LEARN TODAY AND LEAD TOMORROW

Overview

Let eMexo Technologies Best **MERN Stack Training in Electronic City Bangalore** take you from the Basic MERN Stack to Advance MERN Stack Training and make you an expert in developing real-time applications. Here are the major topics we cover under this MERN Stack course Syllabus **MongoDB, Express, React JS, and Node JS**. Each and every topic will be covered in the most practical way with examples for our MERN Stack Course in Electronic City Bangalore.

All the topics will be covered with Practical and hands-on training. Our trainers have industry experience with live project experience in cutting-edge technologies that they teach. We hire only MERN Stack industry specialists as trainers for our **MERN Stack Certification Training in Electronic City Bangalore**.

If you are looking for MERN Stack Certification Course in Electronic City Bangalore, eMexo Technologies is the Best **MERN Stack Training Institute in Electronic City Bangalore**. Come over to our training institute for a free demo class. Let our trainer give you a demo on MERN Stack and only then you take the decision to enroll in the MERN Stack Training Course in Electronic City Bangalore.

Training Features

Real-life Case Studies

Do a real-life case study to understand the usage in real-world scenarios.

Assignments

Each class will be followed by a practical assignment switch that can be completed before the next class.

Preparation for interview

Our trainers are professionals working in multinational corporations. They are experts in their field and they know exactly what the interviewer will look for in the candidate. Experienced trainers not only share interview questions but also conduct mock interviews to help prepare for the actual interview.

Key Features

eMexo Technologies offers the **Best MERN Stack Training Course in Electronic City Bangalore** with the TOP industry expert trainers.

Here are the key features.

- ★ Free Demo Class Available
- ★ Practical Approach
- ★ Expert & Certified Trainers

- ★ 100% Job Oriented Training
- ★ Real World use cases and Scenarios
- ★ Completed 500+ Batches
- ★ Certification Guidance

MongoDB

Unit 1: MongoDB Basics & Concepts

➢ INTRODUCTION TO MONGODB - ARCHITECTURE AND INSTALLATION

- Understanding the basic concepts of a Database
- Database categories: What is NoSQL? Why NoSQL? The benefit over RDBMS
- Types of NoSQL Database, and NoSQL vs. SQL Comparison, ACID & Base Property
- CAP Theorem, implementing NoSQL and what is MongoDB?
- Overview of MongoDB, Design Goals for MongoDB Server and Database, MongoDB tools
- Understanding the following: Collection, Documents, and Key/ Values, etc.,
- Introduction to JSON and BSON documents
- Case study discussion
- Environment setup (live Hands-on) and using various MongoDB tools available in the MongoDB Package

➢ SCHEMA DESIGN AND DATA MODELLING

- Data Modelling Concepts
- Why Data Modelling? Data Modelling Approach
- Analogy between RDBMS & MongoDB Data Model, MongoDB Data Model (Embedding & Linking)
- Challenges for Data Modelling in MongoDB

- Data Model Examples and Patterns
- Model Relationships between Documents
- Model Tree Structures
- Model Specific Application Contexts
- Use Case discussion of Data modeling
- Hands-on

➢ CRUD OPERATIONS

- MongoDB Development Architecture
- MongoDB Production Architecture
- MongoDB CRUD Introduction, MongoDB CRUD Concepts
- MongoDB CRUD Concerns (Read & Write Operations)
- Concern Levels, Journaling, etc.,
- Cursor Query Optimizations
- Query Behavior in MongoDB
- Distributed Read & Write Queries
- MongoDB Datatypes
- MongoDB CRUD Syntax & Queries
- Hands-on

➢ INDEXING AND AGGREGATION FRAMEWORK

- Index Introduction, Index Concepts, Index Types, Index Properties
- Index Creation and Indexing Reference

- Introduction to Aggregation
- Approach to Aggregation
- Types of Aggregation (Pipeline, MapReduce & Single Purpose)
- Performance Tuning
- Hands-on

➢ MongoDB ADMINISTRATION

- Administration concepts in MongoDB
- Monitoring issues related to Database
- Monitoring at Server, Database, Collection level, and various Monitoring tools related to MongoDB
- Database Profiling, Locks, Memory Usage, No of connections, page fault etc.,
- Backup and Recovery Methods for MongoDB
- Export and Import of Data to and from MongoDB
- Run time configuration of MongoDB
- Production notes/ best practices
- Data Managements in MongoDB (Capped Collections/ Expired data from TTL), Hands-on Administrative Tasks
- Hands-on

