin: programming the open Blockchain", O'Reilly publication, third edition.
5. Andreas M. Antonopoulos , Gavin Wood , "Mastering Ethereum: Building Smart Contracts and DApps" , O'Reilly publication, 1st Edition,

Code: BCS605 **Course: Cloud Computing**

Credits: 4 **L:T:P-3:2:0**

SEE: 50Marks **CIE: 50 Marks**

SEEHours:3 **Max. Marks: 100**

Prerequisites if any	
Learning objectives	1. Overview of Cloud Computing and various distributed system models with enabling technologies. 2. Analyze various Computer Clusters for Scalable Parallel Computing. 3. Acquire the clear understanding of Virtual Machines and Virtualization of Clusters. 4. Acquire the basic knowledge of Cloud Platform Architecture over Virtualized Data Centers and acquire the clear understanding of Service-Oriented Architectures for Distributed Computing.

Course Outcomes:

On the successful completion of the course, the student will be able to

COs	**Course Outcomes**	**Bloom'slevel**
CO1	Explain the overview of Cloud Computing and various distributed system models with enabling technologies	Understand
CO2	Illustrate Virtual Machines and Virtualization of Clusters	Apply
CO3	Analyze various Computer Clusters for Scalable Parallel Computing.	Analyze
CO4	Acquire the basic knowledge of Cloud Platform Architecture over Virtualized Data Centers and acquire the clear understanding of Service-Oriented Architectures for Distributed Computing.	Analyze

Mapping with POs and PSOs:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PS
CO1	3	3	-	-	-	2	-	-	-	-	-	-	-	
CO2	3	2	2	2	2	-	-	2	-	2	2	-	-	
CO3	3	-	2	-	-	2	-	-	-	-	-	-	2	
CO4	3	3	-	2	-	2	-	-	-	-	1	-	2	

Mapping Strength:Strong– 3 Medium–2 Low–1

Course Structure

Sl. No	Modules	No. of Lecture Hours	No. of Tutorial Hours	No. of Practical Hours
	Module–1			
1.1	Defining Cloud Computing, Cloud Types: NIST Model, Cloud Cube Model, Deployment models, Service Models.	2	-	-
1.2	Characteristics of Cloud Computing	1	-	-
1.3	Benefits of Cloud computing, Disadvantages of cloud computing.	2	2	-
1.4	Scalable computing over the Internet,.	1	-	-
1.5	System Models for Distributed and Cloud Computing,	1	-	-
1.6	Software Environments for Distributed Systems and Clouds	1	-	-
	Module–2			
2.1	Computer Clusters for Scalable Parallel Computing: Clusteringfor Massive Parallelism	2	-	-
2.2	Computer Clusters and MPP Architectures	2	2	-
2.3	Design Principles of Computer Clusters	2	-	-
2.4	Cluster Job and Resource Management	2	-	-
	Module–3			
3.1	Implementation Levels of Virtualization	2	-	-
3.2	Virtualization Structures/Tools and Mechanisms	2	-	-
3.3	Virtualization of CPU, Memory, and I/O Devices	2	2	-
3.4	Virtual Clusters and Resource Management	2	-	-
	Module–4			
4.1	Cloud Computing and Service Models,	2	-	-
4.2	Data-Center Design and Interconnection Networks,	2	-	-
4.3	Architectural Design of Compute and Storage Clouds,	2	-	-
4.4	Public Cloud Platforms: GAE,AWS	2	2	-

Module–5				
5.1	Services and Service-Oriented Architecture,	2	-	-
5.2	Message-Oriented Middleware,	2	-	-
5.3	Discovery, Registries, Metadata, and Databases,	2	2	-
5.4	Workflow in Service-Oriented Architectures.	2	-	-
	Total No. of Lecture Hours	**40**	-	-
		Total No. of Tutorial Hours	**10**	**-**
			Total No. of Practical Hours	**00**

Textbook:

1. Cloud Computing Bible by Barrie Sosinsky, Wiley India (Chapter1).
2. Distributed And Cloud Computing, Hwang, Kai; Fox, GeoffreyC; Dongarra,JackJ. ELSEVIER INDIA PVT.LTD (Chapter 1, 2, 3, 4, 5, 6) 2013, 1st Edition.

