Industrial Traning Report

ON

Full Stack, Front-End Web Development

(Topic- Portfolio Webpage)

Submitted By

Manish Gautam

Univ. Roll No. - 181500366

Session: 2018-2022

Department of Computer Engineering & Applications

Institute of Engineering & Technology

GLA University

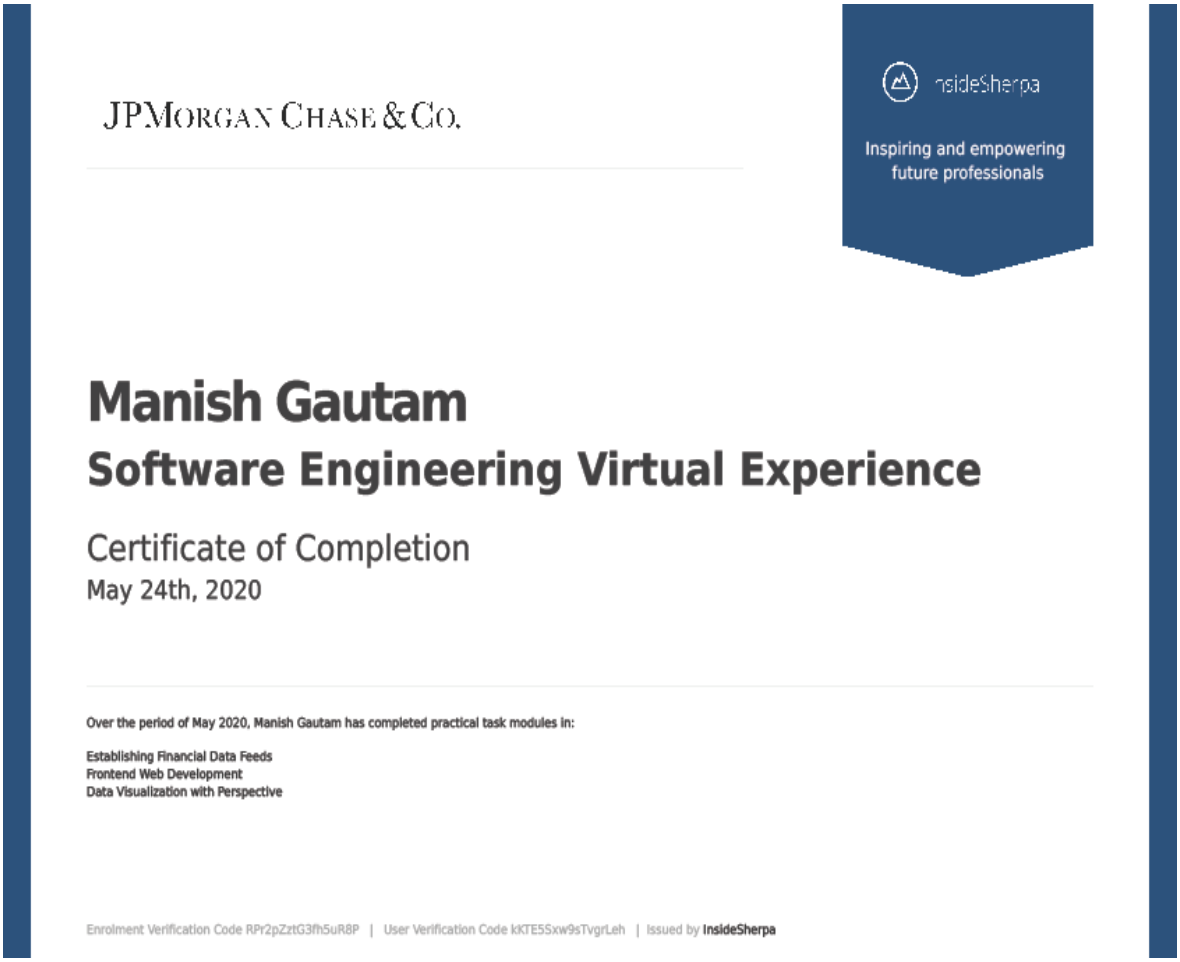
Mathura-281406, India

Declaration

I hereby declare that the work which is being presented in the Industrial Training "**Portfolio Webpage Development**", in partial fulfillment of the requirements for Industrial Training viva voce, is an authentic record of my own work carried under the supervision of "**Inside Sherpa".**

Name of Candidate: Manish Gautam
Univ. Roll. No. :181500366
Course: B.tech
Branch: Computer Science and Engineering (CSE)
Year: 3rd (2020-2021)
Semester: Vth

Certificate



JPMorgan Chase & Co.

insideSherpa

Inspiring and empowering future professionals

Manish Gautam

Software Engineering Virtual Experience

Certificate of Completion

May 24th, 2020

Over the period of May 2020, Manish Gautam has completed practical task modules in:

Establishing Financial Data Feeds
Frontend Web Development
Data Visualization with Perspective

Enrolment Verification Code RPr2pZztG3fh5uR8P | User Verification Code kKTE5Sxw9sTvgrLeh | Issued by InsideSherpa

Summer Training Synopsis

Student Information:

Name: Manish Gautam	University Roll. No.: 181500366
Mobile: +91 8979440857	Email:

Information about Industry/Organization:

Industry/Organization Name with full Address	JP Morgan Chase & Co. Headquarters at New York, New York, United States Online Training
Contact Person	Name & Designation: Manish Gautam Mobile/email: +91 8979440857

Project Information:

Title Of Project/Training/Task	Live Portfolio Full Stack Webpage Development
Role & Responsibility	Learner and Student
Technical Details	Hardware Requirements: Processor- minimum 1 GHz, Recommended 2GHz or more. Network connection - LAN or a wireless adapter (Wi-Fi) Hard Drive – 500 GB or more Memory (RAM): 4 GB or above Processor-Dual-core or above. Software Requirements: Visual Studio code Editor FullStack Languages-HTML, CSS, Javascript etc. HTML,CSS JS extensions.
Training Implementation Details	Fully Implemented
Training Period	Start Date: 10th May,2020 End Date: 24th May,2020 Duration Of Training (In Weeks): 2 weeks

