

**TRAINING REPORT**

On

Web Design for Everybody

(Basics of Web Development and Coding)

*Submitted by*

**Name Surname**

***Registration No:***

**\*\*\*\*\*\*\*\***

**Degree**

**Branch Name**

**College Name, Location**

**Contents**

1. Declaration
2. Acknowledgement
3. Certificates
4. Organization Review
5. Technology Learnt
6. What I Learn
7. Skills Gain
8. Reason for choosing this technology
9. Learning Outcome from training/technology learnt
10. Bibliography

**DECLARATION**

I hereby declare that I have completed my six weeks summer training at **(Web Design for Everybody (Basics of Web Development and Coding)** at **Coursera** from (**Date-From**) to (**Date-To**) under the **Niche Technologies** (Web Development). I have declared that I have worked with full dedication during these six weeks of training and my learning outcomes fulfill the requirements of training for the award of degree of (Degree), College Name.

(Signature of student)

**Name Surname**

**[ Registration No.]**

**ACKNOWLEDGEMENT**

It is a matter of great satisfaction and pleasure to present this report on Working **Web Design for Everybody (Basics of Web Development and Coding**. I take this opportunity to owe my thanks to all those involved in my training. Firstly, I would like to thank **Coursera** for giving the opportunity to complete my project in the organization.

I put on record my sincere thanks to my college, College Name, Location, for giving me such an opportunity. I am extremely grateful to my Professors for the encouragement, discussions and critical assessment of the project.

It was a good experience for me to Learn at Coursera, a pioneer in the Massive Open Online Coursers industry.

I am greatly obliged to Everyone who have shared their expertise and knowledge with me without which the completion of project would not have been possible.

I would like to express my sincere thanks to my institutional mentor Mr. Faculty Name (Professor Designation) for her valuable time and guidance.

I express my gratitude towards staff of Coursera, those who have helped me directly or indirectly in completing the training.

**Name**

**Registration No.**

**ORGANIZATION OVERVIEW**

**Coursera** was founded in 2012 by Stanford University computer science professors Andrew Ng and Daphne Koller. Ng and Koller were inspired by their experiences offering their Stanford courses online in fall 2011, and soon after left Stanford to launch Coursera. Princeton, Stanford, the University of Michigan and the University of Pennsylvania were the first universities to offer content on the platform. Offerings have since expanded to include Specializations – collections of courses that build skills in a specific subject – as well as degrees and a workforce development product for businesses and government organizations.

**MISSION**

We envision a world where anyone, anywhere can transform their life by accessing the world’s best learning experience.

**COURSES**

Every course on Coursera is taught by top instructors from the world’s best universities and educational institutions. Courses include recorded video lectures, auto-graded and peer-reviewed assignments, and community discussion forums. When you complete a course, you’ll receive a sharable electronic Course Certificate.

**ANDREW NG**

***Co-Chair of the Board and Co-Founder of Coursera***

Andrew Ng co-founded Coursera in 2012, served as the company’s Co-CEO until 2014, and is currently the Co-Chair of the Coursera Board.

He is the former Chief Scientist at Baidu, a Chinese language search engine, where he was responsible for driving the company's global AI strategy and infrastructure. Dr. Ng is also an adjunct professor in the computer science department at Stanford University. In 2011, he led the development of Stanford's Massive Open Online Course platform and taught an online machine learning class that was offered to over 100,000 students - the initiative that led to the co-founding of Coursera.

Previously, Dr. Ng was the founding lead of the Google Brain deep learning project. Dr. Ng has authored or co-authored over 100 research papers in machine learning, robotics, and related fields. In 2013 he was named to the Time 100 list of the most influential persons in the world. He holds degrees from Carnegie Mellon University, MIT, and the University of California, Berkeley.

**About this Specialization**

This Specialization covers how to write syntactically correct HTML5 and CSS3, and how to create interactive web experiences with JavaScript. Mastering this range of technologies will allow you to develop high quality web sites that, work seamlessly on mobile, tablet, and large screen browsers accessible. During the capstone you will develop a professional-quality web portfolio demonstrating your growth as a web developer and your knowledge of accessible web design. This will include your ability to design and implement a responsive site that utilizes tools to create a site that is accessible to a wide audience, including those with visual, audial, physical, and cognitive impairments.