Reference Book:

1. Cloud Computing for Dummies by Judith Hurwitz, R.Bloor, M.Kanfman,F.Halper(WileyIndiaEdition).
2. Cloud Security by Ronald Krutzand Russell Dean Vines, Wiley-India.
3. Cloud Computing, APracticalApproach,AnthonyTVelte.
4. Google Apps by Scott Granneman, Pearson
5. A Brief Guide to Cloud Computing,An Essential Introduction to the Next Revolution in Computing, Christopher Barnatt.

Code: BCS606	**Course: Distributed Systems**
Credits: 3	**L:T:P 3:0:0**
SEE: 50 Marks	**CIE: 50 Marks**
SEE Hours: 3	**Max. Marks: 100**

Prerequisites if any	Nil
Learning objectives	1. Learn the fundamentals of distributed systems, through examples 2. Learn to use appropriate remote invocation techniques for communication in DS

Course Outcomes: *On the successful completion of the course, the student will be able to*

COs	**Course Outcomes**	**Bloom's level**
CO1	Explain the applications and challenges arising from the construction of distributed systems	Understand ing
CO2	Make use of appropriate remote invocation technique to communicate in distributed systems	Apply
CO3	Use distributed objects and components to develop applications using CORBA middleware	Apply
CO4	Examine algorithms for clock synchronization and use appropriate concurrency control algorithm for transactions	Analyze

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	-	-	-	-	-	-	-	-	-	3	1	-
CO2	3	3	-	-	-	-	-	-	-	-	-	3	-	2
CO3	3	2	2	-	2	-	-	-	-	-	-	3	2	-
CO4	3	2	3	2	2	-	-	-	-	-	-	3	2	2

Mapping with POs and PSOs:

Mapping Strength: Strong– 3 Medium – 2 Low – 1

Course Structure

Sl. No	Modules	No. of Lecture Hours	No. of Tutorial Hours	No. of Practical Hours
Module – 1: Characterization of Distributed Systems, System models				
1.1	Introduction to DS, examples	1	0	0
1.2	Trends in distributed system. Focus on resource sharing	2	0	0
1.3	Challenges	1	0	0
1.4	Physical models, Architectural Models, Fundamental Models	3	0	0
1.5	Case study: WWW	1	0	0
Module – 2: Inter Process Communication, Remote Invocation				
2.1	Introduction, API for Internet Protocols	1	0	0
2.2	External data representation and Marshalling	2	0	0
2.3	Client – Server Communication	1	0	0
2.4	RR Protocol, RPC, RMI	3	0	0
2.5	Case study: SUN RPC	1	0	0
Module – 3: OS Support, Distributed Objects, DFS				
3.1	Processes, Threads, OS Architecture	2	0	0
3.2	Introduction to Distributed Objects and components	1	0	0
3.3	Case study: CORBA – architecture, service	2	0	0
3.4	Distributed file systems, File service architecture	2	0	0
3.5	Case study: Sun Network File System	1	0	0
Module – 4: Time and Global State				
4.1	Clocks, Events and process status	2	0	0
4.2	Synchronizing physical clocks	2	0	0
4.3	Logical time and logical clocks, Global states	3	0	0
4.4	Distributed debugging	1	0	0
Module – 5: Transactions and Concurrency Control				
5.1	Introduction to Transactions, Nested Transactions	3	0	0
5.2	Locks, Optimistic Concurrency Control, Timestamp ordering	4	0	0
5.3	Comparison of methods for concurrency control	1	0	0
Total No. of Lecture Hours		**40**	**-**	**-**
Total No. of Tutorial Hours			**00**	**-**
Total No. of Practical Hours				**00**

Textbook:

1. George Coulouris, Jean Dollimore, Tim Kindberg, Gordon Blair: Distributed Systems – Concepts and Design, Fifth Edition, Pearson Publications, 2012.