Summary of the Training Work:

About the Course:

I took the Course from LinkedIn Website for Learning Full Stack Web Development Basic concepts
like Html, CSS, JavaScript, etc. and then some advanced concepts like NodeJS and
Express JS.
I also Learned how to make a portfolio website using HTML CSS and JavaScript. It was a very good learning Experience. At Last I Also created a simple project i.e., a website that is used as my Introduction and my portfolio that what I do and what up till now I have done or Implemented via my learning resources from various different platforms.
leaning purpose of English, Hindi, Mathematics for digital learning.
It was built by using all the Concepts that were Taught by the instructors of that course.
The Website Was built Up to a condition of hosting online and Was tested at Local host
Server and is also available online as a Git-Hub live website.

About the Website:

Objective: - The Website was built to Just Have a practical Knowledge of the basic
web development that was learned during this Course And is up-to serve Any
real time life problems to any significant extent. However, the website can used to Introduce myself in a very interactive manner.

Technologies Used: Front End Languages

- HTML
- CSS
- JavaScript

Software Used:

- Visual Studio code Editor
- Full Stack Languages-HTML, CSS, JavaScript etc.
- HTML,CSS JS extensions.

FEATURES PROVIDED:

- Can navigate to any other page.
- Inline links to scroll and understand easily.
- Matched to user view and to maintain responsiveness in order that user can operate at any device like Mobile/PC/Laptop and Tablets.
- External links to redirect to the following page address.

Future Advancements And Scope:

- The hosting by having it with its own domain.
- More additional features to surf through this website.
- GUI can be updated and can be given more styling.
- Additional works and practical implementations can be uploaded.

REQUIREMENTS TO USE THE WEBSITE:

- A simple device – Smartphone/PC/Tablet with a simple browser application.
- A fair Internet connection to get connected to the website.

Acknowledgement

The internship opportunity I had was a great chance for learning. Therefore, I consider myself as a very lucky individual as I was provided with an opportunity to be a part of it.

Bearing in mind previous I am using this opportunity to express my deep gratitude and special thanks to the company **JP Morgan Chase and Co.** who in spite of being a great company busy with its work, took time out to hear, guide and kept many students on the correct path.

I owe my wholehearted thanks to the staff members of **Inside Sherpa**. The valuable information provided by them in their respective fields helped me a lot.

I perceive as this opportunity as a big milestone in my career development. I will strive to use gained skills and knowledge in the best possible way, and I will continue to work on their improvement, in order to attain desired career objectives.

Manish Gautam

B.tech. (3rd Year)

Computer Science and Engineering (CSE)

GLA University, Mathura.

ABSTRACT

WEBSITE DESIGN AND DEVELOPMENT were the main objective of this internship. To develop a web based application or software there are several programming languages that are in use. Some of them are only used for the frontend and backend design of the software. For example HTML3, HTML5, CSS,CSS3,JavaScript, Bootstrap etc. There are also some other programming languages that are used to develop the dynamic functions of the software or application. For example: PHP, Java etc. Nowadays there are also some framework's that use vastly. Frameworks are basically structured programming by using Model, View, and Controller. It is also called as MVC. If we develop web based application that is very useful for us because we can access it from anywhere of the world. It is very helpful for our daily life. That is why I choose subject of my report is "PORTFOLIO WEBPAGE DEVELOPMENT". Working in Inside Sherpa added huge experiences in my upcoming career. Solving real life problems was another key issue. This report takes us through all the details of WEBSITE DESIGN AND DEVELOPMENT knowledge and experience gathered during this internship period.

Table of Content

Introduction

A portfolio website should tell a story to the viewers, and the title to that story should be "What if this person was a website?". No matter where you're at in your career, it's always important to have a good place to show off your work and projects you've collaborated on, what you like to do, and your professional goals.

A portfolio website makes it easy. You can easily customize it to your needs and keep it updated, attracting recruiters and connections alike as you grow your network and establish credibility in your field.

Generally, portfolios are static websites, which can fall under a lot of categories, like one explaining why the person should be hired with all their work till that time showcased, or can be a story like or a timeline like depiction of work experience with projects to showcase, etc.

Portfolio of a person reflect that individual well. Now keeping in mind what I just explained, and the fact that we are talking about a portfolio website for a person who is a full-stack developer, what better way can be there to communicate that fact to the visitors, than making the portfolio itself a full-stack application.

1.1 Motivation:

The motivation behind learning Full Stack technologies is that we can develop both client as well as server software. I usually try to find a project that really makes me work on it.

A project that gives pleasure to myself. I am exploring in this quarantine, then I found an article explaining what Full Stack is and who are Full Stack developers, what is front end and back end.

I have read about it. I was more attracted towards the exploring the frontend. So, I bought a course on Udemy - online courses to explore more in this field.

And that course helped me a lot in understanding these technologies. And the best way to start is to make a website which is much familiar.

Then, I decided to create a portfolio website to showcase the skills that I have acquired. I also wanted to improve my front-end skills, specifically in CSS, so creating a personal website is a great way to do that.

1.2 Objective:

A portfolio website makes it easy. You can easily customize it to your needs and keep it updated, attracting recruiters and connections alike as you grow your network and establish credibility in your field. When someone looks at a portfolio without knowing the person, it's their first impression for that person.

And in many ways, it's actually like meeting that person for the first time.

I have a very good reason to why I think that is. See, when someone meets you for the first time, the immediate activity is that they want to understand you, figure you out the most they can.

And at today's day and age, people have very less time to do that, and so first impression lasts.

Since its very likely that a person visiting your portfolio has not met you in person yet, the immediate activity is exactly the same as what it is like when they meet you in person, that they will try to figure you out, mostly keeping in mind the reason they visited your portfolio.

