React Native Course Contents(5 days)
By Dr. Vishwanath Rao

Prerequisites:

- Basic understanding of JavaScript and ES6 syntax.
- Familiarity with React concepts (components, props, state, etc.).
- Understanding of mobile app development concepts (UI/UX, navigation, etc.).

Objectives:

- Gain proficiency in building cross-platform mobile apps using React Native.
- Understand the React Native app lifecycle and component lifecycle.
- Learn to use React Hooks for state management and side effects.
- Develop responsive app designs for multiple screen sizes.
- Implement global variables and objects for shared state management.
- Master UI components like FlatList for efficient rendering of long lists.
- Understand bridging concepts for integrating native iOS components.
- Implement state management with Redux and middleware like saga and thunk.
- Learn filesystem operations for file persistence and data storage.
- Implement API calls using Axios for fetching data from remote servers.
- Understand databases like SQLite and Realm for local data storage.
- Design common functional components for reuse across the app.
- Implement localization for multi-language support in the app.
- Debug React Native apps effectively.
- Build adaptive user interfaces for various screen sizes.
- Implement navigation between screens in React Native apps.
- Handle user input and form submissions.
- Send HTTP requests to remote servers for data retrieval and updates.
- Implement user authentication and authorization features.
- Build React Native apps without using Expo.
- Implement push notifications for real-time communication.

Course Contents

1. Introduction to React Native

- Overview of React Native framework
- Setting up the development environment
- Creating your first React Native app
- Understanding React Native project structure

2. App Lifecycle and Component Lifecycle

- Introduction to React Native app lifecycle

- Component lifecycle methods (componentDidMount, componentWillUnmount, etc.)
- Handling app state changes and updates

3. React Hooks

- Understanding useState hook for managing component state
- Using useEffect hook for handling side effects and data fetching
- Exploring other hooks like useContext, useReducer, useCallback, etc.

4. Responsive App Design

- Designing UI layouts using Flexbox
- Implementing responsive designs for different screen sizes
- Using Platform-specific styles and components

5. Global Variables and Objects

- Implementing global state management with Context API
- Sharing state between components using context
- Managing app-wide settings and configurations

6. UI Components: FlatList

- Introduction to FlatList component
- Optimizing performance for rendering long lists
- Implementing pagination and item caching

7. Bridging Concepts

- Integrating native iOS components into React Native apps
- Making two-way communication between native and React Native code
- Handling platform-specific implementations and features

8. Redux

- Introduction to Redux for state management
- Implementing Redux store, actions, and reducers
- Using middleware like saga and thunk for async actions

9. Filesystem Operations

- Accessing filesystem for file storage and persistence
- Saving and retrieving files in device directories
- Handling file operations asynchronously

10. API Calling

- Making HTTP requests using Axios library
- Handling API responses and error handling
- Implementing RESTful API calls in React Native apps

11. Databases

- Introduction to SQLite and Realm databases
- Setting up and configuring local databases
- Performing CRUD operations on database records

12. Common Component Design

- Designing reusable functional components
- Implementing HOCs (Higher Order Components) for code reuse
- Best practices for component design and organization

13. Localization

- Implementing multi-language support in React Native apps
- Using i18n libraries for localization
- Handling text translations and language switching in the app

14. Debugging React Native Apps

- Using React Native Debugger for inspecting and debugging
- Utilizing console.log statements and React Native error messages
- Debugging with Chrome Developer Tools or React Native Debugger

15. Building Adaptive User Interfaces

- Implementing responsive designs with Flexbox
- Utilizing media queries and breakpoints for adaptive layouts
- Designing platform-specific UIs for iOS and Android devices

16. Navigation

- Implementing stack navigation with React Navigation
- Navigating between screens and passing params
- Configuring tab, drawer, and modal navigation

17. Handling User Input

- Capturing user input with TextInput and Touchable components

- Implementing form validation and error handling
- Handling gestures and touch events for user interactions

18. Sending HTTP Requests

- Making asynchronous HTTP requests with Axios
- Handling RESTful API calls and responses
- Implementing error handling and data caching

19. User Authentication

- Implementing user authentication flows with Firebase or custom backend
- Managing user sessions and authentication tokens
- Securing routes and restricting access to authenticated users

20. Building React Native Apps without Expo

- Setting up React Native CLI for project initialization
- Managing native dependencies and linking libraries
- Building and running React Native apps on iOS and Android devices

21. Push Notification

- Implementing push notifications with Firebase Cloud Messaging (FCM)
- Handling push notification permissions and payloads
- Sending push notifications from server to devices