WEB TECHNOLOGIES

Who can develop websites and web applications?
If you understand the basic logics of programming (loops, classes, objects, functions etc.), you can easily construct a web application as huge as Google Chrome. Nothing Fancy.

If you like spending your time with a code editor rather than doing other mediocre chores, then yes, you can be the next Zuckerburg.

Coding during web development is relatively easy than that in software designing but it still requires logic and a focused mind-set.

Who cannot develop websites and web applications?
If the only motivation behind all this work is academics and a decent résumé then you might face 'technical' problems as developing on web domains requires a lot of smart-work and dedication. If you don't plan through, you might end up in nowhere with your interest and efforts completely ruined.

Why Development?
The trade of IT and Computer Science (CSE) is basically categorized into three sub-domains; Software engineering, Networking and website development. However former two are also very interesting and fruitful as web development, students are consistently made to believe that those are more important and significant than WebDev.

I can't stress this point enough, Web Development and Designing is third wheel in computer technologies and it can't be under-estimated.

I've personally seen people wandering in software engineering for jobs and careers regardless of their non-interests in field and after failing that, commencing website development.

Web Development is the future. Have a look around and you'll agree.

Myths of Web Development

- WebDev is limited to creating websites.
- WebDev cannot get you 'high-paid' jobs.
- WebDev is easy.

No Dear! Web Development is not limited to websites, you can create browsers, plugins, games, e-commerce and SOFTWARES too. The development scope is too vast, describing it would take more than one article. WebDev can get you jobs in high-reputed companies like Facebook, Microsoft, Google etc. or you could always work as a high-profiled freelancer. Although beginning career might disappoint you, but as you keep growing, so does your pay cheque.

Web Development, although fun, requires hard work especially during PHP/ASP.net phase. You have to work in a specific order if you want to become a good web developer.

Basic Roadmap

Step 1 – HTML5
Though easy it seems, HTML5 is certainly not the old HTML as we know it. HTML5 is the extended functional form of HTML4 with many more features than the former version. With HTML5, you can not only prepare basic webpage structure easily, but can store variables on page itself, create games on it, don't need to rely on flash anymore to run your videos and animation and with a nice 4 months in-depth study, can even create full functioning blog, without even touching server side languages like PHP.

Time Required: 4 months*

Avoid Head-start and HTML5 for dummies book as they are not much practical on subject.

Step 2 – CSS
Who adds colours to rainbows? CSS surely does.

With the application of CSS, you can define your web in a colourful and smooth way. Like HTML, CSS has grew into CSS3, with which, you can also add transitions to pages, scrolls and even mouse hovers. You can even make your site elegant or flashy by using just few lines of CSS3.

Time Required: 1 Month practice with HTML5*

Step 3 – HTML5 & CSS
Now as you learned HTML5 and CSS3 so far, designing pages won't look much hard to you. You can either design your own pages or templates or can work into responsive layouts that can adapt any device's screen.

You can practice as much as you want, but 60-hours* are just enough to practice using HTML5 and CSS3 together. If you're a brilliant student, you can also look up bootstrap3 for responsive layout, before moving to step 4.

Step 4 – HTML5 & JavaScript
Now, after learning how your site should look, you can actually make those buttons useful in some tasks. JavaScript, the best work-around coding language for webpages is versatile, flexible and platform independent. I prefer it over any other web-programming language (client-side). With javascripts, real development starts. You create variables, assign them some values, and pass them as arguments in some function to finally get returned some values or actions (e.g. page loading, redirecting). JavaScript is easy to learn but hard to master, so I recommend moving to next step as soon as you learn the working of functions and variables. JavaScript has many children like node.js and it requires a lot of reading and practice to actually make some use in creating complex web applications. (See Macros).

Step 5 – HTML5 & PHP
So far, it's all about developing a website from front-end. How it looks and how should it react, but real work takes place behind the curtains. PHP is a server-side language, and it handles all the real work and completely relies on your internet connection (unless you're working on local host which I don't recommend for projects).

- *First Half of these learning procedure is working on front-end*
- *Second Half requires more back-coding*

Time required: 4 Months (500 hours at least)*

Step 6 – All above + MySQL
Easiest to work, hardest to maintain. Designing on database can be a real pain if you're not organized and well-planned.

The reason I recommend PHP over any other server-scripting languages is that PHP is great for beginners and if your basic old-school C++ concepts are clear, PHP becomes much easier than you anticipate.

MySQL, with the combination of PHP on Apache server (better than IIS) provides a perfect frame to build webpages and if you want to practice on localhost, try WAMP server.

Practice on local host, implement on remote host.
Books Recommended : Head-first MySQL by O'reilly

Step 7- Python, *Ruby* on rails, Perl etc.
If you followed the steps above in sequence as mentioned, you probably know already the working of a social networking website and how notifications and personal messages (chats) works.

