

FULL STACK WEB DEVELOPMENT

TRAINING (MERN)



JOIN GRRAS FOR
JOB READY SKILLS

EXCEL AS A FULL-STACK DEVELOPER



From Zero to Job Ready

Master Full-Stack

A Full-Stack developer excels in both Front-end and Back-end, crafting a seamless fusion of design and functionality for a complete web experience.

Guess what? They're in **high demand!** Companies seek these tech masters who can do it all.



Course Curriculum

Full Stack Web Development

HTML

- Introduction
- Empty and Container, Semantic Tags
- Elements & Attributes

- Tables & Lists
- Forms and other imp tags

CSS

- Introduction
- Inline, Internal, External
- Box model and Flex model

- Margin & Padding
- Positioning & Pseudo Classes

BootStrap Framework

- Introduction
- Alerts, Modals, Tooltip & Popover
- Button Groups
- Badges
- Carousel
- Collapse
- Dropdowns
- Inputs
- List Groups
- Navbar

- Progress Bars
- Pagination
- Panels
- Responsive Design
- Typography
- Tables
- Tabs/Pills

JavaScript Basic

- Introduction
- Variables
- Operators

- Arithmetic
- Assignment
- Data Types

- Functions
- Objects
- Events
- Strings
- Numbers
- Arrays
- Date Methods
- Math Methods
- Random
- Booleans
- Control statements
- Type Conversion
- Bitwise
- RegExp
- Errors
- Scope
- Strict Mode
- This Keyword
- Let
- Const
- Debugging
- JSON

JavaScript Advance

- Closures
- Coercion
- Constructor
- Call Backs & CallBack Hell
- Call stack, Call queue, Event loop
- Task queue & Microtask
- Hoisting
- IIFE(Immediately Invoked Function Expression)
- Observable & Implementation with RxJs
- Prototype and its methods
- Promise & Async / Await in JS
“this” concepts

React JS Content

- Introduction to Basic building blocks of Web Application Development.
HTML-CSS-JS
- Single Page Apps VS Multi-Page Applications.
- What is a DOM, VirtualDOM and ShadowDOM?
- Installing ReactJS and writing our first Hello World code.
- The Folder Structure of React Application.
- Basic Building Blocks of React JS – (components,state,props,jsx)
- What are the components? Understanding the component basics and different types of components.
- What are the functional components and class components?
- Nested components.
- What are props?
- Passing Props in components.
- Creating reusable & dynamic components



UI and Components Styling In React

- Tailwind CSS , Responsive Design & its uses
- Styling React components using material UI
- Creating a navigation-bar in ReactJS V5
- Introduction to React Router
- Creating Routes and Nested Routes



React Architecture

- Components with state, props and children
- Using state and props, and their differences
- Manipulating the state with `setState()` method
- Difference between stateless and stateful components
- Adding two-way binding
- Conditioning Rendering
- Making API REQUESTS with Axios vs Fetch
- Dynamic rendering from API
- The purpose of keys in lists and Implementing keys in lists
- Handling User Input with forms and events
- Controlled Elements VS Uncontrolled elements
- Communicating child to parent
- Handling user Input of dynamic forms & (MAKING API CALL)
- Custom Forms & Validation
- Showing validation errors and other error messages
- Handling overall form validity
- Using Refs for DOM Access



Redux and Advance Concepts

- Redux, and Understanding
- Adding Redux to the React Project and Redux Devtools
- What is the provider and adding the provider?
- Setting Up the Reducer and the Store
- Dispatching the Actions
Adding Subscriptions
- Connecting React to Redux
- Connecting the store to react
- Dispatching the Actions from within the Component
- Passing and Retrieving Data with Actions
- Switch Case in the reducer
- Updating the State and the Array Immutable
- Combining multiple Reducers
- Handling Authentication in React

- The Context System with React.
- Replacing Redux with Context or using both together.
- Higher-Order Components in React.