➢ SCALABILITY AND AVAILABILITY

- Introduction to Replication (High Availability)
- Concepts around Replication
- What is Replica Set and Master-Slave Replication?
- Type of Replication in MongoDB

- How to set up a replicated cluster & manage replica sets etc.,
- Introduction to Sharding (Horizontal Scaling)
- Concepts around Sharding, what shards, Key
- Config Server, Query Router, etc.
- How to set up a Sharding
- Type of Sharding (Hash-Based, Range, Based, etc.), and Managing Shards
- Hands-on

➢ MongoDB SECURITY

- Security Introduction
- Security Concepts
- Integration of MongoDB with Jaspersoft
- Integration of MongoDB with Pentaho
- Integration of MongoDB with Hadoop/Hive
- Integration of MongoDB with Java
- Integration of MongoDB with GUI Tool Robomongo
- Case Study MongoDB and Java
- Hands-on

➢ APPLICATION ENGINEERING AND MONGODB TOOLS

- MongoDB Package Components
- Configuration File Options
- MongoDB Limits and Thresholds

- Connection String URI Format/ Integration of any compatible tool with MongoDB API and Drivers for MongoDB
- MMS (MongoDB Monitoring Service)
- HTTP and Rest Interface
- Integration of MongoDB with Hadoop and Data Migration MongoDB with Hadoop (MongoDB to Hive)
- Integration with R
- Hands-on

➢ MongoDB ON THE CLOUD

- Overview of MongoDB Cloud products
- Using Cloud Manager to monitor MongoDB deployments
- Introduction to MongoDB Stitch
- MongoDB Cloud Atlas
- MongoDB Cloud Manager
- Working with MongoDB Ops Manager
- Hands-on

➢ DIAGNOSTICS AND FIXES

- Overview of tools
- MongoDB Diagnostic Tools
- Diagnostics Commands
- MongoDB Deployment
- Setup & Configuration, Scalability, Management & Security
- Slow Queries

- Connectivity
- Hands-on

Express JS

Unit 2: Express JS Basics & Concepts

➢ INTRODUCTION TO EXPRESS JS

- The model-view-controller pattern
- Defining Jade templates
- Configuring Express
- Postman configuration
- Using REST
- JSON Data
- Reading POST data
- CRUD operations
- Adding middleware
- Routing
- Error Handling
- Debugging Express
- Hands-on

➢ EXPRESS JS WITH MONGODB AND SQLITE

- How Node.js connects to databases
- RDBMS databases and NoSQL databases

- Connecting to RDBMS and Mongodb databases
- Performing CRUD operations
- Hands-on

➢ SOCKET.IO, THE FRONT-END, AND A CHAT APP

- Getting Started
- Adding Socket.io To Your App
- Exploring The Front-end
- Sending Live Data Back & Forth
- Creating The Front-end UI
- Showing Messages In App
- Working With Time
- Timestamps
- Show Message Time In Chat App
- Chat application Project
- Hands-on

➢ INTRODUCTION TO TASK MANAGERS WITH UNIT TESTING

- Working with gulp
- Working with grunt
- Working with unit and E2E testing
- Hands-on

➢ Express Advanced topics

- ○ Developing template engines for Express
- ○ Process managers for Express apps
- ○ Security updates
- ○ Production Best Practices Security
- ○ Production best practices performance and reliability
- ○ Health Checks and Graceful Shutdown
- ○ Hands-on

React JS

Unit 3: React JS Basics & Concepts

➢ INTRODUCTION TO WEB DEVELOPMENT AND REACT

- ○ Building Blocks of Web Application Development
- ○ Single-page and Multi-page Applications
- ○ Different Client-side Technologies
- ○ MVC Architecture
- ○ Introduction to React
- ○ Installation of React
- ○ JSX and its use case
- ○ DOM
- ○ Virtual DOM and its working
- ○ ECMAScript

- Difference between ES5 and ES6
- NPM Modules
- Hands-On

➢ COMPONENTS AND STYLING THE APPLICATION LAYOUT

- React Elements
- Render Function
- Components
- Class Component
- Component Constructor
- Functional Components
- Multiple Components
- Props
- Props with Class-based Component
- Props with Function based Component
- States
- Component Lifecycle
- React Events
- React Forms
- Different Form Concepts
- Styling in React
- Inline Styling

- CSS Stylesheet
- Building Music Shop Application using React Components
- Hands-On

➢ HANDLING NAVIGATION WITH ROUTES

- Routing
- react-router
- Features of react-router
- Configuration of routing using react-router
- Navigation using Links
- 404 page (Not found Page)
- URL Parameters
- Nested Routes
- Implementing styles using NavLink
- Application Programming Interface
- Build a REST API using JSON-server
- API consumption in React application using Fetch method
- Build a dynamic Music Store application using Routing and API connectivity
- Hands-On

➢ REACT STATE MANAGEMENT USING REDUX

- Need of Redux
- What is Redux?

