

## ABOUT THE TRAINING

During the course of this training, we will take an in depth look at software development using JavaScript ES6+, React, Redux, Node.js and MongoDB along with the basic building block of HTML5, CSS3 and Bootstrap 4. Each participant in this training will learn to build a full stack application which will be hosted on Heroku. We will also be deploying codes to a GitHub library as the training progresses. We will start with a blank text editor and end with a deployed full stack application.

# SOFTWARE DEVELOPMENT TRAINING WITH JAVASCRIPT



## ALABIAN SOLUTIONS LTD



Alabian Solutions Limited

Infinite Grace Plaza

Plot 4 Oyetubo Street

off Obafemi Awolowo Way

Ikeja Lagos

## ABOUT GIT

Git is a distributed version control system used for tracking changes made to files and also for coordinating work done on those files among multiple people. It is primarily used for source-code management in software development. Participants will learn how to use these tools for working in team

## WHY GIT

Git is a powerful version control system that allows you to keep track of changes made to your codebase over time. It helps you to collaborate with others more efficiently by tracking changes, merging code changes, and resolving conflicts.

## COURSE OUTLINE

### Introduction to Version Control

- ❖ What is Version Control
- ❖ Why is Version Control Important
- ❖ Types of Version Control

### Git Branching

- ❖ Understanding Branching
- ❖ Creating and Deleting Branches
- ❖ Merging Branches
- ❖ Merging Branches
- ❖ Resolving Merge Conflicts

### Introduction to Git

- ❖ What is Git
- ❖ Installing Git
- ❖ Configuring Git
- ❖ Git workflow

### Git Remote

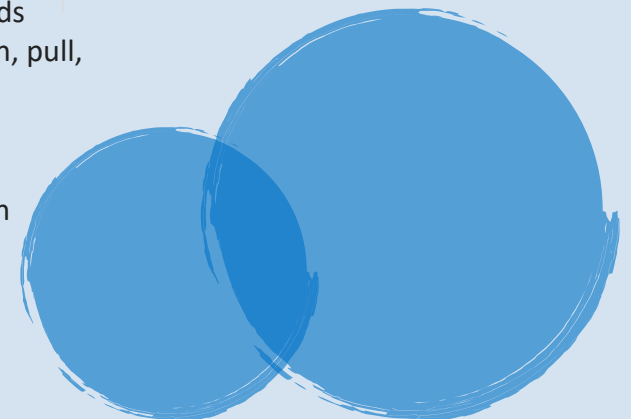
- ❖ Adding Remote Repositories
- ❖ Pushing and Pulling Changes
- ❖ Fetching and Pulling Changes

### Git Basics

- ❖ Creating a Git Repository
- ❖ Cloning a Git Repository
- ❖ Basic Git Commands (add, commit, push, pull, status)

### Git Collaboration

- ❖ Collaborating with Others
- ❖ gitignore



# COURSE OUTLINE

## Overview of HTML

- ❖ HTML Tags
- ❖ HTML Elements
- ❖ HTML Attributes
- ❖ HTML Comments
- ❖ Paragraphs
- ❖ Text Formatting
- ❖ Quotation
- ❖ Links
- ❖ Images
- ❖ Tables
- ❖ Lists
- ❖ Block and Inline Elements
- ❖ HTML Style Guide
- ❖ Iframes
- ❖ File Paths
- ❖ Audio/Video
- ❖ Viewport meta tag for Responsive web Design

## ABOUT HTML

Participants will learn the basics of HTML5, the training will start with instruction on how set up their IDE (Integrated Development Environment) and upload files to a live web server with an ftp client. They will go over HTML5 basics like valid document structure, and the usefulness of writing good HTML5 semantic tags. Each participant is expected to produce a webpage using the tags learnt in this module.

## WHY HTML

HTML is the foundation of web development, and understanding it is essential for anyone who wants to build web applications. Knowing HTML will help you understand how web pages are structured, how they are displayed in a web browser, and how they interact with other web technologies.?

## ABOUT CSS

We will teach participants from the very basics of CSS3 to some fairly advanced concepts like floating, CSS rule conflict resolution, flexbox etc. We will also be taught the 'box model', background property, CSS3 font and even browser developers' tool. This module will be culminated by converting the webpages created in HTML5 modules into a beautiful website.

