LOK JAGRUTI UNIVERSITY (LJU)

INSTITUTE OF ENGINEERING & TECHNOLOGY

Department of Computer Engineering (701)

Bachelor of Technology (B.E.) – Semester - IV

Course Code:	**01713492**
Course Name:	**Full Stack Development with Javascript-2**
Category of Course:	Engineering Science Course (ESC)
Prerequisite Course:	Full Stack Development with Javascript-1(017012191)

Teaching Scheme				
Lecture (L)	**Tutorial (T)**	**Practical (P)**	**Credit**	**Total Hours**
3	**0**	**6**	**6**	**30**

Syllabus

Unit No.	Topic	Prerequisite Topic	Successive Topic	Teaching Hours
01	**JSON(Javascript Object Notation)**			**2 (5%)**
	1.1 JSON Introduction: JSON vs XML, JSON syntax	---	---	
	1.2 JSON Datatypes: Number, Boolean, string, JSON.parse(), JSON.stringify(), Objects, Arrays, Date	DOM(017012191-Unit-9.1)	---	
02	**Node JS-1**			**8 (15%)**
	2.1 Introduction, Setup, REPL, Callbacks:Demonstrating callback using simple JS functions like setInterval(), setTimeout(), NPM Global Module:NodeMon,	Fat arrow functions(017012191-Unit-10.3)	---	
	2.2 Core modules :File system, OS Modules, Path Modules	Fat arrow functions(017012191-Unit-10.3)	---	
	2.3 HTTP Module, Render Response, Read HTML File Server, Routing, JSON Response	Fat arrow functions(017012191-Unit-10.3)	---	
03	**Node JS-2**			**6 (10%)**
	3.1 URL, Querystring, How to create, export and use our own modules	Fat arrow functions(017012191-Unit-10.3)	---	
	3.2 NPMjs (Chalk, Validator), Module Wrapper Function, JSON Processing, Events	---	---	
04	**Express JS-1**			**8 (10%)**
	4.1 Introduction, Features, Environment Setup	---	---	
	4.2 Routing, JSON response passing, HTTP Methods- GET, POST, Middleware,	---	---	
	4.3 How to Link HTML, CSS & JS file in Express JS, Cookies, Session	---	---	
05	**Express JS-2**			**8 (10%)**
	5.1 File Upload, RESTful APIs,	HTML Forms-(017012191-Unit-2.3)	---	
	5.2 Mail send-nodemailer, Express JS templates-View Engine (PUG)	Data formatting tags-(017012191-Unit-2.2)	---	
06	**React JS-1**			**8 (15%)**
	6.1 Introduction, Advantages, Setup, React Render Elements-folder structure	---	---	
	6.2 Node Package Manager (NPM), Package.json, Virtual DOM	---	---	
	6.3 React JSX-Comments, Nested Element, Attributes, Styling in JSX, Components, Class, Functional component, Arrow component, Events, Props	Fat arrow functions(017012191-Unit-10.3)	---	
	6.4 React Routing, Events, Lists, Keys, Map, Filter		---	
07	**React JS-2**			**8 (10%)**
	7.1 Hooks (useState, useEffect, useReducer, useContext)	---	---	
	7.2 React CSS	Syntax and Type of CSS(017012191-Unit-4.1)	---	
	7.3 Forms, API integration with AXIOS	HTML Forms-(017012191-Unit-2.3)	---	
08	**Git**			**2 (5%)**
	8.1 Introduction to Git, Version Control System	---	---	
	8.2 Git Repository, Adding to a Repository	---	---	
	8.3 Centralized vs Distributed Version Control System	---	---	
	8.4 Working with Git-Important Git Commands	---	---	
09	**MongoDB-1**			**6 (10%)**
	9.1 Introduction to NoSQL Database, Features of MongoDB	---	---	
	9.2 Difference between RDBMS and NoSQL, MongoDB vs MySQL, Installation	---	---	
	9.3 Cursor, Create & Drop Database	---	---	
10	**MongoDB-2**			**8 (10%)**
	10.1 Create, Drop Collection, CRUD Operation:Create, Read, Update, Delete, Find	---	---	
	10.2 Operators: Comparison, Logical, Arithmetic, Field Update	---	---	
	10.3 Projection, Limiting and Sorting Records	---	---	