Reference Book:

1. Maarten van Steen. Andrew S. Tanenbaum: Distributed Systems, Third edition, 2017

.**Online Resources:**

1. www.cdk5.net/corba
2. https://www.coursera.org/specializations/pcdp
3. https://onlinecourses.nptel.ac.in/noc21_cs87/

Code: BCSL607	**Course: Machine Learning Lab**
Credits: 1	**L:T:P- 0:0:2**
SEE: 50Marks	**CIE: 50 Marks**
SEE Hours:2	**Max. Marks:100**

Prerequisites if any	Basic knowledge of Python programming, data structures, linear algebra, probability, and statistics
Learning objectives	**1.** Develop an understanding of core Python libraries and apply them to implement and analyze fundamental machine learning algorithms. 2. Gain hands-on experience in designing, testing, and interpreting supervised, unsupervised, and reinforcement learning models using real-world datasets.

Course Outcomes:

On the successful completion of the course, the student will be able to

COs	Course Outcomes	Bloom's level
CO1	Learn Python libraries for data handling and visualization in ML experiments.	Understand
CO2	Implement Machine Learning algorithms like KNN, Regression, Decision Trees, Naïve Bayes, and K-means on datasets.	Apply
CO3	Examine Machine Learning algorithms performance using accuracy and visualization to interpret results.	Analyze

Mapping with Cos and PSOs:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	1	1	2	-	-	-	-	1	-	1	3	2
CO2	3	3	2	2	3	-	-	-	-	2	1	-	3	3
CO3	3	2	3	2	2	1	-	-	1	2	1	1	3	3

Mapping Strength:Strong– 3 Medium–2 Low –1

SL.NO	**Programs**
1	Understand the Basic Python Libraries such as math. NumPy, Pandas, Matplotlib
2	For a given set of training data examples stored in a .CSV file, implement and demonstrate the Find-S algorithm to output a description of the set of all hypotheses consistent with the training examples. **Book 1: Chapter 3**
3	Develop a program to implement k-Nearest Neighbour algorithm to classify the randomly generated 100 values of x in the range of [0,1]. Perform the following based on dataset generated. a. Label the first 50 points {xl,……,x50} as follows: if (xi ≤ 0.5), then xi ∈ Classl, else xi ∈ Classl **a.** Classify the remaining points, x51,……,x100 using KNN. Perform this for k=1,2,3,4,5,20,30 **Book 2: Chapter – 2**
4	Develop a program to demonstrate the working of Linear Regression **Book 1: Chapter – 5**
5	Develop a program to demonstrate the working of Polynomial Regression. **Book 1: Chapter – 5**
6	Develop a program to demonstrate the working of the decision tree algorithm. Use Breast Cancer Data set for building the decision tree and apply this knowledge to classify a new sample. **Book 2: Chapter – 3**
7	Develop a program to implement k-means clustering using Wisconsin Breast Cancer data set and visualize the clustering result. **Book 2: Chapter – 4**
8	WriteaprogramtoimplementthenaïveBayesianclassifierforasampletraining dataset storedasa.CSV file. Compute the accuracy of the classifier, considering few test datasets. **Book 1: Chapter – 8**
9	Develop a program to Implement the Q-Learning Algorithm **Book 1: Chapter – 14**

Course Structure

Suggested Learning Resources:

Books:

1. S Sridhar and M Vijayalakshmi, “Machine Learning”, Oxford University Press, 2021.

2. M N Murty and Ananthanarayana V S, “Machine Learning: Theory and Practice”, Universities Press (India)

Pvt. Limited, 2024.

Web links and Video Lectures (e-Resources):

1. https://www.drssridhar.com/?page_id=1053

2. https://www.universitiespress.com/resources?id=9789393330697

3. https://onlinecourses.nptel.ac.in/noc23_cs18/preview

Code: BCSL657A	**Course: Progressive App Development**
Credits: 1	**L:T:P - 0:0:2**
SEE: NA	**CIE: 50 Marks**
SEE Hours: NA	**Max. Marks: 50**

Prerequisites if any	Basic Web Development Knowledge, Understanding of Client-Server Architecture and Basic programming skills
Learning objectives	To bridge the gap between fundamental web development skills and advanced PWA development, ensuring students are well-equipped to create modern, performant, and user-friendly web applications.

