MERN Stack Developer Training Curriculum

STRUCTURE

MERN (MongoDB, Express, ReactJS, NodeJS) Training Curriculum

Learn Building Amazing Full-Stack Apps with the Best in MongoDB, Express, React and Node.js

About Croma Campus:

Croma Campus Training & Development Private Limited is an education platform since 2010 providing rigorous industry-relevant programs designed and delivered in collaboration with world-class faculty and industry.

- Hands-On Live Projects
- Simulation Test Papers
- Industry Cases Studies
- 61,640+ Satisfied Learners
- 140+ Training Courses
- 100% Certification Passing Rate
- Live Instructor Classroom / Online Training
- 100% Placement Assistance

Course Objectives:

- Learn core computer science concepts from leading industry experts with content structured to ensure industry relevance
- Build an end-to-end application with exciting features and test it
- Earn an industry-recognized course completion certificate
- Make you career-ready as a full stack MERN developer upon successful completion of the program.
- Brush up on your knowledge of software development fundamentals; Agile and Scrum methodologies.
- Learn HTML, CSS, JavaScript to create dynamic websites; and GIT to manage version control systems.

Croma Campus Training Program Deliverables:

- **Session Recordings** - Original Class Room Voice & Video Recording
- **Training Material** - Soft Copy Handbooks
- **Assignments** | Multiple Hands-on Exercises
- **Test Papers** - We provide **Practice Test** as part of our course to help you prepare for the actual certification exam.
- **Live Case Studies**
- **Live Projects** - Hands-on exercises and Project work. You will work on real time industry-oriented projects and assignments for each module to practice.
- **Key focus on Hands-on exercises and Project work**. You will work on real time industry-oriented projects.
- Faculty with more than **10+ Years of Experience** in the Industry.
- **Technical Resume Designing & Job Assistance:** With more than 100+ Clients across the Globe and we help learners to get a good job in their respective field. We also help learners with resume preparation.
- **Interview Q&A**

- **About Croma Campus Training Certificate:** Croma Campus will provide you with an industry-recognized (Certified by **ISO 9001:2015** & **E-Cell IIT Jodhpur**) course completion certificate which has lifelong validity.
- **How I Unlock my Croma Campus Certificate:** Attend Complete Batch & Submit at least One Completed Project.

Course Description:

This program will advance your career as a MERN stack developer. You'll learn top skills such as MongoDB, Express.js, ReactJS, and Node.js ("MERN"), plus GIT, HTML, CSS, and JavaScript to build and deploy interactive applications and services.

Our MERN Full-Stack Developer Course at Croma Campus is designed as per the Industrial trends with 100% Job Assistance. We have a team of highly experienced professionals' faculties with more than 12 years of experience in MERN and other related Technologies.

Our faculty members are fully aware of industry requirements and ready to deliver full-fledged and industrial based Training to students across India. We provide our students with a perfect platform to learn and explore the subject from corporate trainers & industry experts.

This MERN stack developer program offers a comprehensive curriculum using our unique Blended Learning approach to make you career-ready as a full stack MERN developer upon successful completion of the program. So, what are you waiting for?

Make yourself as one of most skilled and knowledgeable resource today!

A Quick Glance at MERN Program:

(MongoDB | Express.JS | React.JS | Node.JS)

What is MERN?

The MERN stack is a JavaScript stack that's designed to make the development process smoother. MERN includes four open-source components: MongoDB, Express.JS, React.JS, and Node.JS. These components provide an end-to-end framework for developers to work in.

Become a MERN Stack JavaScript Developer:

React and Node are a popular combination for building robust full-stack applications. Learn full-stack JavaScript development with MongoDB, Express.JS, React.JS, and Node.JS (MERN). In this comprehensive learning path, you can learn how to build data-driven applications, as well as test, secure, and deploy your code.

- Build web apps using React.JS and Node.JS
- Work with data using MongoDB and Express.JS
- Test, Secure, and Deploy your apps.

Training Curriculum

- JavaScript (ECMAScript (ES6, ES7, ES8, ES9))
- MongoDB
- Express.JS
- React.JS
- Node.JS

MongoDB, a NoSQL database

Express.JS, a web application framework that runs on Node.JS

React.JS, React is an open-source JavaScript library for building user interfaces. It is maintained by Facebook and a community of individual developers and companies.

Node.JS, is an open-source, cross-platform, JavaScript runtime environment that executes JavaScript code outside a web browser.