	10.4 Indexing, Replication, RegEx	---	---	

Sr No.	Practical Title	Link to Theory Syllabus
1	Write a JS to create a Person object using JSON. Store name and age inside object. Print details only of elder person.	Unit-1
2	Write a JS to store an array of objects having height and name. Display names by sorting an array according to height.	Unit-1
3	Add two JSON objects having members like hours, minutes and seconds. After addition, if seconds goes beyond 60, then minutes should be incremented and if minutes go beyond 60, then hour should be incremented.	Unit-1
4	Write a node.js script to read a file and print its contents.	Unit-2
5	Write a node.js script to write contents to the file in original manner. Delete file after finishing writing.	Unit-2
6	Write a program to sort an integer array, where all elements are available in a file separated by white space.	Unit-2
7	Write a program to demonstrate various methods of path module in Node.js.	Unit-2
8	Write a node.js script to jump on a specific code by specifying path on address bar of browser.	Unit-2,3
9	Write a node.js script to print 3 different JSON objects by specifying corresponding object name on address bar.	Unit-2,3
10	Write a node.js script to load a simple.html file on NodeJS server and print its contents.	Unit-3
11	Perform addition of 2 numbers by specifying function definition and call in separate files and exporting them.	Unit-3
12	Write a script to declare and array, complement it and calculate its summation. Print complement of each element in green color and print summation in red color.	Unit-2
13	Write a Node.js script to create two listeners for a common event. Call their respective callbacks. Print no. of events associated with an emitter. Remove one of the listeners and call remaining listeners again. Also print no. of remaining listeners.	Unit-3
14	Perform routing by specifying one page name on address bar using express.js.	Unit-4
15	Write an express.js script to define one JSON array of 3 objects having members name & age. Sort these objects according to age. Means according to age, names must be sorted. If user requests “sortedName” URL, then all names should be printed according to ascending order of age.	Unit-4
16	Demonstrate create, read and delete operations on cookie.	Unit-4
17	Demonstrate session handling by creating a hit counter of a website.	Unit-4
18	Write a node.js script to display specific JSON object using RESTful API on express.js.	Unit-5
19	Write a react.js file to pass an integer value to a component, which is defined as a property in app.js file. Fetch that value in component and print its table on console. Write all necessary files.	Unit-6
20	Write an event handling script to convert all typed letters of text-field to uppercase onChange. Print converted string to suitable HTML tag.	Unit-6,7
21	Demonstrate CRUD operations on MongoDB.	Unit-9
22	Fetch all records of an object in sorted manner of record ID.	Unit-10

Major Components/ Equipment

Sr. No.	Component/Equipment
1	Computer
2	Software: code::blocks, Editor, Various packages to run Node.js, express.js and React.js

Proposed Theory + Practical Evaluation Scheme by Academicians
(% Weightage Category Wise and it’s Marks Distribution)

L:	3	T:	0	P:	6

Note: In Theory Group, Total 4 Test (T1+T2+T3+T4) will be conducted for each subject.
Each Test will be of 25 Marks.
Each Test Syllabus Weightage: Range should be 20% - 30%

Group (Theory or Practical)	Group (Theory or Practical) Credit	Total Subject Credit	Category	% Weightage	Marks Weightage
Theory	3	6	MCQ	20%	40
Theory			Theory Descriptive (Mainly Programming)	30%	60
Theory			Formulas and Derivation	0%	0
Theory			Numerical	0%	0

Expected Theory %	50%		Calculated Theory %	50%	100
Practical	3		Individual Project	15%	30
Practical			Group Project	10%	20
Practical			Internal Practical Evaluation (IPE)	25%	50
Practical			Viva	0%	0
Practical			Seminar	0%	0
Expected Practical %	**50%**		**Calculated Practical %**	**50%**	**100**
Overall %	**100%**			**100%**	**200**

Course Outcome	
1	Understand the concepts of back end scripting language node.js.
2	Explore the new features of node.js using its framework as express.js
3	Learn to create various hands on projects like ToDo list, digital clock using react.js
4	Integration with Mongo DB database
Suggested Reference Books	
1	The ultimate beginner's guide to learn Node.js strp by step, John Bach & Alexander Aronowitz, 3rd edition, mEm Inc
2	Pro express.js, Azat Mardan, Apress publication
3	Full stack react, The complete guide to react.js, Anthony Accomazzo et.al.
4	MongoDB, The definitive guide, Kristina Chodorow, Michael Dirolf, O'Reilly publication

List of Open Source Software/Learning website	
1	www.geeksforgeeks.org
2	www.tutorialspoint.com
3	www.programiz.com
4	www.codingninjas.com

Practical Project/Hands On Project		
Sr. No.	**List of Practical Projects**	**Linked with Unit**
1	Write a program to create a ToDo list to add and remove tasks.	6,7
2	Write a script to create a digital clock running continuously.	6,7
3	Write a script to meet following requirements: (1) Create index.html page and open it on localhost. (2) After clicking on submit button, it should jump on saveSession page. Store username in session. (3) After saving session, redirect to fetchSession page and read session value. Put a logout link button here. (4) Jump on deleteSession page by clicking Logout link. (5) Destroy the session on this page and redirect to index.html page.	4,5
4	Create a file upload script having size and extension validations using express.js.	4,5
5	Write a Node.js script to write the text "This is data" to new.txt file. After that append the text "That is data" to same new.txt file. After that read the file and print file contents on console. After finishing read operation, print the line "Thanks for using my program" on console. All read/write operations are asynchronous.	2,3