Course Outcomes:

On the successful completion of the course, the student will be able to

COs	**Course Outcomes**	**Bloom's level**
CO1	Develop and deploy Progressive Web Apps (PWAs) with offline capabilities and responsive design.	Understanding
CO2	Implement modular applications using microservices architecture and integrate third-party RESTful APIs	Apply
CO3	Apply cloud services for deployment and manage continuous integration/continuous deployment (CI/CD) pipelines.	Apply
CO4	Optimize the performance and security of web applications.	Analyze

Mapping with POs and PSOs:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	-	3	-	3	-	-	-	-	-	-	-	2	-
CO2	3	2	3	-	3	-	-	-	-	-	-	-	-	2
CO3	3	2	3	-	3	-	-	-	2	-	-	3	2	-
CO4	3	3	3	-	3	-	-	-	-	-	-	-	-	2

Mapping Strength: Strong– 3 Medium – 2 Low – 1

Course Structure

Sl.No	Modules	No. of Lecture Hours	No. of Tutorial Hours	No. of Practical Hours
1.	Implement a service worker to cache assets and enable offline functionality in your PWA.	-	-	1
2.	Use CSS techniques and frameworks (e.g., Flexbox, Grid, Bootstrap) to ensure your PWA is fully responsive across different devices.	-	-	1
3.	Implement the Push API and Service Workers to enable push notifications in your PWA.	-	-	1
4.	Create, containerize using Docker, and deploy a simple microservice.	-	-	1
5.	Fetch data from the API and display it in your application using JavaScript or a framework like React.	-	-	1
6.	Apply user-centric design principles to improve the usability and accessibility of your application.	-	-	2
7.	Deploy your application to a cloud service (e.g., AWS, Azure, Google Cloud) and ensure it scales effectively.	-	-	2
8.	Configure a CI/CD pipeline using tools like Jenkins, GitHub Actions, or GitLab CI for automated testing and deployment.	-	-	2
9.	Implement techniques to improve load time and overall performance, such as code splitting, lazy loading, and caching strategies	-	-	2
10.	Apply security best practices to mitigate common vulnerabilities, such as input validation, using HTTPS, and securing APIs.	-	-	2
	Total No. of Lecture Hours	-	-	-
		Total No. of Tutorial Hours	**00**	**-**
			Total No. of Practical Hours	**15**

Textbook:

1. Jason Grigsby, Progressive Web Apps", A Book Apart, 2018
2. Tal Ater, Building Progressive Web Apps, O'Reilly Media, Inc., 2017

Online Resources:

1. MDN Web Docs: Progressive Web Apps: https://developer.mozilla.org/en-US/docs/Web/Progressive_web_apps
2. Google Developers: Progressive Web Apps: https://developers.google.com/codelabs/pwa-training/pwa05--empowering-your-pwa#0

Code: BCSL657B	**Course: Tosca – Automated Software Testing**
Credits: 1	**L:T:P - 0:0:2**
SEE: NA	**CIE: 50 Marks**
SEE Hours: NA	**Max. Marks: 50**

Prerequisites if any	Any Programming language, Discrete Mathematical structures
Learning objectives	• Gain a deep understanding of Tosca's core concepts and functionalities • Learn efficient automation of test cases using Tosca. • Develop expertise in creating effective and reusable test designs. • Acquire skills for building robust automated test scripts

Course Outcomes:

On the successful completion of the course, the student will be able to

COs	**Course Outcomes**	**Bloom's level**
CO1	Demonstrate the creation and management of modules with a strong grasp of identification methods	Apply
CO2	Explain the fundamentals of Tricentis Tosca and its advantages over other automation tools	Understand
CO3	Examine the testing efficiency and quality through effective Tosca automation	Apply

Mapping with POs and PSOs:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12		PSO1	PSO2
CO1	2	-	2	-	3	-	-	-	-	-	-	-		-	-
CO2	2	-	2	-	2	-	-	-	-	-	-	-		-	-
CO3	3	-	3	-	3	-	-	-	-	-	-	-		3	3

