

# FRONT END WEB DEVELOPER

## COURSE SYLLABUS

WWW.BESTSHINEWEBSOFT.IN

WWW.BESTSHINEACADEMY.CO.IN



**Best Shine IT Training Center**  
*simply easy learning*

## COURSE FLOW

BSA Front-end Web developer course starts with introducing participants with front-end technologies (ex: HTML5 & CSS3) and makes them develop responsive one page web application. To cater modern day Front-end requirements (ex: responsive design) getting hands-on with latest frameworks (ex: Bootstrap and ReactJS) becomes important. This course deep dives into frameworks to ensure participants implement front end that scales well across PC, mobile and tablet screens.

Throughout the course foundation paradigms (ex: Design patterns) are stressed to ensure participants clearly understand software engineering fundamentals. Sessions are interleaved for resume and interview preparation for placements.

## MODULE TOPICS

- Single page web application development (HTML5 | CSS3)
- JavaScript for Front-end (JavaScript | jQuery)
- Front-end Frameworks (Bootstrap | ReactJS)
- Foundation paradigms (OOPS | Design Patterns | Object Modelling | JSON | AJAX)
- Backend Integration
- DISHA (Resume & Interview prep package)

## DURATION

4 months

## PLATFORM

- Linux (Fedora / Mandriva /Ubuntu) or Windows Host system
- Firefox, Chrome Browsers

## PREREQUISITES

B.E / B.Tech / M.E / M.Tech / MCA /Bsc

## DELIVERY METHOD

Live Virtual Classes and Intervention Sessions (Assignment / Project Explainer, Live Doubt Clearing and Evaluation)



## Hyper Text Mark-up Language (HTML5)

### Module Objectives:

- BuildingStrong expertise to develop front end application using HTML5
- Implement MVC and responsive design to scale well across PC, tablet and Mobile Phone

### Overview:

Thiscourse provides you hands-on experience and exposure to developing HTML5 based single page application for browsers. This course builds strong foundation on HTML5 which will help developer to use HTML5 concepts for building responsive web application.

### Platform:

- Linux(Fedora / Mandriva / Ubuntu) or Windows Host system
- Firefox, Chrome

### Prerequisites:

Academiclevelweb application knowledge

### Delivery method:

Instructorlead, hands-on exercises backed with assignments and mini project

### DetailedCourse Contents:

- IntroductionHTML
- HTML Basics
- HTML Elements
- HTML5 Semantic
- HTML Attributes
- HTML Headings
- HTML Paragraph
- HTML Styles
- HTML Formatting
- HTML Quotations
- HTML Computer Code
- HTML Comments & Colours
- HTML CSS, Links and Images
- HTML Lists
- HTML Blocks
- HTML Classes
- HTML Layout
- HTML Responsive
- HTML iframes
- HTML JavaScript
- HTML Head
- HTML Entities and URI Code
- HTML Symbols and XHTML
- HTML Charset and Forms



## Cascading Style Sheets (CSS3)

### Module Objectives:

- Building Strong expertise to develop front end application using CSS3
- Implement MVC and responsive design to scale well across PC, tablet and Mobile Phone

### Overview:

This course provides you hands-on experience and exposure to developing CSS3 based web application. This course builds strong foundation on CSS3 which will help developer to use CSS3 concepts for building responsive web application.

### Platform:

- Linux(Fedora / Mandriva / Ubuntu) or Windows Host system
- Firefox, Chrome

### Prerequisites:

Academic level web application knowledge

### Delivery method:

Instructor lead, hands-on exercises backed with assignments and mini project

### Detailed course contents:

<ul style="list-style-type: none"> <li>• Introduction CSS3</li> <li>• CSS3 Syntax</li> <li>• CSS3 How To</li> <li>• CSS3 Colours</li> <li>• CSS3 Backgrounds</li> <li>• CSS3 Boarders</li> <li>• CSS Padding</li> <li>• CSS Height/Width</li> <li>• CSS3 Gradients</li> <li>• CSS3 Shadows</li> <li>• CSS3 Text</li> <li>• CSS3 Fonts</li> <li>• CSS3 2D Transforms</li> <li>• CSS3 3D Transforms</li> <li>• CSS Links</li> <li>• CSS Lists</li> <li>• CSS Tables</li> <li>• CSS Box Model</li> <li>• CSS Outline</li> <li>• CSS Display</li> <li>• CSS Max-width</li> <li>• CSS Position</li> </ul>	<ul style="list-style-type: none"> <li>• CSS Float</li> <li>• CSS Inline-block</li> <li>• CSS Align</li> <li>• CSS Combinators</li> <li>• CSS Pseudo-class</li> <li>• CSS Pseudo-element</li> <li>• CSS Navigation Bar</li> <li>• CSS Dropdowns</li> <li>• CSS Tooltips</li> <li>• CSS3 Images</li> <li>• CSS Attr Selectors</li> <li>• CSS Forms</li> <li>• CSS Counters</li> <li>• CSS3 Animations</li> <li>• CSS3 Buttons</li> <li>• CSS3 Pagination</li> <li>• CSS3 Multiple Columns</li> <li>• CSS3 User Interface</li> <li>• CSS3 Box Sizing</li> <li>• CSS3 Filters</li> <li>• CSS3 Media Queries</li> <li>• CSS3 Responsive</li> </ul>
--	--