The easier you make that process for your visitors, the better is the chance they will find you fit for the reason they are visiting your website.

1.3 Technical Aspect:

For the completion of the tasks the following web languages and scripts have been used: HTML for the construction of the layout, CSS for the design of the layout, JavaScript for dynamic functioning and Bootstrap for responsiveness of the website.

Technologies Used

HTML (Hypertext Markup Language):

Hypertext Markup Language (HTML) is the standard markup language for documents designed to be displayed in a web browser. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

HTML is the standard markup language for creating Web pages. HTML elements are the building blocks of HTML pages.With HTML we construct images and other objects such as interactive forms may be embedded into the rendered page.

HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items.

HTML elements are delineated by tags, written using angle brackets:

Tags such as **Error! Filename not specified** are directly introduce content into the page. HTML is a markup language that web browsers use to interpret and compose text, images, and other material into visual or audible web pages.

Default characteristics for every item of HTML markup are defined in the browser, and these characteristics can be altered or enhanced by the web page designer's additional use of CSS.

HTML is the **language in which most websites are written**. HTML is used to create pages and make them functional.

The code used to make them visually appealing is known as CSS and we shall focus on this in a later tutorial. For now, we will focus on **teaching you how to build rather than design**.

The History of HTML:

HTML was first created by Tim Berners-Lee, Robert Cailliau, and others starting in **1989**. It stands for Hyper Text Markup Language.

Hypertext means that the document contains **links that allow the reader to jump to other places** in the document or to another document altogether. The latest version is known as HTML5.

A **Markup Language** is a way that computers speak to each other to control how text is processed and presented. To do this HTML uses two things: tags and **attributes**

What are Tags and Attributes?

Tags and attributes are the basis of HTML.

They work together but perform different functions – it is worth investing 2 minutes in **differentiating the two**.

HTML Tags

Tags are used to **mark up the start of an HTML element** and they are usually enclosed in angle brackets. An example of a tag is: `<h1>`.

Most tags must be opened `<h1>` and closed `</h1>` in order to function.

HTML Attributes

Attributes contain **additional pieces of information**. Attributes take the form of an opening tag and additional info is **placed inside**.

An example of an attribute is:

```
<img src="mydog.jpg" alt="A photo of my dog.">
```

In this instance, the image source (src) and the alt text (alt) are attributes of the `<img>` tag.

HTML Rules To Remember

1. The vast majority of tags must be **opened** (`<tag>`) and **closed** (`</tag>`) with the element information such as a title or text resting between the tags.
2. When using multiple tags, the tags must be **closed in the order in which they were opened**. For example:

```
<strong><em>This is really important!</em></strong>
```

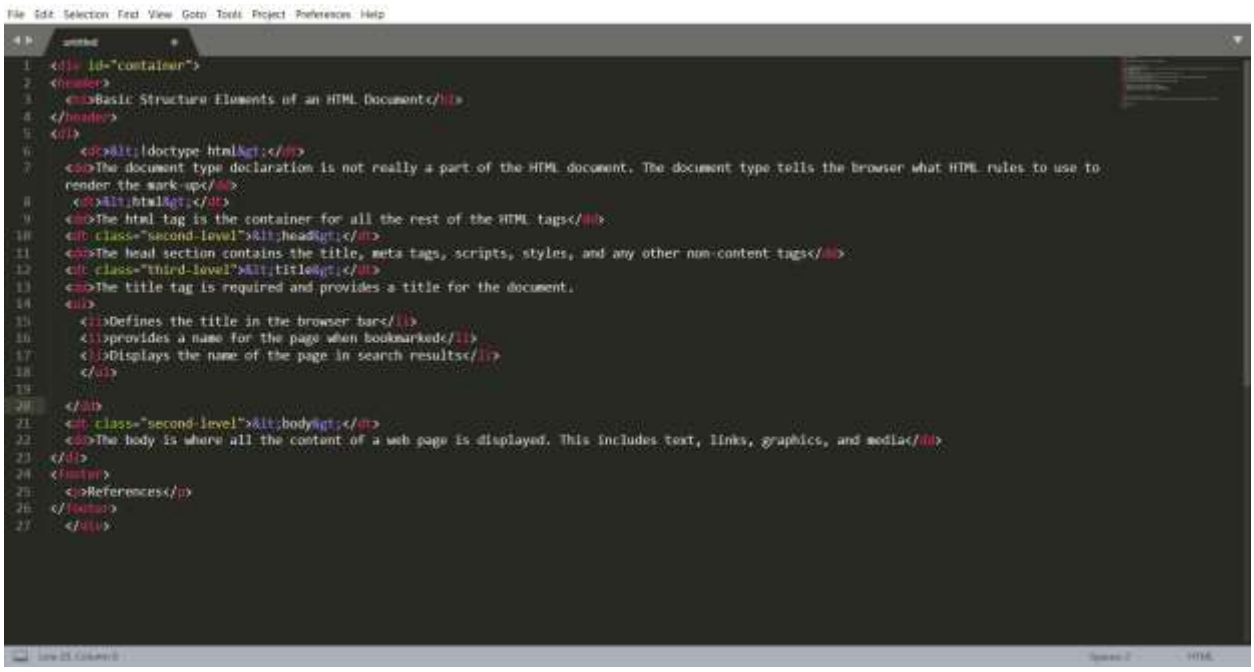
HTML Editors

Now that we've gotten the basic theory out of the way. It's time to **learn how to build our first website**.

