

FULL STACK DEVELOPMENT-2

Course Code	23SO8554	Year	III	Semester	I
Course Category	SOC	Branch	CSE	Course Type	Practical
Credits	2	L – T – P	0-1-2	Prerequisites	HTML,CSS, JAVASCRIPT
Continuous Evaluation:	30	Semester End Evaluation:	70	Total Marks:	100

Course Outcomes		
Upon successful completion of the course, the student will be able to:		
CO1	Build an application that handles routing, and authentication using express JS	L3
CO2	Develop a single page application using RESTful APIs in Express JS	L3
CO3	Apply React JS router and hooks to build an User interface.	L4
CO4	Use MongoDB to connect the application and perform CURD operations	L4

List of Experiments		
S.No.	Experiment	Mapped CO
1	Express JS – Routing, HTTP Methods, Middleware. a. Write a program to define a route, Handling Routes, Route Parameters, Query Parameters and URL building. b. Write a program to accept data, retrieve data and delete a specified resource using http methods. c. Write a program to show the working of middleware.	CO1, CO2
2	Express JS – Templating, Form Data a. Write a program using templating engine. b. Write a program to work with form data.	CO1, CO2
3	Express JS – Cookies, Sessions, Authentication a. Write a program for session management using cookies and sessions. b. Write a program for user authentication.	CO1, CO2
4	ReactJS – Render HTML, JSX, Components – function & Class a. Write a program to render HTML to a web page. b. Write a program for writing markup with JSX. c. Write a program for creating and nesting components (function and class).	CO2, CO3

5	ReactJS – Props and States, Styles, Respond to Events <ol style="list-style-type: none"> Write a program to work with props and states. Write a program to add styles (CSS & Sass Styling) and display data. Write a program for responding to events. 	CO2, CO3
6	ReactJS – Conditional Rendering, Rendering Lists, React Forms <ol style="list-style-type: none"> Write a program for conditional rendering. Write a program for rendering lists. Write a program for working with different form fields using react forms. 	CO2, CO3
7	ReactJS – React Router, Updating the Screen <ol style="list-style-type: none"> Write a program for routing to different pages using react router. Write a program for updating the screen. 	CO2, CO3
8	ReactJS – Hooks, Sharing data between Components <ol style="list-style-type: none"> Write a program to understand the importance of using hooks. Write a program for sharing data between components. 	CO2, CO3
9	MongoDB – Installation, Configuration, CRUD operations <ol style="list-style-type: none"> Install MongoDB and configure ATLAS Write MongoDB queries to perform CRUD operations on document using insert(), find(), update(), remove() 	CO1, CO4
10	MongoDB – Databases, Collections and Records <ol style="list-style-type: none"> Write MongoDB queries to Create and drop databases and collections. Write MongoDB queries to work with records using find(), limit(), sort(), createIndex(), aggregate(). 	CO1, CO4
11	Express JS – Database, RESTful APIs <ol style="list-style-type: none"> Write a program to connect MongoDB database using Mongoose and perform CRUD operations. Write a program to develop a single page application using RESTful APIs. 	CO1, CO4
12	<p><u>CAPSTONE PROJECT:</u></p> <p>Develop a project (student/faculty choice) using full stack technologies like react, node.js, express.js and mongoDB. Example (for reference): E-commerce application Front-End:</p> <ul style="list-style-type: none"> User Interface (UI): A visually appealing and user-friendly interface using HTML, CSS, and JavaScript frameworks like React. Functionality: Features like product listings, shopping cart, user accounts, search functionality, and order history. <p>Back-End (use node.js and express.js,mongodb)</p> <ul style="list-style-type: none"> Server-side Logic: Handling user authentication, product data management, shopping cart operations, and order processing. Database: Storing product information, user data, and order details using a database like PostgreSQL or MongoDB. APIs: Creating and consuming APIs for communication between the front-end and back-end. 	CO1, CO2, CO3, CO4

Learning Resources	
Text Books	
<ol style="list-style-type: none"> 1. Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node, Vasana Subramanian, 2nd edition, APress, O'Reilly. 2. Node.js in Action, Mike Cantelon, Mark Harter, T.J. Holowaychuk, Nathan Rajlich, Manning Publications. (Chapters 1-11) 3. React Quickly, Azat Mardan, Manning Publications (Chapters 1-8, 12-14) 	
Reference Books	
<ol style="list-style-type: none"> 1. Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node by Vasana Subramanian, Apress; 2nd ed. edition (13 May 2019), ISBN-10 : 1484243900, ISBN-13 : 978-1484243909 2. MongoDB, React, React Native Full-Stack Fundamentals and Beyond, by Eric Bush, Zaccheus Entertainment Publication, ISBN-10 : 0997196688, ISBN-13 : 978-0997196689. 3. Quick Start Guide: Build web applications with MongoDB, Express.js, React, and Node by Eddy Wilson Iriarte Koroliova, Packt Publishing, ISBN-10: 1787281086, ISBN-13: 978-1787281080 4. Beginning Node.js, Express & MongoDB Development, by Greg Lim (Author), ISBN-10 : 9811480281 5. React Quickly: Painless web apps with React, JSX, Redux, and GraphQL, by Azat Mardan, Published by Manning Publications, Manning Publications, ISBN-10 : 1617293342, ISBN-13 : 978-1617293344 	
E-Resources & other digital material	
<ol style="list-style-type: none"> 1. ExpressJS - https://www.tutorialspoint.com/expressjs 2. ReactJS - https://www.w3schools.com/REACT (and) https://react.dev/learn# 3. MongoDB - https://learn.mongodb.com/learning-paths/introduction-to-mongodb 	