

MERN STACK TRAINING COURSE

Training duration : 10 Day(s) / 70 Hour(s)

GOALS

- Understand and implement a MERN stack- Explore alternative solutions
- Design an optimized architecture- Anticipate maintenance and evolution
- Master an end-to-end web application- Share and communicate concepts with others
- Advance towards application and functional design

PREREQUISITES

Knowledge of JavaScript, HTML/CSS, SQL

This training can only be funded as part of a company project (company or OPCO coverage). Applications for personal funding and CPF are not accepted.

PUBLIC

Developer

MERN STACK TRAINING COURSE

Training duration : 10 Day(s) / 70 Hour(s)

PROGRAM

REACT OVERVIEW

- What is React?
- What's in a Name?
- React Component Model
- What React Is Not
- What You Will Not Find in React
- Motivation for Creating React
- A React JavaScript Example
- One-Way Data Flow
- JSX
- A JSX Example
- The Virtual (Mock) DOM
- Only Sub-components that Actually Change are Re-Rendered
- React Libraries

ES6 IN A NUTSHELL

- What is ES6?
- ES6 Features
- Using ES6
- Transpiling
- Major Syntax Changes
- let and const
- Variable Scope
- Shadowing Variables
- Arrow Functions
- Arrow Functions As Parameters
- Using 'this' Within Arrow Functions
- Template Literals
- Spread Operator
- ES6 Classes
- Declaring Classes
- Declaring Instance Methods
- Accessor Methods
- Static Methods
- Inheritance With Classes

BABEL COMMAND-LINE INTERFACE

- Babel Transpiler
- Usage Options
- Presets and Plug-ins
- Babel CLI Installation
- Babel Configuration
- Running Babel Command-Line
- A Basic ES6 Development Setup with Babel
- Test the Babel Development Setup
- Adding React to the Development Setup
- Create a Minimal React App – Index.html
- Create a Minimal React App – app.js

MERN STACK TRAINING COURSE

Training duration : 10 Day(s) / 70 Hour(s)

BASIC COMPONENTS AND JSX

What is JSX?

JSX Transpilation to React Code Example

Running the Transpiled Code Babel

The Babel Runtime JavaScript Library

Script Import Skeleton Code

Playing Around in CodePen

React Components

Ways to Create UI Components

Creating a Functional Component Example

Component Names Must Be Capitalized

Creating a React Class-Based Component in ES5

The render Method

Creating a UI Component Using ES6 Class Notation

Using ES6 Classes with React

Which UI Component Creation Syntax Should I Use?

Components vs Elements

Elements Are Immutable

Properties

Property Naming Convention

Properties Default to 'True'

Spread Attributes (an ES6 Feature)

Expressions

REACT FUNCTIONAL COMPONENT CONCEPTS

Functional Components

Nesting JSX Elements

Example of JSX Nesting

Comments in JSX Code

Setting CSS Styles Using Classes

Setting CSS Styles Directly

JSX Escapes Values

Working with Lists of Items

Keys in Lists

Example List With Key

Container vs. Presentational Components

State

Types of State Data

State Hierarchy

Lifting State Up

Props vs. State

Pass Down a Function

Immutability

Immutability – Why?

Virtual DOM and State

Setting state

Updating Input fields

Passing Props to Components

Passing Functions to Components

Event Handling

Event Handler Example

MERN STACK TRAINING COURSE

Training duration : 10 Day(s) / 70 Hour(s)

Event Binding – DOs
 Event Binding – Don'ts
 Passing Parameters to Event Handlers
 Component Life-cycle
 Life-cycle in Functional Components
 App Development Workflow – 1/3
 App Development Workflow – 2/3
 App Development Workflow – 3/3Fragments

REACT COMPONENTS WITH ES6 CLASSES

Classes in ES6
 Functional Components
 Extending React.Component
 The render() Method
 state
 props
 defaultProps
 propTypes
 Component Lifecycle
 Component Life-cycle: Overview
 Component Life-cycle – Render Phase
 Component Life cycle – Commit Phase
 Component Life-cycle – Unmounting
 constructor() example
 componentDidMount() example
 setState(newStateValue)