Hooks

- Introducing Lifecycle methods.
- Why use Lifecycle methods
- React Hooks – An Introduction,
- Why React Hooks
- Moving from classes to functional components
- Introducing the first Hook: useState
- Hook and Second Hook: useEffect Hook
- Data Fetching with hooks/replacing the class lifecycle methods

- The concept of Render Props.
- Error Boundaries.
- React Fragments

- Fetching data on component mount with useEffect
- Using the useRef Hook
- Displaying loading state with useState
- Error Handling and displaying errors with useState
- The useReducer Hook
- Using the useContext Hook

Getting started with Node

- What is Node and its Architecture
- Writing your first code in Node

Node Module System

- Introduction
- Global Object
- Modules and its methods
- Path Module
- OS Module
- File System Module
- Events Module
- Event Arguments
- Extending EventEmitter
- HTTP Module

Package Manager (NPM/YARN)

- Introduction
- Package.json
- Installing a Node Package
- Using a Package
- Package Dependencies
- Updating Local Packages
- DevDependencies
- Uninstalling a Package
- Nodemon



Building Restful API Using Express

- Introduction
- RESTful Services
- Introducing Express
- Building Your First Web Server
- Environment Variables
- Route Parameters
- Handling HTTP GET Requests
- Handling HTTP POST Requests

- Calling Endpoints Using Postman
- Input Validation
- Handling HTTP PUT Requests
- Handling HTTP Delete Requests
- Project- Build the Genres API
- Building RESTful APIs with Express Recat

MongoDB

CRUD Operations Using MongoDB

- Introducing
- Installing on Windows / Mac and its collection
- Schemas & models
- Saving a Document
- Querying Documents

- Comparison Query Operators
- Logical Query Operators
- Regular Expressions
- Counting
- Updating a Document
- Removing Documents

Mongo Data Validation

- Validation
- Built-in
- Custom

- Async
- Validation Errors
- SchemaType Options

Mongoose

Modeling Relationships between Connected Data

- Modelling Relationships
- Referencing Documents
- Population
- Embedding Documents

- Using an Array of Sub-documents
- Transactions
- ObjectId & Validation
- A Better Implementation



Authentication and Authorization

- Introduction
- Creating the User Model

- Registering Users
- Hashing Passwords

- Authenticating Users
- Testing the Authentication
- JSON Web Tokens
- Generating Authentication Tokens
- Storing Secrets in Environment
- Variables Setting Response Headers
- Getting the Current User
- Logging out Users
- Role-based Authorization
- Testing the Authorization
- Authentication and Authorization Recap



Handling and Logging Errors

- Introduction
- Handling Rejected Promises
- Express Error Middleware
- Try Catch Blocks
- Express Async Errors
- Logging Errors
- Logging to MongoDB
- Uncaught Exceptions
- Unhandled Promise Rejections



Deployment

- Introduction
- Heroku intro
- Account creation on Heroku
- Preparing the App for Production
- Adding the Code to a Git Repository
- Deploying to Server
- Testing the app



Cloud Computing

- Introduction of AWS
- AWS Elastic Beanstalk (Run and manage web apps)
- Amazon RDS (Managed relational database service for MySQL, PostgreSQL, Oracle, SQL Server, and MariaDB)
- Amazon DynamoDB (Managed NoSQL database)
- Amazon Simple Storage Service (S3) (Scalable storage in the cloud)
- Heroku Cloud Platform

WHO IS THIS COURSE FOR



Recent Graduates
and Professionals.



3rd and 4th Year College
Students are encouraged
to join.



No Prerequisite
required.



Course Duration

16+ Weeks | 4 months



Sessions

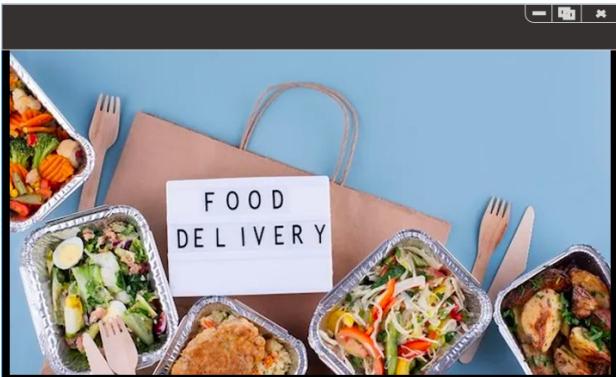
90+ live tech Session
10 Value Added Sessions