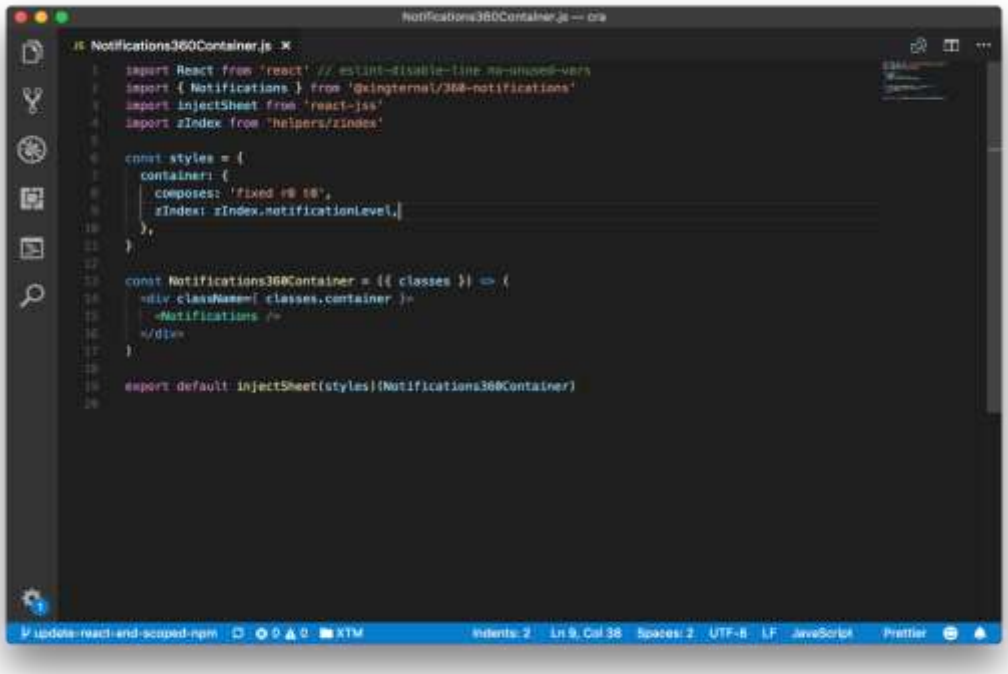
First off, we must ensure that we have the right tools. Most important, we need an HTML editor.

There are many choices on the market. Here are a handful of the most popular:

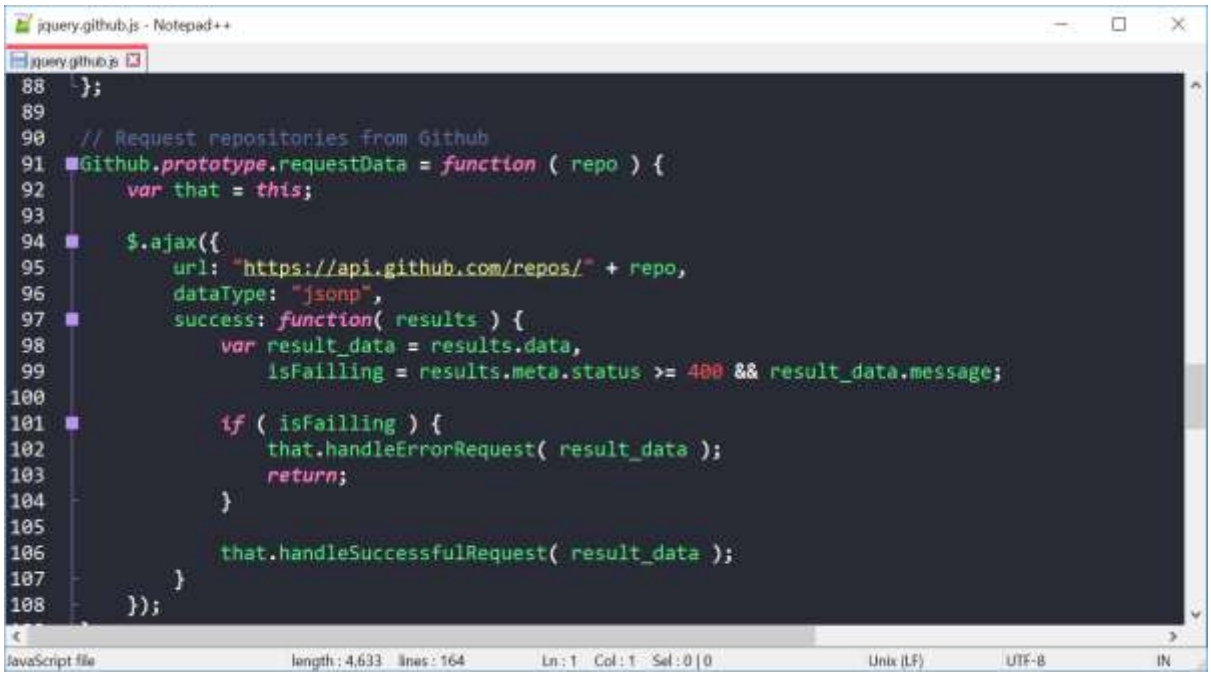
1. Sublime Text 3



2. VS Code Editor



3. Notepad++



CSS (Cascading Style Sheets):

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML.

CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript. It is designed to enable the separation of presentation and content, including layout, colors, and fonts.

This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate CSS file which reduces complexity and repetition in the structural content as well as enabling the ".CSS" file to be cached to improve the page load speed between the pages that share the file and its formatting.

Separation of formatting and content also makes it feasible to present the same markup page in different styles for different rendering methods, such as onscreen, in print, by voice (via speech-based browser or screen reader), and on Braille-based tactile devices.

CSS also has rules for alternate formatting if the content is accessed on a mobile device.

CSS has a simple syntax and uses a number of English keywords to specify the names of various style properties.

A style sheet consists of a list of rules. Each rule or rule-set consists of one or more selectors, and a declaration block.

JAVASCRIPT:

JavaScript is a programming language that conforms to the ECMA Script specification.

JavaScript is high-level, often just-in-time compiled, and multiparadigm. It has curly-bracket syntax, dynamic typing, prototype-based object orientation, and first-class functions.

Alongside HTML and CSS, JavaScript is one of the core technologies of the World Wide Web.