You can create an e-commerce easily, without even moving further to next step. But what if you want to create websites like FlipKart, Amazon, Microsoft, Google or Facebook?

Those are more complicated than just HTML5, PHP and MySQL.

There are other server and client side coding languages to help you do wonders you always wished. Python is one of the most practical, efficient and profitable coding language that is being used today and the best thing about learning python is **it's very easy**.
Although, Python is independent of PHP but I still recommend learning PHP first as it will prepare your mind-set and you will learn python more quickly.

I worked on PHP for six years, and learned python in 12 hours.
There are other coding languages and framework on which you can work on like Ruby on rails. But Python should be your first preference.

Step 8 – Mix-it-up
The final step. Mix-it-up.

Use HTML5 with python, load python codes in HTML5 browser, implement PHP codes on python or simply write a python application to work up some JavaScript.

Mobile internet usage is growing at a steady pace, so much so that it is expected to overtake desktop browsing as early as 2016. It means that it has become more than necessary for website owners to think about channelizing their work into designing mobile friendly, handheld device compliant websites.

What is a Responsive Website?
A responsive website, simply improves the viewing experience; it fits into any device irrespective of resolution. It means a responsive website virtually fits into any device that uses a web browser. Other than being compatible with a unique variety of resolutions, a responsive website works flawlessly across a range of devices, including smartphones, tablets and smartphones.

Why you need to consider a Responsive Website?

No more abandoned checkout at online stores and cluttered viewing experience. With a website fast and responsive you can load it effortlessly. Smartphones can be a great device for viewing websites but when it comes to legibly search out data or filling out information on a website, even smartphones require the loading of mobile friendly websites. With a responsive website, you can ensure easier and effortless browsing

Have a Unique Type of Specific Content

A responsive website quite often uncomplicates the entire task of bowing. With a website that is responsive you can display specific content. For example; if you are organizing information on your website using a display ad, it would pop up differently on different devices, somewhere with uneven aspect ratio. With a responsive website built to work, you can use thumbnails and specific points of contacts to evenly manage ads

Single Format of App

If a website is built compatible for devices, you require not to invest individually to have applications. It means you can keep your cost down, while allowing your website to have a unique application on different variety of devices. So, no need to develop a unique app for different app source

Different other benefits of having a responsive website

SEO benefits – Instead of having an SEO campaign for sites where your website hosted, with responsive website you can just need to do only SEO for your source.

- A responsive website is often the answer to a modern website that is neatly designed
- A responsive website needs you to publish content only once while you require not to write the content again for a different source.

These are some of the vital benefits of having a responsive website. If you require an articulated website, with unique loading attribute; that never works off the mark; rather put your brand to your audience in an improved manner; you can think about utilizing a responsive website. Having a responsive website depends on how you use it.

Mobile internet usage is growing at a steady pace, so much so that it is expected to overtake desktop browsing as early as 2016. It means that it has become more than necessary for website owners to think about channelizing their work into designing mobile friendly, handheld device compliant websites.

What is a Responsive Website?

A responsive website, simply improves the viewing experience; it fits into any device irrespective of resolution. It means a responsive website virtually fits into any device that uses a web browser. Other than being compatible with a unique variety of resolutions, a responsive website works flawlessly across a range of devices, including smartphones, tablets and smartphones.

Why you need to consider a Responsive Website?

No more abandoned checkout at online stores and cluttered viewing experience. With a website fast and responsive you can load it effortlessly. Smartphones can be a great device for viewing websites but when it comes to legibly search out data or filling out information on a website, even smartphones require the loading of mobile friendly websites. With a responsive website, you can ensure easier and effortless browsing

Have a Unique Type of Specific Content

A responsive website quite often uncomplicates the entire task of bowing. With a website that is responsive you can display specific content. For example; if you are organizing information on your website using a display ad, it would pop up differently on different devices, somewhere with uneven aspect ratio. With a responsive website built to work, you can use thumbnails and specific points of contacts to evenly manage ads

Single Format of App
If a website is built compatible for devices, you require not to invest individually to have applications. It means you can keep your cost down, while allowing your website to have a unique application on different variety of devices. So, no need to develop a unique app for different app source

Different other benefits of having a responsive website
SEO benefits – Instead of having an SEO campaign for sites where your website hosted, with responsive website you can just need to do only SEO for your source.

- A responsive website is often the answer to a modern website that is neatly designed
- A responsive website needs you to publish content only once while you require not to write the content again for a different source.

These are some of the vital benefits of having a responsive website. If you require an articulated website, with unique loading attribute; that never works off the mark; rather put your brand to your audience in an improved manner; you can think about utilizing a responsive website. Having a responsive website depends on how you use it.

Introduction

In order to make websites look and function a certain way, web developers utilize different languages. The three core languages that make up the World Wide Web are HTML, CSS, and JavaScript.