Mapping Strength: Strong– 3 Medium – 2 Low – 1

Course Structure

Sl. No	Modules	No. of Lecture Hours	No. of Tutorial Hours	No. of Practical Hours
Module – 1: Automate Web Applications, Desktop Application Testing&API Testing				
1.1	Practice creating automated tests for simple HTML applications using Tosca's capabilities1.	-	-	2
1.2	Develop test cases for desktop applications and understand the integration with Tosca.	-	-	2
1.3	Learn to automate API testing and validate responses using Tosca	-	-	1
Module – 2: Mobile Testing, Data-Driven Testing&Risk-Based Testing				
2.1	Execute automated tests on mobile applications and explore Tosca's mobile testing features.	-	-	2
2.2	Implement data-driven tests to validate application behavior under various data sets.	-	-	2
2.3	Apply risk-based testing methods to prioritize test execution based on business impact.	-	-	2
Module – 3: Continuous Integration&Test Data Management				
3.1	Integrate Tosca with CI/CD pipelines for continuous testing practices	-	-	2
3.2	Manage and maintain test data efficiently using Tosca's test data management tools.	-	-	2
	Total No. of Lecture Hours	-	-	-
		Total No. of Tutorial Hours	-	-
			Total No. of Practical Hours	**15**

Textbook:

1. Tricentis Tosca Fundamentals: A comprehensive guide to automating web application testing with Tosca4.
2. Introduction to Tosca Automation Testing: Detailed course curriculum for Tosca Automation Testing6

Reference Book:

1. Tosca Training Curriculum by Croma Campus: Industry-oriented training material aligned with certification exams

Code: BCS657C	**Course:** Agile Development
Credits: 1	**L:T:P - 1:0:0**
SEE: NA	**CIE: 50 Marks**
SEE Hours: NA	**Max. Marks: 50**

Prerequisites if any	Nil
Learning objectives	1. To Recognize the importance to be Agile 2. To understand basics of agile 3. To become Expert in Agility

Course Outcomes:

On the successful completion of the course, the student will be able to

COs	**Course Outcomes**	**Bloom's level**
CO1	Explain the concept of Agility in Agile processes.	Understand
CO2	Illustrate the principles of Agile Process, its evolution and adoptability in the development.	Apply
CO3	Demonstrate the concepts of agile management, classification of agile methods and agility in design, testing and documentation effectively	Apply

Mapping with POs and PSOs:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	1	1	-	3	-	-	-	-	2	-	3	1	1
CO2	3	2	2	2	3	-	-	-	2	2	-	3	2	2
CO3	3	2	2	-	3	-	-	-	2	2	-	3	2	2

Mapping Strength: Strong– 3 Medium – 2 Low – 1

Course Structure

Sl.No	Modules	No. of Lecture Hours	No. of Tutorial Hours	No. of Practical Hours
Module – 1: Agile				
1.1	**Agile? :** Understanding Success, Beyond Deadlines, The Importance of Organizational Success, Enter Agility	1	-	-
1.2	How to Be Agile?: Agile Methods, Don't Make Your Own Method, The Road to Mastery, Find a Mentor.	1	-	-
1.3	The Genesis of Agile, Introduction and background, Agile Manifesto, and Principles, Simple Design, User Stories, Agile Testing, Agile Tools	2	-	-
Module – 2: Mastering Agility				
2.1	**Mastering Agility** :Values and Principles: Commonalities, About Values, Principles, and Practices, Further Reading	1	-	-
2.2	Improve the Process: Understand Your Project, Tune and Adapt, Break the Rules	1	-	-
2.3	Rely on People :Build Effective Relationships, Let the Right People Do the Right Things, Build the Process for the People	1	-	-
2.4	Eliminate Waste :Work in Small, Reversible Steps, Fail Fast, Maximize Work Not Done, Pursue Throughput	2	-	-
Module – 3: Agile Methodology				
3.1	Theories for Agile management – agile software development – traditional model vs. agile model -	1	-	-
3.2	Classification of agile methods – agile manifesto and principles – agile project management	2	-	-
3.3	Agile team interactions – ethics in agile teams	1	-	-
3.4	Agility in design, testing – agile documentations – agile drivers, capabilities and values	2	-	-
	Total No. of Lecture Hours	**15**	-	-
	Total No. of Tutorial Hours		**00**	-
	Total No. of Practical Hours			**00**