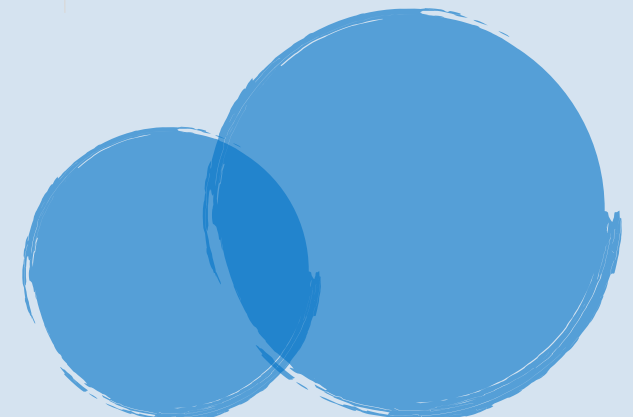
## WHY CSS

Knowledge of CSS is a valuable skill in the web development industry, and it can help you advance your career. Many job postings for web developers require knowledge of CSS, and employers are always looking for candidates who can create visually appealing and responsive web pages.

## COURSE OUTLINE

### INTRODUCTION TO CSS

- ❖ Benefits of CSS
- ❖ CSS Syntax
- ❖ How to add CSS to HTML
- ❖ Code commenting in CSS
- ❖ CSS Selectors (ID, Class, Tags, Attributes)
- ❖ CSS Selectors
- ❖ Pseudo-classes
- ❖ Pseudo Elements
- ❖ Applying Colors
- ❖ Background Styling
- ❖ CSS-Text Fonts
- ❖ Height and Width
- ❖ Position
- ❖ Float
- ❖ Flex
- ❖ CSS Borders
- ❖ Box model
- ❖ Margins and Padding
- ❖ Outline in CSS
- ❖ Text Formatting
- ❖ Fonts styling
- ❖ Using Icons
- ❖ Lists
- ❖ CSS Tables
- ❖ DropDowns





# B

## COURSE OUTLINE

### Layout

- ❖ Breakpoints
- ❖ Containers
- ❖ Grid
- ❖ Columns
- ❖ Gutters
- ❖ Utilities
- ❖ Z-index

### Forms

- ❖ Form control
- ❖ Select
- ❖ Checks & radios
- ❖ Input group
- ❖ Input group
- ❖ Layout

### Components

- ❖ Accordion
- ❖ Alerts
- ❖ Badge
- ❖ Buttons
- ❖ Button group
- ❖ Card
- ❖ Carousel
- ❖ Close button
- ❖ Collapse
- ❖ List group
- ❖ Modal
- ❖ Navbar

### Content

- ❖ Images
- ❖ Tables
- ❖ Grid
- ❖ Figures

## ABOUT BOOTSTRAP

In this training, we are going to show participant how to build a mobile friendly Web Application with minimal effort using the latest version of Bootstrap. It will amaze you to see how we can simplify the task of Front-end development using Bootstrap.

## WHY BOOTSTRAP

Bootstrap provides a set of pre-built components and styles that can save a significant amount of time when building web pages. By using Bootstrap, you can focus on customizing and fine-tuning your web pages rather than building everything from scratch.

## COURSE OUTLINE

### Introduction to JavaScript Data Types JavaScript Operators

- ❖ Overview of JavaScript
- ❖ Developer console
- ❖ Adding JavaScript into HTML
- ❖ JavaScript Statements
- ❖ Basic Syntax
- ❖ JavaScript Comments
- ❖ Interaction: alert, prompt, confirm
- ❖ Numbers
- ❖ Strings
- ❖ Boolean
- ❖ Array
- ❖ Objects
- ❖ Null
- ❖ Undefined
- ❖ Arithmetic Operators
- ❖ Assignment operators
- ❖ Comparison Operators
- ❖ JavaScript Unary Operators
- ❖ Ternary Operator
- ❖ Instanceof Operator

### Variables in JavaScript

- ❖ The use of variables?
- ❖ JavaScript Let
- ❖ JavaScript Const
- ❖ JavaScript Reserved Words

### Conditionals

- ❖ If-else
- ❖ Switch Case in JavaScript

## ABOUT JAVASCRIPT

For some of the participants JavaScript will be the first programming language they will be learning. Our instructors will simplify the act of programming. JavaScript will add interactivity to web pages. The module will cover just about everything - from common language constructs and data types to objects, functions, arrays, closures, Document Object Model (DOM) and Browser Object Model (BOM) API. Apart from just adding interactivity to a webpage we teach participant how to use JavaScript to create Web Applications and prepare them for other JavaScript Library they will learn in the training.

## WHY JAVASCRIPT

JavaScript is the backbone of front-end web development, and it is used to build user interfaces, control animations, and handle user input. By learning JavaScript, you can create visually appealing and responsive user interfaces that enhance the user experience.