## JavaScript

### Module Objectives:

- Understand core features of JavaScript
- Build interactive and user-friendly frontend applications using HTML, CSS and JavaScript
- Apply OOP concepts by learning JavaScript
- Debug frontend applications using Google Chrome debugger.

### Overview:

This course provides you hands-on experience and exposure to develop frontend application using JavaScript. Starting with introduction, this course deep dives into core features like Event handling, Form handling and Regular expressions. Specific focus is given for Document Object Model (DOM) and manipulating HTML using various DOM APIs. Along with building JavaScript features this course builds sound foundations in Algorithms, Problem solving and Debugging techniques which is critical for a web application developer.

### Platform:

- Linux(Fedora / Mandriva / Ubuntu) or Windows Host system
- VSCode editor with Live server
- Google Chrome Debugger

### Prerequisites:

- HTML and CSS
- Basic understanding of OOPS

### Delivery method:

Instructor lead, hands-on exercises backed with assignments and mini project

### Detailed course contents:

#### Algorithms and Problem solving

- Problem solving - what?
- Introduction to SDLC
- Polya's rules
- Algorithm design methods
- Pseudo code creation
- Flowcharts

#### Introduction to JavaScript

- History of JavaScript
- Advantages
- Limitations
- Script element
- Creating your first JavaScript program
- Coding convention
- Setting up development environment (with VSCode)

#### Types and Statements

- Keywords in JavaScript
- Overview of Data types
- Primitive Data types Non-primitive Data types
- Conditional statements
- I/O in JavaScript
- Loops



**Operators**

- Introduction to operators
- Operator precedence and associativity
- Deep dive into operators
- Arithmetic
- Comparison
- Ternary
- Logical
- Language
- Bitwise

**Functions - Level I**

- Introduction to functions
- Function definition Passing
- values Returning values
- Robust parameter handling
- Local and global variables
- Functions as objects
- Function constructor
- 

**Functions - Level II**

- Function invocation patterns
- Recursion functions
- Generator functions Arrow
- functions Variadic functions
- JavaScript scopes Function
- closures
- 

**Arrays and Strings**

- Introduction to Arrays
- Array declaration
- Array access methods
- Multi-dimensional arrays
- String properties
- String access methods

**Regular Expressions**

- Introduction to RegExp
- Regular expression usage
- Modifiers
- RegExp patterns
- RegExp methods
- String methods for RegExp
- Type conversion in JavaScript

**Objects in JavaScript**

- Introduction to objects
- Type of objects in JavaScript
- Creating objects



- Object methods
- Constructor function
- Prototype in JavaScript
- Inheritance using prototype chain

#### Event handling

- JavaScript events
- Event handler
- Event flow
- Event bubbling and capturing
- Event listeners
- Event types

#### Document Object Model(DOM)

- Introduction to DOM
- Types of DOM
- DOM standards and methods
- Manipulating documents using DOM
- Handling images
- Table manipulation
- Animation
- Node and Node-list handling

#### Browser Object Model(BOM)

- Introduction to BOM
- DOM vs BOM differences
- Window object and methods
- BOM navigator
- BOM history
- BOM location
- BOM timer
- Introduction to Cookies
- Session and persistent cookies

#### Form Handling

- Introduction to forms
- Form processing
- Forms object
- Accessing data from forms
- Form validation
- Additional features in forms
- Validation APIs

#### Debugging Techniques

- JavaScript Errors
- Error handling mechanisms
- Introduction to Google Chrome debugger
- Deep dive into debugger window
- Introduction to Breakpoints
- Changing variable values in runtime
- Avoiding mistakes



## jQuery

### Module Objectives:

- Building Strong expertise to develop front end application using HTML5, CSS3 and JavaScript along with jQuery and AngularJS framework
- Implement MVC and responsive design to scale well across PC, tablet and Mobile Phone

### Overview:

This course provides you hands-on experience and exposure to develop jQuery Mobile based responsive web application. It builds strong foundation of jQuery which will help developer to apply concepts for responsive web frontend development.

