

How I Increased My Productivity by 50% as a Web Developer

Introduction



Based on five years of experience in web development mainly with React, JavaScript, etc. Productivity is an important factor that differentiates an experienced developer from a beginner. Higher productivity offers many advantages such as meeting your deadlines much faster, lesser stress and reduced number of potential bugs.

1) Learning a Web development framework

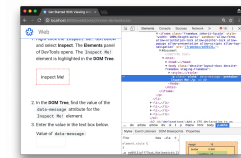
The aim of frameworks is to provide a common structure so that developers don't have to redo it from scratch and can reuse the code provided. In this way, frameworks allow us to cut out much of the work and save a lot of time.



Currently, there are several JavaScript-based web frameworks such as React JS, Angular, Vue JS, etc. React JS is the most popular and widely used among them.

Get started with React JS: [React Roadmap](#), [Learning React JS from scratch in 2022](#)

2) Inspect Element For CSS Change



Designing the webpage and UI components with CSS styles can consume a lot of time. You will have to refresh the Webpage regularly to test every newly added styles. Additionally in some cases the stylesheets may not be applied or override.

The inspect element feature from chrome assists you with debugging CSS stylesheets. Further, new CSS code can be tested without reloading the page.

Resource: [How to Use Inspect tool](#)

Video Tutorial: [Chrome DevTools - Crash Course](#)

3) Keyboard Shortcuts Come in Handy



Learning Keyboard shortcuts for your text editor can be more effective than you might think. The shortcut keys not only save time but also reduces stress on fingers while switching back and forth between mouse and keyboard.

Resource: [useful VS Code shortcuts](#)

4) Design First and Code Later

Many times after writing code we may not achieve the desired end-product or that codebase might become too complicated.

These issues can be minimized by spending a good chunk of time for designing the structure and approach for the solution before developing the solution.

5) Creating Reusable Components



A reusable component is a piece of UI that can be used in various parts of an application to build more than one UI instance. Reusable components reduce effort with less lines of code which results in fewer potential bugs.

By introducing concepts such as higher-order functions and higher order component, reusability can be further increased.

Resource: [Building reusable UI components with React Hooks](#)

6) Improving my Typing Skill

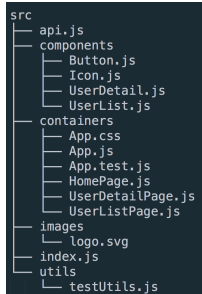


The faster typists developers compared to lesser typing skills spend less time typing in characters which means more time doing the many other things that good software developers do: scripting routine tasks, writing and running tests, documenting for users and maintainers, etc.

Resource: [Improve your typing skills online](#)

7) Better Folder Structure

Many developers work in a large team with a project which could span for years. Hence, this could lead to creating duplicate components and the developer might spend a lot of time understanding code written by someone else.



Maintaining a good folder structure requires less effort and reduces issues associated with readability and duplicate components.

Resource: [React JS folder structure](#)