## Javascript functions

- ❖ User-defined functions
- ❖ Anonymous Functions
- ❖ Functions as First-Class Citizens
- ❖ Callback Functions
- ❖ Inbuilt Functions
- ❖ Understanding variable scopes in JavaScript
- ❖ this in JavaScript

## Event handling

- ❖ Introduction to Events
- ❖ Page Load Events
- ❖ The onLoad Event
- ❖ The DOMContent
- ❖ Mouse Events
- ❖ Keyboard Events
- ❖ Scroll Events

## Document Object Model (DOM) Traversing elements

- ❖ Introduction to
- ❖ Document Object Model
- ❖ Select Element By Id
- ❖ Select Elements By Name
- ❖ Select Elements By Tag Name
- ❖ Select Elements By Class Names
- ❖ Select Element By CSS Selectors

## DOM Manipulating elements

- ❖ Create Elements
- ❖ Append Child Nodes
- ❖ InnerHTML

## Javascript Form

- ❖ Get the Parent Element
- ❖ Get Siblings of an Element
- ❖ Get Children of an Element
- ❖ How to Create a Form with the JavaScript?
- ❖ Form validation

## ES6

- ❖ Template Literals
- ❖ Object Literal Syntax
- ❖ Default Parameters
- ❖ Rest Parameters
- ❖ Spread Operator
- ❖ Destructuring

## Asynchronous JavaScript

- ❖ Promises
- ❖ Async/await

## Error Handling

- ❖ try...catch try...catch...finally
- ❖ Throw an Exception

## ABOUT REACT

React.js is a JavaScript library that is used to build interactive User Interfaces (UI) without hassles. It makes creating interactive UI's fun. Built and maintained by Facebook; React is spreading crazily across the industry, so fast that everybody is hiring a React developer.

## WHY REACT

Learning React is a valuable skill for anyone interested in front-end web development or software engineering. Its reusability, performance, flexibility, community support, job opportunities, and scalability make it a popular and essential tool for building modern web applications.

## COURSE OUTLINE

### Introduction to React

- ❖ What is React JS?
- ❖ Why use React instead of JavaScript
- ❖ Project Overview
- ❖ Roadmap

### React Setup

- ❖ Environment Setup
- ❖ Creating New Project
- ❖ Analyzing Project Structure

### React basics and components

- ❖ Components & Why React is Component-Based?
- ❖ JSX
- ❖ React Elements
- ❖ Building Component
- ❖ Styling Components
- ❖ Dynamic Data in components
- ❖ Passing data using Props

### State and event handling Contents conditional rendering

- ❖ Handling Events in React
- ❖ Why are States required?
- ❖ useState() Hook
- ❖ Form Handling
- ❖ Handling form Submission
- ❖ Rendering of List
- ❖ Importance of key attribute
- ❖ Conditional Statements & Output

### Handling side effects

- ❖ What are side effects & useEffect()
- ❖ useEffect() hook
- ❖ Cleanup function

## HTTP Calls using Axios/fetch

- ❖ Making first network Call with fetch
- ❖ Network call using Axios
- ❖ Handling data and states
- ❖ Async and Await
- ❖ Updating the data

## Redux

- ❖ Introduction to Redux
- ❖ Actions
- ❖ reducers
- ❖ The Redux store
- ❖ dispatch
- ❖ Connecting Redux to React

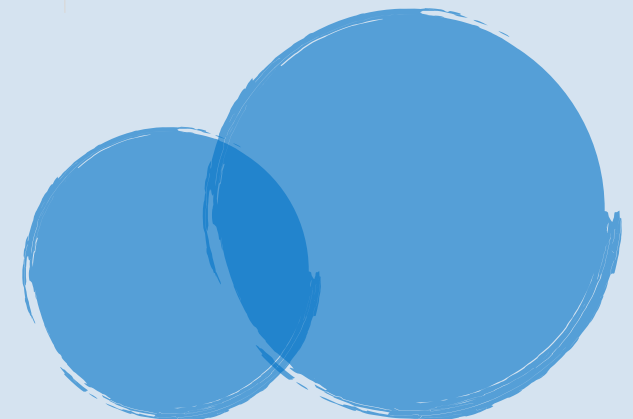
## Deployment

## Multipage Single page application with react-router

- ❖ Introduction to routing
- ❖ What is Routing, how it works & installation
- ❖ Basic Routing
- ❖ Link and NavLink
- ❖ Dynamic Routes
- ❖ Nested Routes
- ❖ Redirection and Not Found Page

## Global state management with context

- ❖ What is React Context?
- ❖ React context vs Redux
- ❖ When to use React Context?
- ❖ Application of react context



## ABOUT NODE JS

Node.js is a Server-side framework that is written for JavaScript programming. The fact that you can now write JavaScript both at the Front-end and Back-end makes JavaScript a much sought-after programming language. A solid understanding of JavaScript will make learning Node.js a breeze. Applications built with Node.js are efficient, fast and performs high.