REACT ROUTER

Routing and Navigation
 react-router
 Creating a react-router based project
 A Basic Routed Component
 Router vs. BrowserRouter
 The Route component
 <Switch>
 Redirect Route
 Navigating with <Link>
 Navigating with <NavLink>
 Route Parameters
 Retrieving Route Parameters
 QueryString Parameters
 Using Router with Redux

STATE MANAGEMENT FOR REACT

React State Basics – Props and State
 Props
 State in Class Based Components
 Managing State with Hooks in Functional Components
 The Problem with Props and State
 Redux State Library

MERN STACK TRAINING COURSE

Training duration : 10 Day(s) / 70 Hour(s)

Redux Advantages
Redux Disadvantages
Basic Rules for State Management
Types of State
Data State
Communication State
Control State
Session State
Location State
Location State Side Effects

USING REACT HOOKS

Functional Component Shortcomings
Hooks Overview
Hook Rules
React Linter Example
Functional Component Props
The useState Hook
Functional Component using the useState hook
useState with Multiple Variables
useState can also be used with Objects
The useEffect Hook
useEffect Hook Example
Using useEffect Hook to Load Data
Restricting when useEffect is Called
The useContext Hook
Additional Hooks
The useReducer Hook
An Example Reducer Function
Calling and Using useReducer
The useMemo Hook
useMemo Example
The useCallback Hook
useCallback Example
The useRef Hook
Using useRef to Hold Values
The useImperativeHandle Hook
useImperativeHandle Hook Example
The useEffect Hook

UNIT TESTING REACT WITH REACT TESTING LIBRARY

React Testing Framework
Features
Snapshot Testing
Code Coverage
Interactive Mode
Projects created with create-react-app
Default App Component Test
Unit Tests
Anatomy of a Unit Test
Common Matchers
Combining Tests

MERN STACK TRAINING COURSE

Training duration : 10 Day(s) / 70 Hour(s)

Running Tests
 Testing Promise based async code with 'done'
 Setup and Teardown
 react-testing-library
 A Simple Component Test
 A Simple Snapshot Test
 Running and Updating Snapshot Tests
 Building Component Tests
 Calling Render
 Render Properties
 Simulating Events
 Testing Results
 Using Query Functions
 Text Matching
 Counter Component
 counter-test.js

INTRODUCTION TO MONGODB

MongoDB
 MongoDB Features
 MongoDB on the Web
 Positioning of MongoDB
 MongoDB Applications
 MongoDB Data Model
 MongoDB Limitations
 MongoDB Use Cases
 MongoDB Query Language (QL)
 The CRUD Operations
 The find Method
 The findOne Method
 A MongoDB Query Language (QL) Example
 Inserts
 MongoDB vs Apache CouchDB

WORKING WITH DATA IN MONGODB

Reading Data in MongoDB
 The Query Interface
 Query Syntax is Driver-Specific
 Projections
 Query and Projection Operators
 MongoDB Query to SQL Select Comparison
 Cursors
 Cursor Expiration
 Writing Data in MongoDB
 An Insert Operation Example
 The Update Operation
 Update Operation Options
 An Update Operation Example
 A Remove Operation Example
 Limiting Return Data
 Data Sorting
 Aggregating Data

MERN STACK TRAINING COURSE

Training duration : 10 Day(s) / 70 Hour(s)

Aggregation Stages
Accumulators
An Example of an Aggregation Pipe-line
Map-Reduce
Data Backup

INTRODUCTION TO NODE.JS

What Is Node.js?
Applications of Node.js
Installing Node.js and NPM
« Hello, Node World ! »
How It Works
Node.js is built on JavaScript: Benefits
Traditional Server-Side I/O Model
Disadvantages of the Traditional Approach
Event-Driven, Non-Blocking I/O
Concurrency
Using Node Package Manager (NPM)
The Express Server Framework

BASIC WEB APPLICATION DEVELOPMENT

Introduction to the HTTP Module
The Request Handler Callback Function
The Server Object
Example Use of Server Object
The Request Object
The Response Object
Parsing Request Body
Serving Static Files
The HTTP Client API
Making POST/PUT/etc. Requests
Where To go from Here?