### Platform:

- Linux(Fedora / Mandriva / Ubuntu) or Windows Host system
- Firefox, Chrome

### Prerequisites:

Academic level web application knowledge

### Delivery method:

Instructor lead, hands-on exercises backed with assignments and mini project

### Detailed course contents:

- Introduction to jQuery
- jQuery Syntax
- jQuery Selectors
- jQuery Events
- jQuery Effects
- jQuery HTML
- jQuery Traversing
- jQuery AJAX
- jQuery Misc.

## Bootstrap

### Module Objectives:

- To become proficient in Bootstrap concepts
- To develop a web pages based on Bootstrap

### Overview:

This course is targeted for fresh engineers or professional who want to build competency in the Bootstrap Based web development.

### Platform:

- Linux(Fedora / Mandriva / Ubuntu) or Windows Host system
- Firefox, Chrome

### Prerequisites:

Academic level web application knowledge

### Delivery method:

Instructor lead, hands-on exercises backed with assignments and mini project

### Detailed course contents:

- Introduction to Bootstrap
- Bootstrap Basics



- Bootstrap Grids
- Bootstrap Themes
- Bootstrap CSS
- Bootstrap JS

## React JS

### Module Objectives:

- Create Single Page Applications (SPA) using React
- Learn core React features and use-cases
- Integrate React Applications into Firebase backend
- Deploy React applications into popular servers

### Overview:

This course provides you building SPA using React and its core features by taking a hands-on approach. Starting with Introduction this course deep dives into core React features like components, state and event binding and rendering. Followed by core features, participants will learn to integrate React with backend services (ex: Firebase) and consuming HTTP services in forms of REST APIs. The course exposes participants with advanced features like Routing, Authentication and authorization followed by deploying into popular servers (ex: GitHub)

### Platform:

- Linux(Fedora / Mandriva / Ubuntu) or Windows Host system
- Firefox, Chrome Browsers
- Firebase(Backend integration)

### Prerequisites:

- OOPS concepts
- Knowledge of HTML, CSS and JavaScript

### Delivery method:

Instructor lead, hands-on exercises backed with assignments and mini project

### Detailed course contents:

#### Introduction to React

- History of React
- Key Benefits of React
- React development environment
- Creating your first React Application
- React Source code structure

#### JSX

- Introduction to JSX
- Coding in JSX
- Expressions in JSX
- Working with HTML
- Conditional Constructs

#### Components

- Introduction to components
- Why Components?
- Writing JSX code in components
- Adding CSS



- Populating Data Dynamically
- Passing data through "props"
- Multiple Components

### State and Event Binding

- Introduction to Events
- Event Handlers
- Working with state
- Data Binding
- Controlled and Uncontrolled Components

### Rendering Lists and Conditional Contents

- Rendering lists of data
- Using stateful list
- Keys in data
- Conditional Contents
- Adding dynamic styles

### Debugging Techniques

- Understanding error message
- Code flow and warnings
- Breakpoints
- Using React Dev Tools

### Class based Components

- What and Why?
- Adding first class based component
- Working with states and events
- Component Life cycle and in action
- Class based vs functional component

### HTTP

- Introduction to HTTP
- Methods in HTTP
- Response code in HTTP
- Introduction to REST interfaces
- Characteristics of REST
- Introduction to JSON
- JSON data representation
- GET request
- Using async and await
- Handling Http errors
- useEffects()
- POST request

### Custom React Hooks

- What are custom hooks?
- Creating and using custom hooks
- Custom HTTP hooks



**Forms**

- Introduction to Forms
- Working with user input and submission
- Adding validation
- Working with custom hooks

**Routing and Navigation**

- Introduction to Single Page Applications
- What is Routing?
- Why Routing?
- Installing react router
- Defining and using routes
- Working with links
- Dynamic routes with Params
- Nested routes
- Redirecting the user
- Working with query Param
- Sending and getting data via HTTP

**Authentication**

- What is authentication?
- Why and How?
- Authentication tokens
- Setting up
- Adding signup
- Showing feedback to the user
- Adding User login
- Managing authstate with context
- Using tokens
- Redirecting the user
- Adding logout
- Protecting frontend pages
- Persisting User authentication status
- Adding auto logout

**Deployment**

- Introduction Deployment steps Adding lazy loading Building the code for production
- Getting started with deployment Handling routes and finishing deployment
- 
- 





## Best Shine IT Training Center

*simply easy learning*

WEB STACK ACADEMY

2<sup>nd</sup> Floor,  
Near CUB Bank ,  
Srikandapuram,  
Kumbakonam To  
Karaikkal Main Rd

+91-9003152804  
+91-9944591198  
admin@bestshineacademy.co.in /  
bestshineacademy@gmail.com  
www.bestshinewebsit



**Best Shine IT Training Center**  
*simply easy learning*