## WHY NODE JS

learning Node.js is a valuable skill for anyone interested in back-end web development or software engineering. Its use of JavaScript, scalability, speed, cross-platform compatibility, large ecosystem, community support, and job opportunities make it a popular and essential tool for building modern server-side applications.

## COURSE OUTLINE

### Node.js

- ❖ What is Node.js
- ❖ Node.js Process Model
- ❖ Install Node.js
- ❖ Node.js Console/REPL
- ❖ Node.js Basics
- ❖ Node.js Modules
- ❖ Local Modules in Node.js
- ❖ Export Modules in Node.js
- ❖ Node Package Manager (NPM)
- ❖ Create Web Server in Node.js
- ❖ Node.js File System
- ❖ Debugging Node.js Application
- ❖ Node Inspector for Debugging
- ❖ Node.js Application
- ❖ Node.js EventEmitter
- ❖ Serving Static Files from Node.js
- ❖ Database Access
- ❖ Node.js Database Access
- ❖ Access MongoDB in Node.js

### Express

- ❖ Overview
- ❖ Installation
- ❖ Routing
- ❖ HTTP
- ❖ Methods
- ❖ URL Building
- ❖ Middleware
- ❖ Templating
- ❖ Static
- ❖ Files
- ❖ Form Data
- ❖ Database
- ❖ Authentication
- ❖ RESTful
- ❖ APIs
- ❖ Error handling
- ❖ Debugging

### Deployment



## ABOUT MONGODB

MongoDB is a Database Management System that lets you store your application data. When building software with JavaScript Technologies, we use React to create the User Interface, Node.js to write a program to store, remove, edit and delete data in our MongoDB Database.

## WHY MONGODB

MongoDB is a flexible database that can store data in a variety of formats, including JSON, BSON, and CSV. This makes it easy to work with different types of data, such as text, images, and multimedia content.

## COURSE OUTLINE

### Introduction

- ❖ What is MongoDB
- ❖ Install MongoDB
- ❖ MongoDB Basics
- ❖ MongoDB Shell
- ❖ MongoDB Data Types

### CRUD Operation

- ❖ Inserting document
- ❖ insertOne
- ❖ insertMany

### Selecting documents

- ❖ findOne
- ❖ find

### Comparison query operation Logical query operator Element query operators

- |  |                               |            |
|--|-------------------------------|------------|
| ❖ \$eq: Equal To Operator                  | ❖ \$and: Logical AND Operator | ❖ \$exists |
| ❖ \$lt: Less Than Operator                 | ❖ \$or: Logical OR Operator   | ❖ \$type   |
| ❖ \$lte: Less Than or Equal To Operator    | ❖ \$not: Logical NOT Operator |            |
| ❖ \$gt: Greater Than Operator              | ❖ \$nor: Logical NOR Operator |            |
| ❖ \$gte: Greater Than or Equal To Operator |                               |            |
| ❖ \$ne: Not Equal To Operator              |                               |            |
| ❖ \$in: In Operator                        |                               |            |
| ❖ \$nin: Not In Operator                   |                               |            |



## Deleting documents

- ❖ deleteOne
- ❖ deleteMany

## Aggregation

- ❖ Aggregation Pipeline
- ❖ \$avg
- ❖ \$count
- ❖ \$sum
- ❖ \$max
- ❖ \$min

## Array query operators

- ❖ \$size
- ❖ \$all
- ❖ \$elemMatch

## Indexes

- ❖ MongoDB Indexes
- ❖ Create Index
- ❖ Unique Index
- ❖ Compound index
- ❖ Drop Index

## Using Mongoose

## Deployment

## Sorting and limiting

- ❖ sort(): Sorting documents
- ❖ limit(): Limiting documents

## Updating documents

- ❖ updateOne: Update one Document
- ❖ updateMany: Update Multiple Documents
- ❖ \$inc: Increase / Decrease Field Value
- ❖ \$min: Update Field Value
- ❖ \$max: Update Field Value
- ❖ \$mul: Mutiply Field By a Number
- ❖ \$unset: Remove Fields
- ❖ \$rename: Rename Fields
- ❖ Upsert