**What I learn**

* Add interactivity to web pages with JavaScript
* Apply responsive design to enable page to be viewed by various devices
* Describe the basics of Cascading Style Sheets (CSS3)
* Use the Document Object Model (DOM) to modify pages

**Skills Gain**

* *Html5*
* *Cascading Style Sheets (CCS)*
* *JavaScript*
* *Responsive Web Design*

**TECHNOLOGIES**







**Technology Learnt**

**Introduction to HTML5**

Thanks to a growing number of software programs, it seems as if anyone can make a webpage. But what if you actually want to understand how the page was created? There are great textbooks and online resources for learning web design, but most of those resources require some background knowledge.

This course is designed to help the novice who wants to gain confidence and knowledge. We will explore the theory (what actually happens when you click on a link on a webpage?), the practical (what do I need to know to make my own page?), and the overlooked (I have a page, what do I do now?).

Throughout the course there will be a strong emphasis on adhering to syntactic standards for validation and semantic standards to promote wide accessibility for users with disabilities. The textbook we use is available online, “The Missing Link: An Introduction to Web Development and Programming” by Michael Mendez from [www.opensuny.org](http://www.opensuny.org).

This course will appeal to a wide variety of people, but specifically those who would like a step-by-step description of the basics. There are no prerequisites for this course and it is assumed that students have no prior programming skills or IT experience. The course will culminate in a small final project that will require the completion of a very simple page with links and images. The focus of this course is on the basics, not appearance. You can see a sample final page at <http://intro-webdesign.com/html5-plain.html>.

This is the first course in the Web Design for Everybody specialization. Subsequent courses focus on the marketable skills of styling the page with CSS3, adding interactivity with JavaScript and enhancing the styling with responsive design.

**Skills you will gain**

* Web Design
* Web Accessibility
* Html
* Html5

**Introduction to CSS3**

The web today is almost unrecognizable from the early days of white pages with lists of blue links. Now, sites are designed with complex layouts, unique fonts, and customized color schemes.

This course will show you the basics of Cascading Style Sheets (CSS3).

The emphasis will be on learning how to write CSS rules, how to test code, and how to establish good programming habits.

When done correctly, the styling of a webpage can take enhance your page. When done incorrectly the result can be worse than no styling at all. To ensure that your sites do not put up barriers for people with cognitive and/or physical disabilities, you will learn how to evaluate pages using the standardized POUR accessibility guidelines.

Upon completion of the course, learners will be able to sketch a design for a given HTML page. Using that design they will use CSS to implement the design by adding fonts, colors, and layouts.

This is the second course in the Web Design For Everybody specialization.

Subsequent courses focus on adding interaction with JavaScript and enhancing the styling with responsive design.

It will be difficult to complete this course if you do not have access to a laptop or desktop computer for the homework.

**Skills you will gain**

* Web Design
* Style Sheets
* Cascading Style Sheets (CCS)
* Web Development

**Interactivity with JavaScript**

If you want to take your website to the next level, the ability to incorporate interactivity is a must.

But adding some of these types of capabilities requires a stronger programming language than HTML5 or CSS3, and JavaScript can provide just what you need.

With just a basic understanding of the language, you can create a page that will react to common events such as page loads, mouse clicks & movements, and even keyboard input.

This course will introduce you to the basics of the JavaScript language.

We will cover concepts such as variables, looping, functions, and even a little bit about debugging tools.

You will understand how the Document Object Model (DOM) is used by JavaScript to identify and modify specific parts of your page.

After the course, learners will be able to react to DOM Events and dynamically alter the contents and style of their page.

The class will culminate in a final project - the creation of an interactive HTML5 form that accepts and verifies input.

This is the third course in the Web Design For Everybody specialization.

A basic understanding of HTML and CSS is expected when you enroll in this class.

Additional courses focus on enhancing the styling with responsive design and completing a capstone project.

**Skills you will gain**

* Document Object Model (DOM)
* JavaScript
* Web Development

**Advanced Styling with Responsive Design**

It used to be the case that everyone viewed webpages on about the same size screen.

But with the explosion of the use of smartphones to access the Internet, the landscape of design has completely changed.

People viewing your site will now expect that it will perform regardless of the platform (smartphone, tablet, laptop, or desktop computer).

This ability to respond to any platform is called responsive design.

This course will expand upon the basic knowledge of CSS3 to include topics such as wireframes, fluid design, media queries, and the use of existing styling paradigms such as Bootstrap.

After the course, learners will be able to:

\*\* Explain the mobile-first paradigm and the importance of wireframes in the design phase

\*\* Create sites that behave across a range of platforms

\*\* Utilize existing design frameworks such as Bootstrap

This is the fourth course in the Web Design For Everybody specialization.

