

Proven working experience in web programming

1. Proven working experience in web development means having practical hand-on experience in developing website and web application it implies that person has successfully worked on real-world projects their ability to code, debug and deploy web-based solution.

2. Keys aspects of proven Experience in web Programming

- **Hands-on Development:** Experience in Writing code for frontend (HTML, CSS, JavaScript, React.js) and backend (Node.js, Express.js, Mongo DB, etc).
- **Project implementation:** Successfully completing web projects, either for company, freelance Clients or personal work.
- **Programming Solving Skills:** Debugging issues, optimizing performance and improving security in web application.
- **Familiarity with frameworks and tools:** Working knowledge of web technologies, databases, APIs, and version control system (like git).
- **Dev Tools & Build Tools:**
 - **Git & GitHub** – Version control
 - **Postman / Thunder Client** – API testing
 - **Vercel / Netlify** – Deployment for frontend apps
- **Deployment and Maintenance:** Deploying applications on Platforms like Vercel, Netlify, or AWS and ensuring they run smoothly.
- **Project implementation:** Project implementation means building and launching a project step by step. It's about taking an idea and turning it into a real, working application.

◆ Key phases of project implementation

- **Planning and Preparation:** Define project Scope, set goal, allocate resources, and prepare timelines.
- **Execution:** Develop the product, integrate technologies and write code if it's a Software project.
- **Monitoring and Testing:** Track progress, fixed bug, and ensure the project meets requirements.
- **Deployment:** Launch the project to end-user or a production environment.

◆ Steps in Deployment

- **Frontend:** vercel, Netlify, GitHub Pages.
- **Backend:** Render, railway, Aws, Digital Ocean.
- **Website Layout** → The structure of a webpage (header, navigation, content, footer).
- **User Interface (UI)** → The visual elements that users interact with (buttons, forms, menus).
- **Standard HTML/CSS Practices** → Writing **clean, semantic, and responsive** code using modern

◆ Types of Layouts in Web Design

- **Fixed Layout** → Uses a specific width (e.g., 1200px), does not adjust on different screens.
- **Fluid Layout** → Uses percentages (%) so elements **resize dynamically**.
- **Responsive Layout** → Adapts to different devices using **CSS Media Queries**.
- **Grid Layout** → Uses **CSS Grid or Flexbox** for structured design.
- **Single-Page Layout** → Everything is on **one page** (e.g., scrolling websites).

◆ What is Bootstrap?

- Bootstrap is a popular open-source CSS framework used for designing responsive and mobile-first websites quickly. It provides pre-built CSS styles, JavaScript components, and a responsive grid system to help developers create visually appealing web pages without writing extensive custom CSS.

◆ Why Use Bootstrap?

- Speeds up development.
 - Makes websites responsive automatically.
 - Reduces the need for writing custom CSS.
 - Provides consistent design across different browsers.
-
- .NET web development, Bootstrap is commonly used with **ASP.NET Core MVC** & **Blazor** to create stylish and responsive UIs.