INTRODUCTION TO EXPRESS

Introduction to Express
Basic Routing Example
Defining Routing Rules
Route Path
The Response Object
Supplying URL Parameters
Ordering of Routes
Defining Catch All Route
Full Example Web Service

EXPRESS MIDDLEWARE

Introduction to Express Middleware
Writing a Middleware Function
Binding to a Path

MERN STACK TRAINING COURSE

Training duration : 10 Day(s) / 70 Hour(s)

- Order of Execution
- Raising Error
- Handling Error
- Serving Static Files
- Handling POST Request Body
- Enable Response Compression

ACCESSING MONGODB FROM NODE.JS

- Getting Started
- The Connection URL
- Obtaining a Collection
- Inserting Documents
- Updating a Document
- Querying for Documents
- Deleting a Document
- Connection Pooling

BUILDING REACT APPS WITH REDUX

- Redux
- Redux Terminology
- Redux Principles
- Redux: Actions
- Redux Action Types
- Action Creators
- Dispatching Actions
- Data Flow Basics
- Redux Reducers
- Pure Functions
- Reducer Example
- Returning Default State
- Creating a Development Environment with create-react-app
- Using Redux with React
- Initializing the Store
- Immutability
- Benefits of Immutable State
- Mutability of Standard types
- Copying Objects in JavaScript
- Copying Arrays in JavaScript
- One Store – Multiple Reducers
- Combining Reducers
- Components and Redux
- The React-Redux Package
- Wrapping App with Provider
- mapStateToProps
- mapDispatchToProps
- Using Mapped Properties and Methods
- Wrapping Components with Connect
- Configure Store
- Programming Advice – MultiTab Console

INDEXES IN MONGODB

MERN STACK TRAINING COURSE

Training duration : 10 Day(s) / 70 Hour(s)

Indexes in MongoDB
Creating an Index
Listing Existing Indexes
The Sort Order
Using the Explain() Method
Main Index Types
The _id Field
Single and Compound Indexes
Multikey Indexes
Other Index Types
Index Properties
Examples of Creating Indexes with Index Properties

EVENTS IN NODE JS

Event Driven Programming
Event Driven Programming (Contd.)
Event Emitter
EventEmitter Class
EventEmitter Class – Inheritance
The Event Loop and Event Handler
Phases Overview
Event Handlers
Example (Using EventEmitter as an Object)
Example (Inheriting from EventEmitter)
EventEmitter Functions
Issue with 'this' Keyword in Callback Functions
Handling this Problem
Controlling Event Callbacks in the Event Loop

ASYNCHRONOUS PROGRAMMING WITH CALLBACKS

Callbacks
Creating a Callback Function
Calling The Callback Function
Callback – Another Example
Issue with 'this' Keyword in Callback Functions
Handling this Problem
Handling this Problem – Method 1 (Storing in Another Variable)
Handling this Problem – Method 2 (Using Bind Function)
Handling this Problem – Method 3 (Using ES6 Arrow Functions)
Error Handling without Callback
Error Handling with Callback
Asynchronous Callback
setImmediate() and nextTick()
API Example

ASYNCHRONOUS PROGRAMMING WITH PROMISES

Synchronous and Asynchronous
The Problems with Callbacks
Introduction to Promises

MERN STACK TRAINING COURSE

Training duration : 10 Day(s) / 70 Hour(s)