Course Content:

JavaScript with ECMAScript

Module 1: Introduction to JavaScript Framework

- History of JavaScript
- What is ES6 (ECMAScript 6/JavaScript 6)
- ES6 Module System
- A Word on Bable
- Block Scope, Let & Const
- Template Literals
- Arrow Functions
- Spread and Rest Operators
- Object Literal Improvements
- De-structuring
- Classes
- Inheritance
- Static Properties and Methods
- Promises
- Iterators and Iterables
- Generators
- Modules
- New Features in ES6
- JavaScript let
- JavaScript const
- JavaScript Arrow Functions
- JavaScript Classes
- Default parameter values
- Array.find()
- Array.findIndex()
- Exponentiation (**) (ECMA Script 2016)

Node.JS: (Node.JS Coding Standards & Best Practices)

Module 1: Introduction

- Audience
- Pre-requisites
- About Node
- Execute Node
- Features
- Who use Node
- Concepts
- Where to use
- Where not to use

Module 2: Environment Setup

- Text Editor
- NodeJs Run Time
- Download NodeJs
- Installation
- Executing

Module 3: First Application

- Creating a NodeJs Application
- Make a request to NodeJs Server

Module 4: REPL Terminal

- What is REPL?
- Starting REPL
- REPL Commands
- Stopping REPL

Module 5: Packager Manager (NPM)

- Installing Modules using NPM
- Global vs Local Installation
- Using packages.json
- Attributes of packages.json
- Uninstalling Module
- Updating Module
- Searching Module
- Create a Module

Module 6: Call backs Concept

- Using Props

- Default Props
- State and Props

Module 7: Event Loop

- Validating Props

Module 8: Event Emitter

- Set State
- Force Update
- Find DOM Node

Module 9: Buffers

- Lifecycle Methods

Module 10: Streams

- Simple
- Complex

Module 11: File System

- Simple
- Child

Module 12: Global Objects

- What is Refs
- Using Refs

Module 13: Utility Modules

- What is Keys
- Using Keys

Module 14: Web Modules

- Install a React Router
- Add a Router
- Create Components

Module 15: Express Framework

- What is Flux
- Flux Elements
- Flux Props

Module 16: Restful API

- Install REDUX
- Create Files and Folders
- Actions
- Reducers
- Store
- Root Component
- Other Components

Module 17: Scaling Application

- Install React CSS Transitions Group
- Add a CSS File
- Appear Animation
- Enter and Leave Animations

Module 18: MongoDB

- Connecting Node and MongoDB
- Database Creation, Drop
- Collection Operations
- Documents Operations

Module 19: Application

- Node and MongoDB Application

Express JS – Web Application Framework

Module 1: Express.JS Basics

- Introduction, Advantages
- Environment Setup
- Basic Application

Module 2: Advanced Concepts of Express.JS

- Request
- Response
- GET
- POST
- Routing
- HTTP Methods
- URL Building
- Middleware
- Templates
- Static Files
- Form Data
- Database
- Cookies
- Sessions
- File Upload
- Authentication
- REST FUL API’s
- Scaffolding
- Debugging

MongoDB – No SQL Database

Module 1: MongoDB Basics

- Introduction, Advantages
- History, Features
- No SQL Databases
- Advantages over RDBMS

Module 2: Environment Setup

- Install MongoDB
- MongoDB Shell
- MongoDB Data Model
- MongoDB Datatypes

Module 3: Database

- Create Database
- Drop Database

Module 4: Collection

- Create Collection
- Drop Collection

Module 5: CRUD Documents

- Insert Documents
- Update Documents
- Delete Documents
- Query Documents

Module 6: Methods

- Limit ()
- Sort ()
- Skip ()

React.JS Course Content:

(An Open-Source JavaScript library for Building User Interfaces)

Module 1: Introduction of React.JS

- Audience
- Pre-requisites
- About React
- Features
- Advantages
- Limitations

Module 2: Environment Setup

- Create of Root Folder
- Install Global Packages
- Add Dependencies and Plugins
- Create the Files
- Set Compiler, Server and Loaders
- html
- JSX and Main.js
- Running the Server

Module 3: JSX

- What is JSX?
- Using JSX
- Nested Elements
- Attributes
- JavaScript Expressions
- Styling
- Components
- Naming Convention

Module 4: Components

- Stateless
- State full

Module 5: State

- What is State?
- Props

Module 6: Props Overview

- Using Props
- Default Props
- State and Props

Module 7: Props Validation

- Validating Props

Module 8: Component API

- Set State
- Force Update
- Find DOM Node

Module 9: Component Life Cycle

- Lifecycle Methods

Module 10: Forms

- Simple
- Complex

Module 11: Events

- Simple
- Child

Module 12: Refs

- What is Refs?
- Using Refs

Module 13: Keys

- What are Keys?
- Using Keys

Module 14: Router

- Install a React Router
- Add a Router
- Create Components

Module 15: Flux Concept

- What is Flux?
- Flux Elements
- Flux Props

Module 16: Using Flux and REDUX

- Install REDUX
- Creating Components
- Working with States
- Composing Components
- Actions

- Event Handlers
- Reducers
- Store
- Root Component
- Other Components
- Unit Testing – Tools, React, REDUX

Module 17: Animations

- Install React CSS Transitions Group
- Add a CSS File
- Appear Animation
- Enter and Leave Animations

Module 18: Higher-Order Components

- What is Higher-Order Component?