- Redux Architecture
- Redux Action
- Redux Reducers
- Redux Store
- Principles of Redux
- Pros of Redux
- NPM Packages required to work with Redux
- More about react-redux package
- Hands-On

➢ ASYNCHRONOUS PROGRAMMING WITH SAGA MIDDLEWARE

- Need of Async operations
- Async Workflow
- Action Creators
- How to write Action Creators?
- Handling Async Actions via Reducers
- Middleware
- Redux-Saga
- Generators in Redux-Saga
- Saga Methods()
- Major Sections of Redux-Saga
- Building a Product List application using Redux-Saga Middleware

- ○ Debugging application using Redux Devtools
- ○ Hands-On

➢ REACT HOOKS

- ○ Caveat of JavaScript classes.
- ○ Functional components and React hooks
- ○ What are React hooks?
- ○ Basic hooks
- ○ useState() hook
- ○ How to write useState() hook when the state variable is an array of objects
- ○ useEffect() hook
- ○ Fetch API data using useEffect() hook
- ○ useContext() hook
- ○ Rules to write React hooks
- ○ Additional hooks
- ○ Custom hooks
- ○ Hands-On

➢ FETCH DATA USING GRAPHQL

- ○ What is GraphQL?
- ○ Cons of Rest API
- ○ Pros of GraphQL
- ○ Frontend backend communication using GraphQL

- Type system
- GraphQL datatypes
- Modifiers
- Schemas
- GraphiQL tool
- Express framework
- NPM libraries to build the server side of GraphQL
- Build a GraphQL API
- Apollo client
- NPM libraries to build client side of GraphQL
- How to setup Apollo client
- Hands-On

➢ **REACT APPLICATION TESTING AND DEPLOYMENT**

- Define Jest
- Setup Testing environment
- Add Snapshot testing
- Integrate Test Reducers
- Create Test Components
- Push Application on Git
- Deploy App on Nginx

- Create Docker for React Application
- Hands-On

➢ INTRODUCTION TO REACT NATIVE

- Native Applications
- React Native
- React Native Elements
- Expo CLI
- Build a shopping cart mobile application using React Native
- React Native installation and setup
- Working with Styles and Layout
- Hands-On

➢ BUILDING REACT NATIVE APPLICATION WITH API

- Native modules
- Native Navigation libraries
- Integration of Redux with React Native
- React Native and Redux major components
- Redux Thunk middleware
- NPM libraries
- Shopping cart application using React Native and Redux
- Integration of Redux actions, store, and reducers In React Native application
- Hands-On

Node JS

Unit 4: Node JS Basics & Concepts

➢ Introduction to Node.js

- What is Node.js?
- Why Node.js?
- Installing NodeJS
- Node in-built packages (buffer, fs, http, os, path, util, url)
- Node.js Modules
- Import your own Package
- Node Package Manager (NPM)
- Local and Global Packages
- Push code to GitHub
- Hands-On

➢ File System Module and Express.js

- Get Input from Users
- Pass Multiple Arguments with Yargs
- File System Module
- Operations associated with File System Module
- JSON Data
- HTTP Server and Client
- Sending and receiving events with EventEmitters

- Express Framework
- Run a Web Server using Express Framework
- Routes
- Deploy the application using PM2 and Nginx
- Hands-On

➢ Asynchronous Programming

- Call Stack
- Callbacks, Callback Queue, and Event Loop
- Callback Abstraction
- Callback Chaining
- Promises
- Promise Chaining
- Request Package
- Customizing HTTP Requests
- Error handing with appropriate HTTP codes
- Introduction to template engine (EJS)
- Hands-On

➢ Integration with MongoDB and Email Server

- Introduction to NoSQL Databases and MongoDB
- Installation of MongoDB on Windows
- Installation of Database GUI Viewer

- Inserting Documents
- Querying, Updating, and Deleting Documents
- Connect MongoDB and Node.js Application
- Exploring SendGrid
- Sending emails through Node.js application using SendGrid
- Hands-On