Requirements for Using Promises
Creating Promises Manually
Calling the Promise-based Function
Making APIs that support both callbacks and promises
Using APIs that support both callbacks and promises
Chaining then Method / Returning a Value or a Promise from then Method
Promisifying Callbacks with Bluebird
Using Bluebird
Bluebird – List of Useful Functions
Benefit of using Bluebird over ES6 for Promisification
Error Handling in Promise-based asynchronous functions

USER API ROUTES & JWT AUTHENTICATION

Creating The User Model
Request & Body Validation
User Registration
Implementing JWT
Custom Auth Middleware & JWT Verify
User Authentication / Login Route

DEPLOYMENT TO SERVER AND MONITORING

MERN STACK TRAINING COURSE

Training duration : 10 Day(s) / 70 Hour(s)

Prise en compte du handicap

Pour les personnes en situation de handicap, afin de nous permettre d'organiser le déroulement de la formation dans les meilleures conditions possibles, contactez-nous via notre formulaire de contact ou par mail (formation@access-it.fr) ou par téléphone (0320619506). Un entretien avec notre référente handicap pourra être programmé afin d'identifier les besoins et aménagements nécessaires.

Modalités et moyens Pédagogiques, techniques et d'encadrement mis en œuvre

Répartition théorie/pratique : 45%/55%. Cette formation se compose d'une alternance d'apports théoriques, de travaux pratiques s'articulant autour d'une application fil rouge, de démonstrations, de phases d'échanges entre participants et de synthèses de la part du formateur.

Formation accessible à distance de n'importe où et n'importe quand, via un ordinateur type PC disposant d'une connexion à Internet à haut débit (ADSL ou plus).

Pour assurer un démarrage dans les meilleures conditions au premier jour de la formation, notre service logistique se met systématiquement en relation, en amont, avec vous afin de réaliser un test de validation technique et de vous présenter l'environnement de formation.

Pendant toute la durée de la formation, le stagiaire dispose d'une assistance technique et pédagogique illimitée, par e-mail, avec un délai de prise en compte et de traitement qui n'excède pas 24h. En complément, le stagiaire peut planifier un rendez-vous pédagogique avec un formateur expert afin d'échanger sur des éléments de la formation.

La durée de la formation affichée sur cette page est une durée estimée qui peut varier en fonction du profil du stagiaire et de ses objectifs (notamment s'il souhaite valider sa formation par le passage d'un examen de certification).

Durant la formation, le formateur prévoit :

Des démonstrations organisées en modules et en séquences découpées le plus finement possible, en suivant le programme pédagogique détaillé sur cette page ;

Des énoncés et corrigés de travaux pratiques à réaliser tout au long de la formation ;

Des travaux pratiques sont proposés ; la plateforme prévoit l'environnement technique nécessaire à la réalisation de l'ensemble des travaux pratiques ;

Le formateur valide les connaissances acquises après chaque TP ;

Il est proposé un ou plusieurs livres numériques faisant office d'ouvrage(s) de référence sur le thème de la formation.

Validation et sanction de la formation

Une attestation mentionnant les objectifs, la nature et la durée de l'action et les résultats de l'évaluation des acquis de la formation sera remise au stagiaire à l'issue de sa formation par courrier électronique.

A la demande, il sera délivré un certificat de réalisation.

Type de formation

Professionalisante ayant pour objectif le perfectionnement, l'élargissement des compétences

Moyens permettant de suivre l'exécution de l'action

Le contrôle de la présence des stagiaires sera assuré par la vérification de l'assiduité des participants. Le stagiaire signera une feuille de présence par demi-journée de formation. Celle-ci sera également signée par le formateur.

Modalité d'évaluation des acquis

Durant la formation, le participant est amené à mettre en pratique les éléments du cours par la réalisation de travaux pratiques réalisés sur PC.

La validation des acquis du stagiaire est faite par le formateur à la fin de chaque atelier. Cette validation individuelle est possible du fait du faible nombre de participants par session de formation (6 personnes maximum).

À la fin de la formation, le stagiaire a donc atteint les objectifs fixés par la formation.

En complément, pour les stagiaires qui le souhaitent, certaines formations peuvent être validées officiellement par l'éditeur en passant un examen de certification.