JavaScript enables interactive web pages and is an essential part of web applications.

The vast majority of websites use it for client-side page behavior, and all major web browsers have a dedicated JavaScript engine to execute it.

As a multi-paradigm language, JavaScript supports event-driven, functional, and imperative programming styles.

It has application programming interfaces (APIs) for working with text, dates, regular expressions, standard data structures, and the Document Object Model (DOM).

However, the language itself does not include any input/output (I/O), such as networking, storage, or graphics facilities, as the host environment (usually a web browser) provides those APIs.

JavaScript engines were originally used only in web browsers, but they are now embedded in some servers, usually via Node.js.

They are also embedded in a variety of applications created with frameworks such as Electron and Cordova.

BOOTSTRAP:

Bootstrap is a free and open-source CSS framework directed at responsive, mobile first front-end web development.

It contains CSS and (optionally) JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.

Bootstrap is a web framework that focuses on simplifying the development of informative web pages (as opposed to web apps). The primary purpose of adding it to a web project is to apply Bootstrap's choices of color, size, font and layout to that project.

As such, the primary factor is whether the developers in charge find those choices to their liking. Once added to a project, Bootstrap provides basic style definitions for all HTML elements.

The result is a uniform appearance for prose, tables and form elements across web browsers. In addition, developers can take advantage of CSS classes defined in Bootstrap to further customize the appearance of their contents.

For example, Bootstrap has provisioned for light- and dark colored tables, page headings, more prominent pull quotes, and text with a highlight.

Bootstrap also comes with several JavaScript components in the form of jQuery plugins. They provide additional user interface elements such as dialog boxes, tooltips, and carousels.

Each Bootstrap component consists of an HTML structure, CSS declarations, and in some cases accompanying JavaScript code.

They also extend the functionality of some existing interface elements, including for example an auto-complete function for input fields.

Software Requirement Analysis: Requirements

Hardware Requirements Specification

- Processor: Intel CORE i3 (3th Gen)
- Main Memory (RAM): 256 MB
- Cache Memory: 512 KB
- Monitor: 14inch Color Monitor
- Keyboard: 108 Keys
- Mouse: Optical Mouse
- Hard Disk: 160 GB

Software Requirements Specification

- Full Stack Programming Language: HTML, CSS, JavaScript
- Additional Tools: Visual Studio Code / Sublime text Editor/Notepad++
- Operating System: Windows 7 or higher

Software Design

1. System Design

Design is the first step into the development phase for any engineered product or system. Design is a creative process. A good design is the key to effective system.

The term “design” is defined as “the process of applying various techniques and principles for the purpose of defining a process or a system in sufficient detail to permit its physical realization”.

It may be defined as a process of applying various techniques and principles for the purpose of defining a device, a process or a system in sufficient detail to permit its physical realization.

Software design sits at the technical kernel of the software engineering process and is applied regardless of the development paradigm that is used.

The system design develops the architectural detail required to build a system or product.

As in the case of any systematic approach, this software too has undergone the best possible design phase fine tuning all efficiency, performance and accuracy levels.

The design phase is a transition from a user oriented document to a document to the programmers or database personnel.

System design goes through two phases of development:

- Logical Design
- Physical Design

2. Logical Design

The logical flow of a system and define the boundaries of a system. It includes the following steps:

▪ Reviews the current physical system – its data flows, file content, volumes, frequencies etc.

▪ Prepares output specifications – that is, determines the format, content and frequency of reports.

▪ Prepares input specifications – format, content and most of the input functions.

▪ Prepares edit, security and control specifications. ▪ Specifies the implementation plan.

▪ Prepares a logical design walk through of the information flow, output, input, controls and implementation plan.

▪ Reviews benefits, costs, target dates and system constraints.

3. Physical Design

Physical system produces the working systems by define the design specifications that tell the programmers exactly what the candidate system must do.

It includes the following steps :

▪ Design the physical system.

▪ Specify input and output media.

▪ Design the database and specify backup procedures.

▪ Design physical information flow through the system and a physical design.

▪ Plan system implementation.

▪ Prepare a conversion schedule and target date.

▪ Determine training procedures, courses and timetable.

▪ Devise a test and implementation plan and specify any new hardware/software.

4. Database design

Database Design is a collection of processes that facilitate the designing, development, implementation and maintenance of enterprise data management systems.

Properly designed database is easy to maintain, improves data consistency and are cost effective in terms of disk storage space.

The main objectives of database designing are to produce logical and physical designs models of the proposed database system.

Testing

Introduction

The implementation phase of software development is concerned with translating design specification into source code.

The preliminary goal of implementation is to write source code and internal documentation so that conformance of the code to its specifications can be easily verified, and so that debugging, testing and modifications are eased.

This goal can be achieved by making the source code as clear and straightforward as possible.

Simplicity, clarity and elegance are the hallmark of good programs, obscurity, cleverness, and complexity are indications of inadequate design and misdirected thinking.

Source code clarity is enhanced by structured coding techniques, by good coding style, by, appropriate supporting documents, by good internal comments, and by feature provided in modern programming languages.

The implementation team should be provided with a well-defined set of software requirement, an architectural design specification, and a detailed design description. Each team member must understand the objectives of implementation.

Terms in Testing Fundamental

1. Error:

The term error is used in two ways. It refers to the difference between the actual output of software and the correct output, in this interpretation, error is essential a measure of the difference between actual and ideal. Error is also to use to refer to human action that result in software containing a defect or fault.

2. Fault:

Fault is a condition that causes to fail in performing its required function. A fault is a basic reason for software malfunction and is synonymous with the commonly used term Bug.

3. Failure:

Failure is the inability of a system or component to perform a required function according to its specifications.

A software failure occurs if the behavior of the software is the different from the specified behavior. Failure may be caused due to functional or performance reasons.

- **Unit Testing**

  The term unit testing comprises the sets of tests performed by an individual programmer prior to integration of the unit into a larger system.