Textbook:

1. James shore, Chromatic, O'Reilly, The Art of Agile Development, 2007

Reference Book:

1. Ken Schawber, Mike Beedle, "Agile Software Development with Scrum",Pearson, 2008

Online Resources: https://www.nptelvideos.com/video.php?id=904

Code: BCS657D	**Course: Devops**
Credits: 1	**L:T:P - 1:0:0**
SEE: NA	**CIE: 50 Marks**
SEE Hours: NA	**Max. Marks: 50**

Prerequisites if any	--NIL--
Learning objectives	Understanding of DevOps principles, tools, and practices. To effectively collaborate in a DevOps environment, automate processes, and manage infrastructure as code.

Course Outcomes:

On the successful completion of the course, the student will be able to

COs	**Course Outcomes**	**Bloom's level**
CO1	explain the core concepts, principles, and benefits of DevOps.	Understanding
CO2	Demonstrate Proficiency in using version control systems and implementing collaborative development workflows.	Apply
CO3	Demonstrate Skill in provisioning and managing infrastructure using Infrastructure as Code (IaC) tools.	Apply

Mapping with POs and PSOs:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	-	2	-	2	-			-	-	-	1	-	-
CO2	-	-	3	-	3	-			-	-	1	-	2	3
CO3	-	3	3	-	3	-			-	-	-	1	3	3

Mapping Strength: Strong– 3 Medium – 2 Low – 1

Course Structure

Sl. No	Modules	No. of Lecture Hours	No. of Tutorial Hours	No. of Practical Hours
Module – 1: Introduction to Devops and its Principles				
1.1	Introduction to DevOps and its Principles: Understanding DevOps: Definition, goals, and benefits	1	-	-
1.2	Key principles of DevOps: Continuous integration, continuous delivery, automation, and collaboration	1		
1.3	DevOps culture and mindset, Case studies of successful DevOps implementations	1		
1.4	Version Control and Collaboration Tools: Introduction to version control systems (e.g., Git)	1		
1.5	Branching and merging strategies, Collaborative development workflows, Introduction to code review processes and tools(example GITHUB, Bit bucket)	1		
Module – 2: Continuous Integration and Build Automation				
2.1	Continuous integration (CI) concepts and benefits	1	-	-
2.2	Introduction to build automation tools (e.g., Jenkins, Travis CI)	1	-	-
2.3	Configuring and managing CI pipelines, running automated tests and generating reports	1	-	-
2.4	Infrastructure as Code (IaC): Introduction to Infrastructure as Code (IaC) and its benefits	1	-	-
2.5	Infrastructure provisioning tools (e.g., Terraform, CloudFormation), Building and managing infrastructure using IaC	1	-	-
Module – 3: Configuration Management				
3.1	Introduction to configuration management tools (e.g., Ansible, Puppet)	1	-	-
3.2	Managing system configurations and deployments	1	-	-
3.3	Configuration drift and remediation, Automating software deployments	1	-	-

3.4	Continuous Delivery and Deployment: Introduction to continuous delivery and deployment concepts	1	-	-
3.5	Release management and versioning, Automating deployment pipelines	1	-	-
	Total No. of Lecture Hours	15	-	-
	Total No. of Tutorial Hours		**00**	**-**
	Total No. of Practical Hours			**00**

Textbook:

1. "Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation" by Jez Humble and David Farley.
2. "DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations" by Gene Kim, Jez Humble, Patrick Debois, and John Willis.

Reference Book:

1. "Site Reliability Engineering: How Google Runs Production Systems" edited by Betsy Beyer, Chris Jones, Jennifer Petoff, and Niall Richard Murphy.
2. "Accelerate: The Science of Lean Software and DevOps: Building and Scaling High Performing Technology Organizations" by Nicole Forsgren, Jez Humble, and Gene Kim.

Online Resources:

1. NPTEL Course Link: https://nptel.ac.in/courses/128106012

** *** **