In the IT world, the internet is an essential platform, whether its for developing or for consumer use. When developing a website, typically three main languages come into play. These languages are JavaScript, CSS, and HTML. HTML is the backbone of most webpages. Essentially, it is used to create the structure of how a specific website would look like, from the headings, to the paragraphs, the body, links, and even images.

Markup Languages

Markup languages are the languages in which the web is written. The most common markup language used is HTML, which uses tags to annotate text so that a computer can then manipulate the text. Most markup languages are human readable, and use annotations that are distinguishable from the annotated text. There are many different kinds of markups and languages, but all are consistent in the way in which they annotate documents.

Hypertext

Hypertext is defined as the arrangement of information inside a database that allows the user to receive information and to navigate from one document to another by clicking on highlighted words or pictures inside the primary document. Hypertext is

the base of the World Wide Web, because it enables user to click on other links to get more information. Hypertext is a term used for all links, whether it appears as texts or other graphical part.

Hypertext Markup Language (HTML)

HTML is the conventional markup language used to create and edit web pages and web applications. HTML is used for creating the basic structure of a website. HTML consists of different elements preceded by an opening tag, <tag>, and a closing tag, </tag>. The content between the tags, <html> and </html>, is the content of the webpage. The content between the tags, <head> and </head>, is the title of the webpage. This text is displayed between the <title> and </title> tags. The content between the tags, <body> and </body>, is the main content of the webpage. The content can include links , paragraphs, headings, and various other elements.

Here are the most commonly used HTML tags:

Tag	Description
<h1> - <h6>	Gives a web page a heading. 1 is the largest heading you can have and 6 is the smallest.
<p>	Starts a paragraph in your web page.
<i>	Italic font style.
<b>	Bold font style.
<a>	Inserts hyperlinks onto a web page.
&	Starts an unordered or ordered list.
<!DOCTYPE>	Defines the document type of the web page.
<!-- -->	Allows you to insert comments into your HTML code. Comments aren't displayed in on the web page, but are helpful for organization.

<img>	Inserts an image onto a web page.
 	Inserts a line break between bodies of text.

Web Design Programs

Web Design Programs help the webpage creator manage and create the content of a website. Many Web Design programs have many built in tools that ease the process of creating a website. Such programs are Dreamweaver and Sublime. There are also publishing programs like Wordpress and Ghost that allow the user to have more of a GUI based interface for blogging and managing a website.

Sublime

Sublime is a text editor that allows the web developers, programmers, software engineers, etc. manipulate code. It's not only for HTML and CSS it can be set-up for many different programming languages and new productivity tools. One contribution that Sublime has that many do not is the "Package Control" Tool. The tool gives you full access to an entire library of content to better your coding experience. For example, there is a package you can install called Emmet, helping in typing massive amount of HTML, if you type "html:5" and press "tab" then emmet will push out all the correct syntax for an HTML 5 document.

SASS

SASS is some what like emmet but is more of a language. It is a Ruby engraved language that gives CSS much more capabilities like variables and nesting. Like emmet it makes writing CSS much faster and more efficient saving the programmer lots of time.

Dynamic Web Content

Client-Side Scripting

Generally refers to computer programs on the web that are executed by the user's web browser, instead of on a web server, enabling web pages to be scripted. Client-side scripts do not require additional software on the server but instead utilize the user's web browser to understand the scripting language in which it is written.

Server-Side Scripting

Server-side scripting is a technique used in web development that involves using scripts on a web server which produce a unique response for each user's request to the website.

Combination technologies

When both client side and server side scripting collectively build a webpage it is known as a web application. This web application can manage user interaction, security, and help improve performance between the client and server. Web

applications can include anything from online stores to instant messaging services as long as both server and client sides execute scripts to a achieve a common goal in unison.

JavaScript

JavaScript is a scripting language that is used along with HTML and CSS as the three core components of the World Wide Web. JavaScript has first-class functions and is used in most websites. JavaScript does not have any I/O which means that it has to be embedded in the host environment. JavaScript is also used in PDF documents, game development, and desktop and mobile applications. JavaScript is most commonly used to make DHTML by adding client-side behavior to HTML pages.

Worldwide Web Consortium

Worldwide Web Consortium (W3C) is an international community of web members to meet the Web standards. It was founded by Tim Berners-Lee, an inventor of the Web, back in the 20th century. W3C is designed to reach a full potential of the Web and to make it accessible to all users from all over the world. Also, another aim for W3C was to make standards to maintain the growth of the Web in a single direction rather than splitting into competing groups. Here are some of the standards for W3C:

·Accessibility

·Web Authoring

·Web Performance

·Cascading Style Sheets

·HTML5

·Web Fonts

·Widgets

·Media Access

·Mobile Web Applications

·Internationalization of Web Design and Applications

·Mobile Web Authoring

·XML

·Graphics

·RDF

·HTTP