➢ REST APIs and GraphQL

- REST API
- REST API in Express
- Postman
- MongoDB Driver API
- Express Router
- Mongoose API
- GraphQL
- GraphQL Playground
- Hands-On

➢ Building Node.js Applications using ES6

- ES6 variables
- Functions with ES6
- Import and Export withES6
- Async/Await

- Introduction to Babel
- Rest API with ES6
- Browsing HTTP Requests with Fetch
- Processing Query String
- Creating API using ES6
- Transpilation
- Building Dashboard API
- Creating dashboard UI with EJS
- ES6 Aside: Default Function Parameters
- Data Validation and Sanitization
- Hands-On

➢ User Authentication and Application Security

- Authentication
- Types of Authentication
- Session Vs Tokens
- JSON Web Tokens
- Bcrypt
- Node-localstorage
- Hands-On

➢ Dynamic Client-Server Interaction using Socket.IO

- Web Sockets

- Socket.io
- Broadcasting Events
- Sharing Your Location
- Event Acknowledgements
- Form and Button States
- Rendering Messages
- Working with Time and Timestamps for determining Location of Messages
- Storing Users, Rendering User List, Tracking Users Joining and Leaving
- Deploying the Chat Application
- Redis – Building API with Redis
- Hands-On

➢ Testing Node.js Applications

- Writing Tests and Assertions
- Testing Asynchronous Code
- Testing an Express Application
- Setup and Teardown
- Testing with Authentication
- Advanced Assertions
- Mocking Libraries
- Wrapping up User Tests
- Setup Task Test Suite

- Testing with Task Data
- Hands-On

➢ **Microservices Application**

- Why Microservices?
- What are Microservices?
- Why Docker?
- What is Docker?
- Terminologies in Docker
- Child Processes
- Types of child process
- Hands-On

FAQs

1. How is the training organized? How much percentage is theoretical and how much is practical hands-on?

We at eMexo believe nothing beats hands-on practice when it comes to learning a concept. Our teaching methodology is 100% practical and hands-on-oriented. You learn a concept, you practice it then and there with the trainer. We also give you assignments for each topic which you can practice at home and any doubts regarding the topic can be cleared with the trainer the next day.

2. What is the course duration? How and when do you plan to complete the course?

We generally cover our courses in 60 hours, however, we are aware that we can't put a hard- stop to learning with a number. Our trainer will make sure that you have learned everything that is part of the curriculum. This could mean 48 hours or 60 hours, doesn't matter.

3. What is the material provided in the training?

We have industry standard course material which is used by our trainers to train you. At the end of the training apart from the notes which you have taken during the course, we will also provide you with the training material which was used. This training material includes the training content, interview questions, etc.

4. Do you help in preparing for the interview?

Our trainers are working professionals who work in MNCs. They are the expert in their domain and they know exactly what an interviewer looks into a candidate. Our expert trainers apart from sharing the interview questions will also conduct mock interviews to help you prepare for the real interview.

5. Who are your trainers?

Our trainers are industry experts who work in their respective technologies day in and day out. They work in MNCs and are technology experts within their organizations.

6. **What is the total batch size per course?**

We maintain a strict batch size of a maximum of 5 students. We also provide exclusive one-to-one training as well. Talk to our training partner to get more details.

7. Do you provide certification for the course?

Yes, at the end of training we provide a certification of completion.

8. Will I be joining a new batch or being merged with another batch?

You will be added to a new batch.

9. Is fast-track training available?

Yes, we provide fast-track training as well for those who want to complete the course faster. The curriculum and the total hours required to complete the course will remain the same. However, the trainer will be spending more hours with you to complete the course.

10. Do you assist in job placement?

Our trainers are expert professionals in their organizations and they often act as the interviewer to hire new candidates. Our trainers will help you prepare your resume with industry standards. After all, they know exactly what to look for in a resume.

11. Timings for training - Regular training/weekend training

We provide both regular and weekend training. Talk to our training partner to learn more about the timings.

12. Will you be working on a live project during training?

Yes, apart from doing the hands-on practice our trainer will also be taking a real-world project and working with you for the implementation.

13. What happens if I miss a class?

If you miss a class the content of that class will be taught to you again. With us, you might miss a class but not the content.

14. Can I attend a demo before the actual class?

Yes, absolutely! Talk to our training counselor on phone at **+91-9513216462** or email us at **info@emexotechnologies.com** to arrange a free demo. You can also fill in the contact us form below and we will call you to discuss your training requirements.