A basic understanding of HTML and CSS is expected when you enroll in this class.

Additional courses focus on adding interactivity with the JavaScript Programming Language and completing a capstone project.

**Skills you will gain**

* Bootstrap (Front-End Framework)
* Responsive Web Design
* JavaScript
* Cascading Style Sheets (CCS)

**Web Design for Everybody Capstone**

The capstone will develop a professional-quality web portfolio.

Students will demonstrate the ability to design and implement a responsive site for a minimum of three platforms.

Adherence to validation and accessibility standards will be required.

The evolving student implementations will be reviewed each week by capstone peers and teaching assistants to make sure that the student keeps up with the agenda of the course.

Upon completion of this course students will feel comfortable creating and/or updating existing front-end sites, utilizing existing frameworks, and testing sites for accessibility compliance.

This course is only open to students who have completed the first four courses in the Web Design for Everybody specialization:

* Introduction to HTML5
* Introduction to CSS3
* Interactivity with JavaScript, and
* Advanced Styling with Responsive Design.

**Reason for choosing this technology**

*I will present the information for you in 5 key areas that ultimately show why web development is still the superior and more effective solution for most people’s foray into the digital space.*

**Reason 1: Reaching Your Audience: “Is anyone out there?”**

Websites can target the biggest and broadest audience, unlike the restrictions to a platform that constrain native applications.

Your clients and customers can use the ease of any device, from a computer to mobile phone, as they have the capability to display your content through the web browser.

Native applications require users to access the app store on their devices or download programs – creating one more process or step in having access to your content.

In other words, native applications are bound by targeted visibility, as users must seek out your application to download, rather than simply accessing a URL.

Distribution of your content is more flexible and agile with web development as there are no stringent app store requirements and content restrictions to follow.

Often times, your application will fall through the cracks in the app marketplaces as it is overshadowed by the glitz of major-league developers.

The biggest draw of reach with web development lies in the power of utilizing SEO techniques to target your audience.

Search engines do not have access to the closed environments of applications and thus, native applications take a backseat to the focused SEO marketing and advertising that websites can capitalize on.

**Reason 2: Design: “Looks are important.”**

Presentation of your content and controlling the user experience in exploring your services is crucial in conveying an effective message.

Web development provides a generally universal solution to address this through the use of HTML, CSS, and JavaScript.

Rather than having to worry about cross-platform interfaces and capabilities across different marketplaces, it is arguably more manageable to deal with cross-browser accessibility.

In other words, it is easier to utilize CSS media queries to account for different browser window sizes than having to create and modify your UI each for the Windows Store, iOS app store, and Android marketplace.

Furthermore, web development frameworks, like Bootstrap, make the hassles of browser compatibility easier to digest, as well.

In terms of presenting a uniform design, web development is a lot easier to manage and execute.

**Reason 3: Development: “Looking Under the Hood.”**

As previously mentioned, web development uses the benefits of HTML, CSS, and JavaScript to create effective and captivating websites with relative ease.

Compare that to native applications, in which code must be specifically written for each marketplace, as the language and processes are significantly different. This requires significant resources as additional development skills and hardware must be acquired.

How easy is web development compared to that?

Your website only has to be built once with cross-browser support and it will be useable on every device. There is a lot of ease and speed in building your application and reaching your audience quicker!

One of the limitations of websites is that it cannot take advantage of the device’s hardware, like geolocation and accessing the camera, but there are updates to HTML5 that make this a possibility.

Web development is definitely a smoother path to developing your application.

**Reason 4: Cost: “How much?!”**

Without sacrificing quality, developing a website is definitely far cheaper than creating native applications.

Rather than having to develop for multiple platforms, maintaining one codebase on a website will cost less, as there is one common skillset to utilize and focus on.

Compatibility and performance issues will be limited to the browser capabilities, for the most part, unlike native applications, in which hardware updates can affect application performance and experience.

Ultimately, there is less investment in acquiring and retaining developer skills as a web developer than as a native application developer.

An important side note is the potential for monetization through the two applications.

Revenue from native apps would require in-app purchases, platform-specific ads, or purchase of the app itself. Basically, if people don’t take the time and effort to download your app, it’s hard to see any returns.

Additionally, any revenue created from app purchases is subject to marketplace fees, which can be as high as 30%.

Websites also need to generate traffic to see any money, but it’s a lot more passive – there’s no “entry fee” or initial barrier to access the content.