  A program unit is usually small enough that the programmer who developed it can test it in great detail, and certainly in greater detail than will be possible when the unit is integrated into an evolving software product.

  In the unit testing the programs are tested separately, independent of each other. Since the check is done at the program level, it is also called program teasing.

- **Module Testing**

  A module and encapsulates related component. So can be tested without other system module.

- **Subsystem Testing**

  Subsystem testing allows for data-driven decisions that drive the product design of components, subsystems, and systems to meet the product requirements.

There are four categories of tests that a programmer will typically perform on a program unit.

I. Functional test
II. Performance test
III. Stress test
IV. Structure test

4. Functional Test:

Functional test cases involve exercising the code with Nominal input values for which expected results are known; as well as boundary values (minimum values, maximum values and values on and just outside the functional boundaries) and special values.

5. Performance Test

Performance testing determines the amount of execution time spent in various parts of the unit, program throughput, response time, and device utilization by the program unit.

A certain amount of avoid expending too much effort on fine-tuning of a program unit that contributes little to the overall performance of the entire system.

Performance testing is most productive at the subsystem and system levels.

6. Stress Test

Stress test are those designed to intentionally break the unit. A great deal can be learned about the strengths and limitations of a program by examining the manner in which a program unit breaks.

7. Structure Test

Structure tests are concerned with exercising the internal logic of a program and traversing particular execution paths.

Some authors refer collectively to functional performance and stress testing as “black box” testing. While structure testing is referred to as “white box” or “glass box” testing.

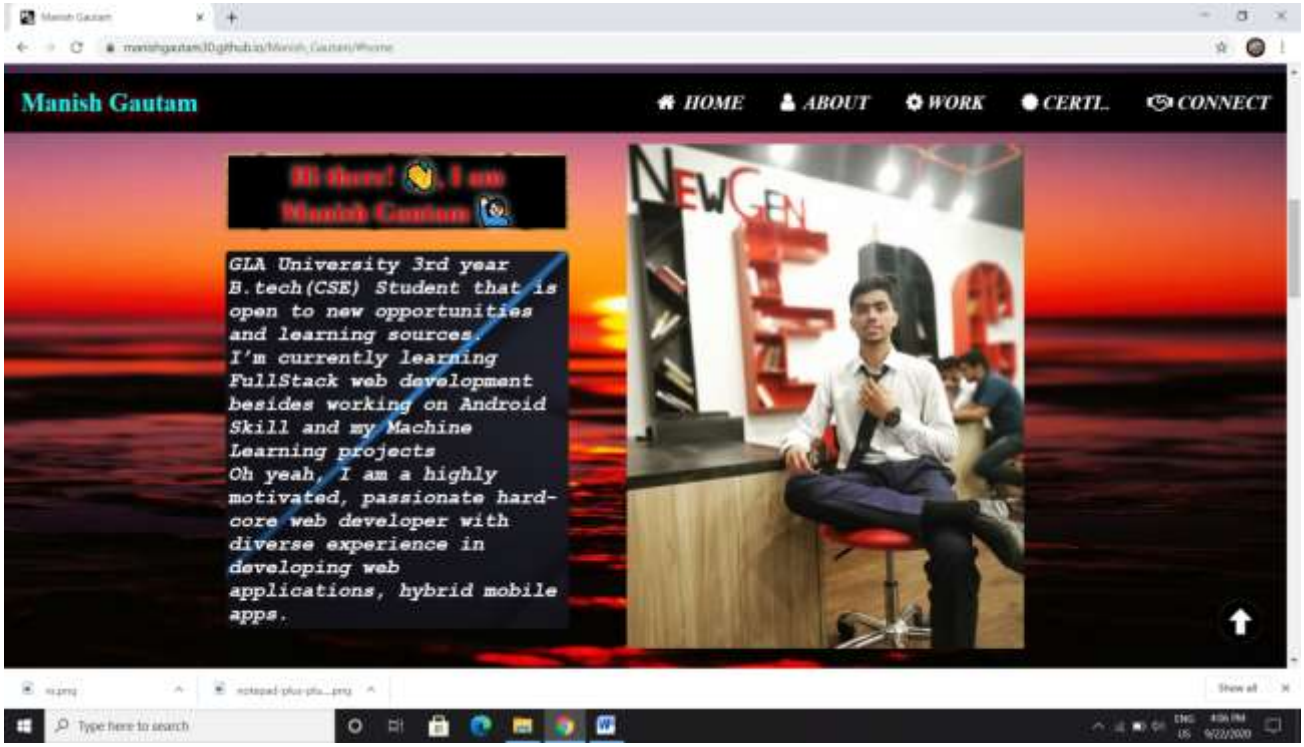
Implementation

In my Project of Portfolio-site, initially I have created a home page where I have inserted a simple and decent background image with a Title heading as “Entrepreneurship”.