Access it étant centre d'examen ENI, les examens peuvent être réalisés sur demande à distance ou dans nos locaux de Villeneuve d'Ascq.

Les candidats à la certification doivent produire un travail personnel important en vue de se présenter au passage de l'examen, le seul suivi de la formation ne constitue pas un élément suffisant pour garantir un bon résultat et/ou l'obtention de la certification

Assistance Post-Formation

Toute personne ayant suivi une formation avec Access it bénéficie d'une assistance post-formation d'une durée de 1 mois. Ce nouveau service d'accompagnement permet aux stagiaires rencontrant des difficultés dans la mise en œuvre des connaissances acquises de solliciter l'aide de nos formateurs sur des aspects relatifs aux programmes de formation suivis. Pour en bénéficier, il suffit de se rendre sur la page contact de notre site web et de remplir le formulaire. Une réponse est apportée par mail ou par téléphone dans un délai de 48 heures.

Centre d'Examen

Notre centre de formation agréé Qualiopi bénéficie de l'agrément d'ENI et ICDL pour les certifications informatiques. C'est pour nos clients la garantie de pouvoir suivre des formations préparant à des certifications professionnelles.

Aide à l'orientation

Pour chacune des grandes thématiques couvertes par notre offre de formation, nous proposons via nos spécialistes un rendez-vous physique ou téléphonique qui via un diagnostic permettra aux personnes souhaitant être accompagnées dans le choix d'un programme ou dans la définition d'un parcours de formation une orientation vers les programmes les plus adaptés à leurs besoins et à leur niveau.

Aspects Pratiques

Dès leur inscription, les participants sont contactés par nos services qui s'assurent que les débits internet constatés sur le lieu depuis lequel ils souhaitent se former sont suffisants pour suivre la formation dans des conditions optimales.

À l'occasion de cet appel, nos experts s'assurent également qu'ils disposent du matériel nécessaire pour suivre la formation (PC Portable, webcam, Micro-casque..).

Avant le début de la formation, les participants reçoivent un lien leur permettant d'accéder à la classe virtuelle et leurs identifiants personnels de connexion. Un aide à l'utilisation de la solution de visioconférence utilisée leur est également proposée.

Le jour de la formation, ils se connectent à la classe virtuelle depuis leur navigateur internet. Ils voient et entendent le formateur ainsi que les autres participants et peuvent à tout moment communiquer avec eux.

Ils participent aux échanges et réalisent les ateliers dans les mêmes conditions que s'ils étaient en salle. Grâce à nos outils de prise en main à distance, les formateurs peuvent à tout moment prendre la main sur leurs postes pour les aider ou vérifier leurs TP.

Tout au long de la formation, les participants peuvent bénéficier de l'assistance immédiate de nos experts en composant le numéro qui leur a été communiqué avant la formation.

Des bilans intermédiaires ont lieu en présence des participants du formateur et du référent pédagogique d'Access it afin de vérifier l'état d'avancement de la session, les difficultés rencontrées et permettre d'éventuels actions correctives.

Bénéfices pour les participants

Se former depuis leur lieu de travail ou leur domicile,

Accéder sans se déplacer à la qualité d'une formation délivrée par un formateur consultant ayant une expérience probante sur le sujet animé.

Bénéficier à distance de la richesse d'une formation interentreprises : échanges avec le formateur et les autres participants, partages d'expériences, ateliers pratiques...

Pouvoir se former en toutes circonstances et notamment en cas d'imprévu.

Bénéfices pour l'entreprise

Optimiser ses budgets en limitant les frais de déplacement et d'hébergement.

Proposer à tous ses collaborateurs, quelle que soit leur situation géographique, des formations de qualité (en Inter comme en Intra).

Limitier les temps de déplacement.

Proposer davantage de choix dans les formations à des collaborateurs peu mobiles.

Assurer la montée en compétences de ses collaborateurs quelles que soient les circonstances