1.

BOS	**Computer Science**
Class	**S.Y.B.Sc.C.S.**
Semester	**III**
Course Name	**Full stack Development Paper-I**
Course Code	**PUSCS305(F)**
Type of course	**Discipline Specific Elective**
Level of the Subject	**Medium**
Credit points	**2**

Course Objectives:

1. Provide an overview of React.js and its key features, including virtual DOM, component-based architecture, and declarative syntax.
2. Understanding JSX: Introduce JSX (JavaScript XML), the syntax extension used in React.js for defining the structure and layout of components.
3. State and Props: Explain the concepts of state and props in React.js and how they are used to manage and pass data between components.
4. React Router, a popular routing library for React.js, and demonstrates how to implement client-side routing in a React application.

Unit No.	Name of Unit	Topic No.	Content	No of Lectures
1	Getting Started with React.js	1.1	**Introduction:** Introduction to React and its benefits, Environment setup for React development, Refresher on ES6 concepts, Create React App, Folder Structure.	15
		1.2	**Templating using JSX:** Understanding component architecture and its significance, Introduction to components and their types,Working with React.createElement to create elements,Using expressions, logical operators, attributes, and children in JSX.	
		1.3	**Working with Props and State** :	

			Understanding the concept of state and its significance in React, Setting and reading component states, Working with props to pass data between components, Validating props using propTypes, Using default props to supply default	

			values, Rendering lists using the React key prop and the map function.	
2	Understanding Component Lifecycle:	2.1	Stateful Function Components with Hooks: Using hooks, specifically useState, useEffect, useContext, and useReducer,Creating custom hooks,	15
		2.2	Overview of the lifecycle of a React component,Handling side effects and managing cleanup,Working with events and error management,	
		2.3	Exploring the React event	
3	Working with Forms	3.1	Managing controlled and uncontrolled components. Utilizing the defaultValue prop. Accessing the DOM element using the React ref prop. Building a currency converter project.	15
		3.2	Understanding useContext and useReducer hooks, Creating custom hooks.	
		3.3	Routing with React Router: Setting up React Router for navigation in single-page applications,Configuring routes with BrowserRouter and HashRouter. Making routes dynamic using route params,Working with nested routes and navigation using Link and NavLink components.	
4	State Management and Redux:	4.1	Introduction to Redux and its principles. Installing and setting up Redux. Creating actions, reducers, and the store.	15

		4.2	Setting up Redux for state management. Implementing actions and reducers for the catalog and cart.	
		4.3	Using the connect() higher-order function to connect components. Utilizing Redux Hooks for state management. Implementing middleware and persistence in Redux.	
			Total No. of Lectures	**60**

Course Outcomes:

1. Gain a solid understanding of React.js and its core concepts, including virtual DOM, component-based architecture, and JSX syntax.
2. Develop the ability to create reusable and modular components in React.js, allowing for efficient and scalable front-end development.
3. Effectively manage state and props in React.js components, enabling dynamic rendering and interaction within applications.
4. How to bind event handlers, handle form submissions, and update component state based on user interactions.
5. React Router, a popular routing library for React.js, and demonstrates how to implement client-side routing in a React application.
6. To apply the concepts learned throughout the course and build real-world React.js applications.

Reference Books:

1. Functional Web Development with React and Redux by Alex Banks and Eve Porcello:
2. React Up and Running: Building Web Applications by Stoyan Stefanov:
3. Pro React by Cassio de Sousa Antonio:
4. React Cookbook: Create Dynamic Web Apps with React using Redux, Webpack, Node.js, and GraphQL by Carlos Santana Roldan:
5. Fullstack React: The Complete Guide to ReactJS and Friends by Anthony Accomazzo, Nate Murray, and Ari Lerner:

Case study 1 :

Travelduck is the biggest Polish marketplace that connects individual adventure trip organizers with their customers. They offer a wide range of activities, from trekking, biking and sailing in Poland to exploring Chernobyl or diving in Sri Lanka.

Challenge:

Travelduck's goal was to consolidate the market of boutique trips offered by individual providers on a very fragmented market. The end goal was to create an easy-to-use marketplace where tour organizers can showcase and sell their adventure trips.

1. Create different Components as per requirements.
2. Create a React FORM for taking adventure trips details.
3. Create routes for different types activates

Case study 2 :

SnowShow is the most prominent Polish winter tour provider, and also one of the biggest in Europe. They target Alps in terms of destination and students and grads as their audience. Their uniqueness lies in the combination of sport (skiing and snowboarding) with music. SnowShow organizes over 60 trips yearly. The biggest one, called "Music Fest", attracts over 2000 adventurous people and the biggest music stars.

Challenge:
We started our journey with SnowShow in the fall of 2015. Their system already existed and was developed by a freelancer. SnowSnow needed a more robust solution with a trustworthy and scalable team. The booking system we took over was complex and included multiple implicit business logic. Making it simple, readable and self-descriptive was the main business and technical challenge for us at the beginning of the project.
Create different Components as per requirements.
Create a React FORM for taking Books details
Create routes for different types of Tour

BOS	**Computer Science**
Class	**S.Y.B.Sc.C.S.**
Semester	**III**
Course Name	**Front End development with React.js Practical**
Course Code	**PUSCS305(F)**
Type of course	**Discipline Specific Elective**
Level of the Subject	**Medium**
Credit points	**2**

Sr. No.	Practicals name
1.	Create a new React project using Create React App.
2.	Review ES6 features like arrow functions, template literals, destructuring, spread syntax,
3.	**Templating using JSX:** 1. Create a simple React component using JSX. 2. Use expressions, logical operators, attributes, and children in JSX.
4.	**Understanding Components and their types:** 1. Explore the concept of components in React. 2. Understand the difference between functional components and class components
5.	**Working with Props and State:** 1. Create a component that accepts and uses props. 2. Set and read component states using useState. 3. Pass data between components using props.
6.	**Stateful Function Components with Hooks:** 1. Create stateful function components using useState. 2. Utilise useEffect for handling side effects and cleanup.
7.	**Lifecycle Methods and Event Handling:** 1. Understand the lifecycle of a React component. 2. Handle events using the React event system.
8.	**Accessing the DOM Element using the React ref prop:** Use the ref prop to access and manipulate DOM elements in React.
9.	**Routing with React Router:** 1. Install and set up React Router for navigation in single-page applications. 2. Create dynamic routes using route parameters.

10.	**Installing and Setting up Redux:** 1. Install the Redux library and set up the Redux store. 2. Add middleware to handle asynchronous actions.