A simple typing of the URL into any browser allows users to immediately interact with any advertisements or shopping carts on the site.

What is the bottom line?

Development costs, as well as integrating payment options and third-party integrations, are much more flexible and cost-effective with website applications.

**Reason 5: Upkeep: “Don’t Just Set It and Forget It.”**

The development hurdles with native applications was mentioned earlier in the article,

but it is worth expanding upon, as native app support becomes increasingly more of an issue as your business expands and your code becomes more complex and feature-rich.

Multiple codebases means having to support a large number of devices that are continuously being updated and upgraded.

Web applications can suffer from the same headaches of supporting upgrades, but it is far more manageable and less resource-intensive to operate in a website environment.

Major (and minor) updates can be immediately pushed live to the website and you will get instant feedback and interaction with any new desired implementation changes.

On the other hand, native application updates require approval from the app store and additionally require the updates to be downloaded by the user.

***Deployment is a breeze with web development!***

**Learning Outcome from training/technology learnt**



1. **HTML 5**

With HTML 5 a whole new kind of web applications is possible.

You are now able to develop full featured apps that run on any kind of devices - from mobiles to desktops.

1. **Create stuff**

Being a web developer gives you the power to create new cool things. If you can imagine it you can build it (or kind of).

You don't need any kind of material - just your knowledge about web development.

1. **Low Starting Costs**

Starting with web development costs very little.

Assuming you already have a computer and an internet connection, you only need some webspace when you website is ready for launch - and even this can be free.

1. **Share it with the world**

Websites can be reached from all over the world. If you think about, you realize how powerful this is.

You can literally reach every person on this planet.

1. **Job market**

The web is still growing and web developers are a scarce resource that have good chances on the job market.

**6. Mobile Devices**

With mobile device getting more powerful, there are even possibilities for web developers to create cool stuff

1. **Fast Results**

Unlike traditional desktop programming, web development gives you instant results. No more dump "Hello world!" programs anymore.

1. **It's easy.**

Beginning web development is relatively easy –

you only need your web browser and an editor. Both should be installed on your computer already.

Don't get me wrong, it takes a lot to develop great web applications

like e.g. Google Docs, but barrier of entry is very low.

1. **Being creative**

Web development lets you express yourself.

You can try everything you want and share it with the world.

1. **It's Fun**

Most important, web development is a fun experience.

**Most Important aspect of Web Development**

Wireframing is a way to design a website service at the structural level. A wireframe is commonly used to lay out content and functionality on a page which takes into account user needs and user journeys.

Wireframes are used early in the development process to establish the basic structure of a page before visual design and content is added.

#### **Wireframing is essential in UI design**

A wireframe is a layout of a web page that demonstrates what interface elements will exist on key pages. It is a critical part of the interaction design process.

The aim of a wireframe is to provide a visual understanding of a page early in a project to get stakeholder and project team approval before the creative phase gets under way. Wireframes can also be used to create the global and secondary navigation to ensure the terminology and structure used for the site meets user expectations.

#### **Advantages of Wireframing**

One of the great advantages of wireframing is that it provides an early visual that can be used to review with the client. Users can also review it as an early feedback mechanism for prototype usability tests.

Not only are wireframes easier to amend than concept designs, once approved by the client and the users they provide confidence to the designer.

From a practical perspective, the wireframes ensure the page content and functionality are positioned correctly based on user and business needs.

#### **Disadvantages of Wireframing**

As the wireframes do not include any design, or account for technical implications, it is not always easy for the client to grasp the concept. The designer will also have to translate the wireframes into a design, so communication to support the wireframe is often needed to explain why page elements are positioned as they are. Also, when content is added, it might initially be too much to fit within the wireframe layout, so the designer and copywriter will need to work closely to make this fit.

**This is How Wire Frame looks like – Made by me @ wireframe.cc**

DESKTOP VERSION





* **It is quicker and cheaper to review and amend the structure of the key pages in a wireframe format.**
* **Iterating the development of the wireframes to a final version will provide the client and the design team confidence that the page is catering to user needs whilst fulfilling the key business and project objectives.**

**Bibliography**

1. **Coursera**

Learning or training Platform

1. **Wireframe.cc**

Used to sketch wireframe from scratch for free

1. **Medium**

Articles related to web or future web development

1. **Learncodeonline.cc**

Reference material for code and learning course material

1. **Google**

Believe in true power of googling: whether it’s anything from code to images to learning stuff.