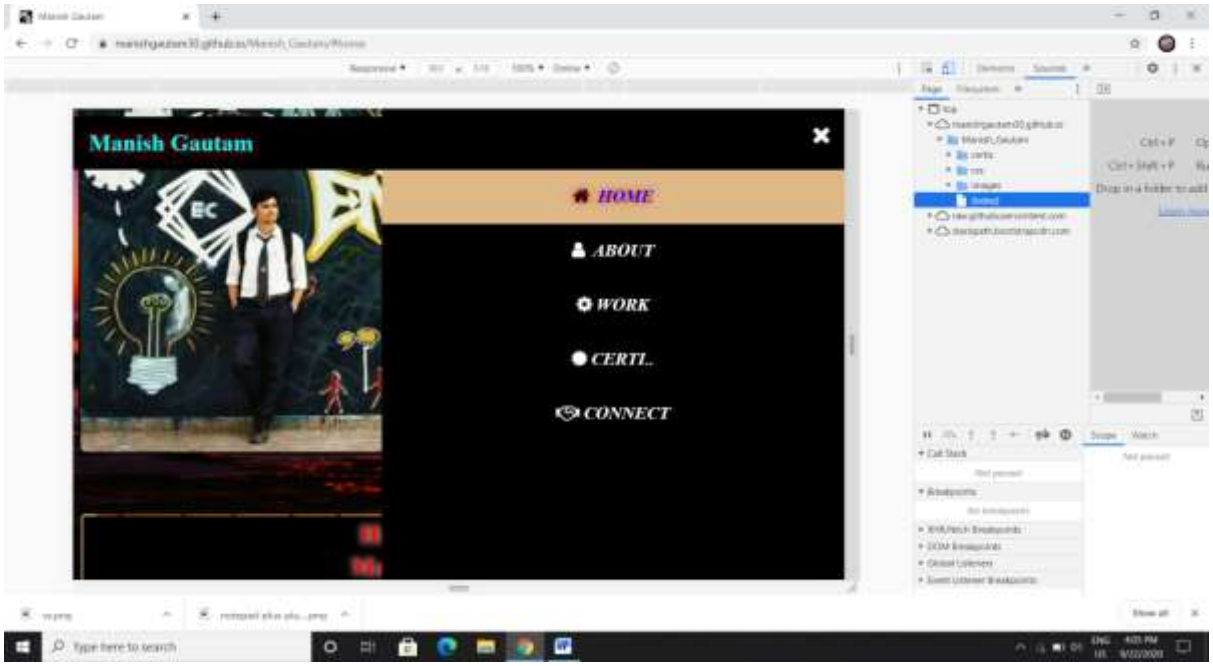
And at the top of the page lies a navigation bar with some button added in order to move or navigate anywhere on the portfolio webpage even having my name highlighted with the aqua color.

In this home page, there are some links on navigation bar, which navigate the user to that section with a smooth scrolling effect.

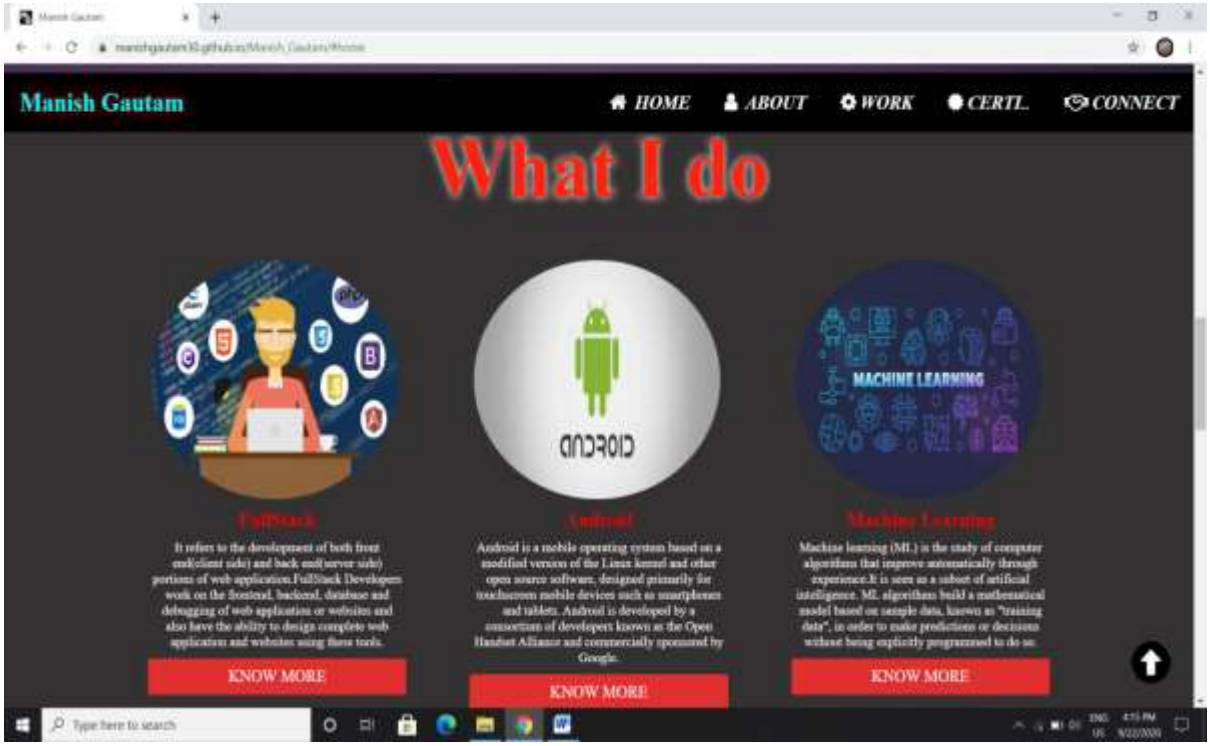
Next is the About Section, where I introduce myself.



This navigation bar section is responsive. Whenever the screen resolution is kept below 1000px in width, all the text get center aligned and these list items drop out to form columns tht split into 5 rows. 1 item each row.