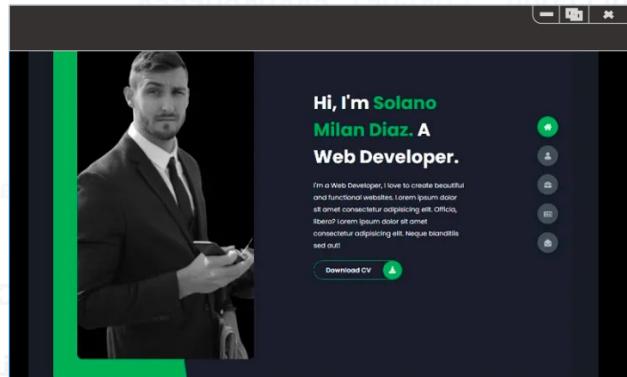
Class Format

2 Hours per day
Online/Offline.

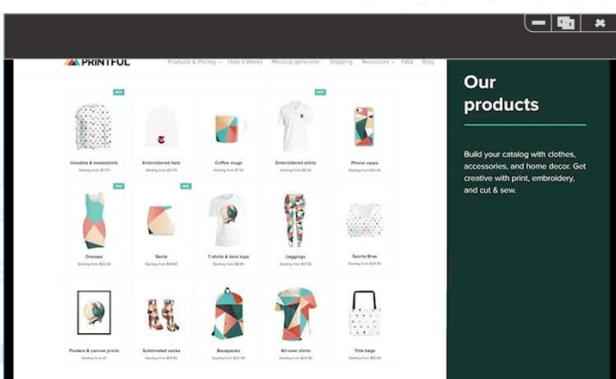
HANDS-ON INDUSTRY EXPERIENCE



FOOD DELIVERY APP



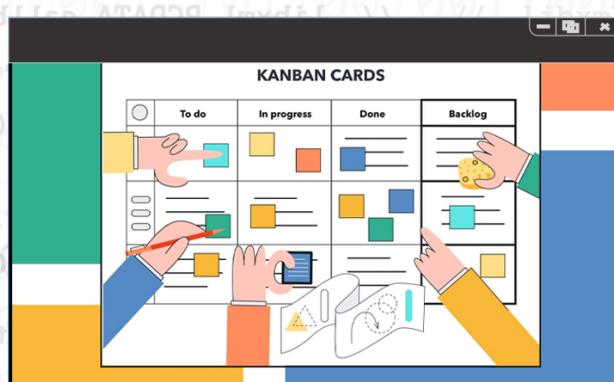
PORTFOLIO



ECOMMERCE



WEATHER APPLICATION



KANBAN BOARD

And Explore more Impactful Projects with Us.....



WWW.GRRAS.COM

YOUR PATH TO SUCCESS



Personalised Mentorship

Individualised guidance from seasoned mentors invested in your success.



24/7 Community Feedback

Engage with a dynamic community for continuous learning and support.



Hands-On Activities

Practical, skill-enhancing activities that reinforce theoretical knowledge.



Real-Life Problem Solving

Tackle authentic challenges, refining your problem-solving acumen.



Job Assistance

Dedicated support to facilitate a seamless transition into your career.

TRIUMPHS UNVEILED

INSPIRING SUCCESS STORIES



Praveen Singh

Hello Everyone. My Name Is Praveen Singh. Grras Solutions is the best place to learn the Full Stack Development training course. If you are fresher and want to make career in Full-Stack developer. the best place in GRASS Solutions in Jaipur. where you definitely gained your skills and easily get job in Your domain. also they provide mock interviews, test series and more activities to achieve your goals.



Sakxi Chaudhary

Hello everyone, Grras Solution is one of the best IT Training Institute throughout the world. If anyone would like to join IT training course I must prefer one and only Grras Solution. I had joined Full Stack development course and I got more knowledge from here. Staff nature is very helpful and supportive. The part is that you can ask any doubts without nervousness.

Thankful to Grras Solution Pvt. Ltd.



Manish Adtani

This is one of the best institute where you can learn any technology literally from scratch. I personally enrolled in Mern Stack course and classes started from the very basic of programming. The labs are upto the mark. Timings of the classes are also suitable according to our schedule of regular classes. You are given assignments and labs for the practice purpose and all the assignments are regularly checked by respective instructor.

EMPOWER YOUR TOMORROW WITH GRRAS



219, Himmat Nagar, Behind Kiran Sweets, Gopalpura Turn, Tonk Road, Jaipur, Rajasthan - 302018



9001997178

trainings@grras.com

www.grras.com

grrasofficial