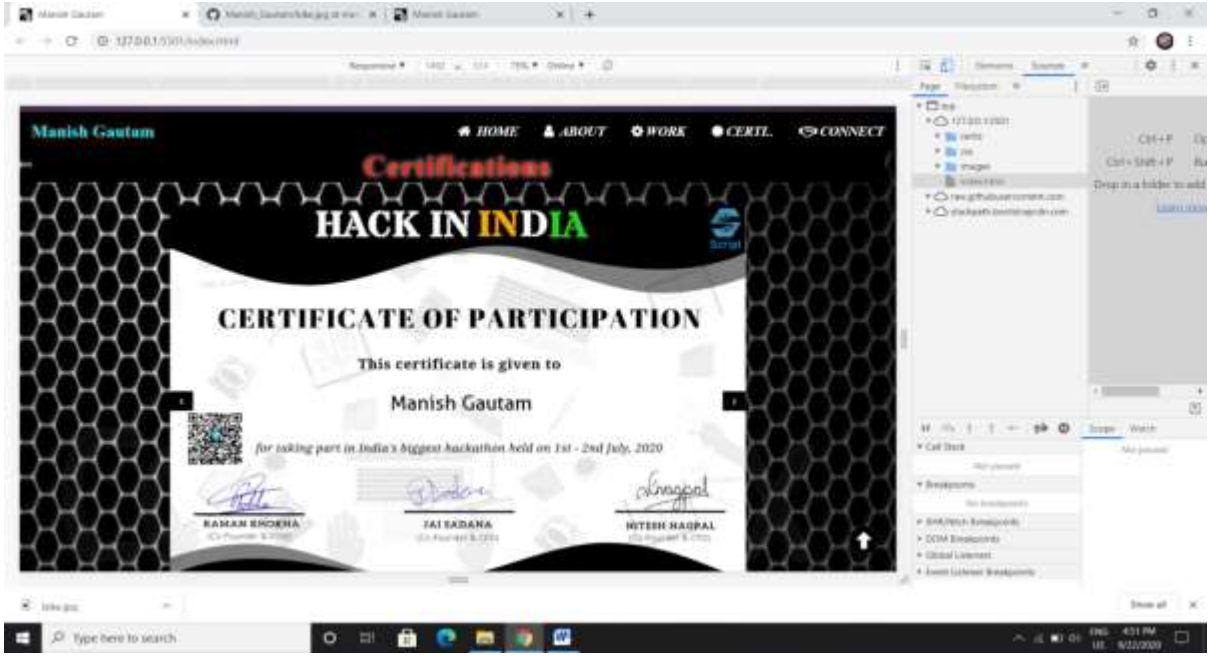


Next is Skills Section known as ‘WHAT I DO’, where I have written about my learning implementations and my skills. I have also used buttons to navigate to the following learning places from where we can do master in different skills.



This is my Work section, where I have displayed some of my work. It have the work done by me in different fields during my learning and implementation. There icons have a hover effect also. And it is also responsive. These 4 columns split into rows.

Next is Certification Section, where I attached my certificates in the form of scrolling bar. In this images of certificate change to the next and back with the buttons on left and right of the slider. I also use a dark gradient image in the background.



At last, there is a Follow me section, where I linked my social media accounts and a contact me button, where user can directly mail me. And there is a hover effect over the each button as well as the link.

Conclusion

As a conclusion, I can say that this internship was a great experience. Thanks to this project, I acquired deeper knowledge concerning my technical skills but I also personally benefited.

Currently web pages are a common part of web applications, and one of the most popular language for web development used by developers worldwide. If we surf internet we can see millions of websites built with HTML and CSS.

Indeed, I grew more independent in work and also in everyday life. I realized that I could do more things than I thought like learning new things by myself. There are huge opportunities available for the students who want to work in this field. Many private and public organizations hire web designer for their online work and website development.

With the rapid advent of online industry, the demand of web development professionals is increasing and this has created a huge job opportunity for the aspirants in the upcoming days.

Also an experienced person in this field can also work as a freelancer; there are many online companies which provide online projects to the individuals.

Internship Outcomes

Problems

There are more problems to create when I working with CSS. It is very easy but sometimes it's not matched with my expectation, and placed in appropriate place. Web designing is not an easy task. It takes a lot of creativity, uniqueness and brainstorming sessions to come up with a nice web design. Only a web designer can tell about the challenges they go through on regular basis. Web designs which are liked and appreciated by target market and clients are mostly the result of constant development and regressive critical thinking. Website's sync with multiple apps along with its ability to be responsive are just a few factors which are pondered by the web designer. This makes web designing a challenging role. On the contrary, there are some other challenges which are faced by web designers. These challenges are about keeping a website responsive enough so that it can be viewed and accessed on all devices. Sometimes websites take time to load, so in order to provide a better user experience, one of the toughest challenges for web design is to make website load faster [9]. Clients often encounter issues with their website and turn to the designer/developer to correct them, but when they check the site, it looks fine from their end. Sometimes, these issues are caused by out-of-date web servers or changes made incorrectly by the customer. Time spent detecting the source of a problem means a delay in tackling the problem itself.

Solutions

Really web development is so best and ultimate interested and hard for me, by the internship training I could learn an understand many objectives with interest as if was hard after learn I can understand web development is so interested then other language. I did solution of CSS and other problem in my training period. Clean code reduces unexpected errors, which can affect page load. Automated website monitoring solutions often allow the developers to view their clients' websites in real time and to set alerts to notify them when potential issues begin to arise. Not only does this allow the developers to identify a problem before the customer does, in many cases it also gives them the chance to fix it before the issue has any impact on the client's business.

Future Scope of the Project.

There are various modifications and transformations that can be done to this project.

Especially if I don't have any experience. Finding work can be a real challenge. A successful internship can help me turn an experience into a career opportunity.

So as a successful internship my future scopes will be:

- To be work IT company.
- Can work as a Software Engineer.
- Can work as a Web Designer.
- Can work as a Web Developer.
- Can work as a QA Tester.

Link for the code:

https://github.com/manishgautam30/Manish_Gautam

Link for the Portfolio-Site:

https://manishgautam30.github.io/Manish_Gautam

REFERENCES

- https://www.w3schools.com/html/
- https://www.beta-labs.in/2020/06/web-development.html
- https://www.coursera.org/learn/website-coding/home/welcome
- https://www.youtube.com/channel/UCFbNIlppjAuEX4znoulh0Cw
- https://www.youtube.com/watch?v=Kd2IAgT-orQ&t=417s
- https://www.tutorialspoint.com/html/html_tutorial.pdf
- https://fontawesome.com/icons
- https://github.com/tblong/Responsive-Website-Development-and-Design/tree/master/01_Responsive%20Website%20Basic