

**PROJECT REPORT**

**ON**

Work’n’Appoint

A

Job Providing Platform

**----------------------------**

**SUBMITTED IN THE PARTIAL FULFILLMENT FOR THE AWARD OF THE**

**DEGREE OF**

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE & ENGINEERING**

**Submitted By:**

**Rahul Saini(1217281)**

**Pankaj Verma(1217269)**

**Parneet Kaur(1217271)**

***Department of Computer Science and Engineering***

**Seth Jai Parkash Mukand Lal Institute Of Engineering & Technology, (JMIT) Radaur**

### (Kurukshetra University, Kurukshetra) (2017-2021)

**ACKNOWLEDGEMENT**

Perseverance, inspiration & motivation have always played a key role in success of any venture. At this level of understanding it is quite difficult to understand the wide spectrum of knowledge .Like all the other studies, Industrial training has its own importance. It is the most important part of the curriculum for an Engineering student. This work is the result of the interaction of a number of minds that directly or indirectly have contributed in the making of this training a success. I would like to extend my thankfulness to all my co-trainees for their flow of inspiration and cheerful support in the endeavour.

Rahul Saini (1217281)

Pankaj Verma (1217269)

Parneet Kaur (1217271)

#### DECLARATION

Perseverance, inspiration & motivation have always played a key role in success of any venture. At this level of understanding it is quite difficult to understand the wide spectrum of knowledge .Like all the other studies, Industrial training has its own importance. It is the most important part of the curriculum for an Engineering student. This work is the result of the interaction of a number of minds that directly or indirectly have contributed in the making of this training a success. I would like to extend my thankfulness to all my co-trainees for their flow of inspiration and cheerful support in the endeavour.

Rahul Saini (1217281)

Pankaj Verma (1217269)

Parneet Kaur (1217271)

#### 

#### TABLE OF CONTENTS

**S. No Contents Page No.**

1. Acknowledgment 3

2. Declaration 4

3. Table of Contents 5

4. List of Figures 6

5. Details of individual in project 8

6. Introduction 9

6.1 Objective 10

6.2 Description Of Project 11

7. System Development Life Cycle 12-17

7.1 Introduction 13-14

7.2 Feasibility Study 15

7.2.1 Economic Study 16

7.2.2 Technical Study 16

7.2.3 Behavioural Study 17

8. Software Req. Analysis & Specification 18-20

9. Development Environment 21-39

9.1 Introduction To HTML 22-25

9.2 Introduction To CSS 26-29

9.3 Introduction To Javascript 30-33

9.4 Introduction To Node Js 34-37

9.4 Introduction To Visual Studio 38-39

10. Testing and debugging 40-45

11. Conclusion 56

12. Future Scope 57

13. Bibliography 58

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **Figure No.** | **Name** | **Page No.** |

|  |  |  |
| --- | --- | --- |
| 1.1 | Stages of SDLC | 14-16 |
| 1.2 | Visual Studio Code | 46-49 |
| 1.3 | Architecture | 17 |
| 1.4 | Implementation example 1 | 50 |
| 1.5 | Implementation example 2 | 51 |
| 1.6 | Implementation example 3 | 51 |
| 1.7 | Implementation example 4 | 52 |
| 1.8 | Implementation example 5 | 53 |
| 1.9 | Implementation example 6 | 54 |
| 2.1 | Advantages | 55 |
| 2.2 | Plagerism | 59 |

**CHAPTER 1**

**DETAILS OF INDIVIDUAL IN PROJECT**

**THIS PROJECT IS CREATED BY:-**

* Rahul Saini

1217281

CSE-4th Year

JMIT

* Pankaj Verma

1217269

CSE-4th Year

JMIT

* Parneet Kaur

1217271

CSE-4th Year

JMIT

**CHAPTER 2**

**INTRODUCTION**

**OBJECTIVE**

This is basically a website that helps job seekers to find some job .It is a job providing platform. Companies can find the right person to complete a task and interested candidates may get the job.

It basically deals in providing employment to the job seekers. This website is basically designed to allow employers to post job requirements for a position to be filled and are commonly known as job boards.Candidates; especially the best ones crave information, they want to know about your company and it’s culture so they can make an informed decision and your Career Site is the best place for you to control this flow of information and sell them on your organisation**.**

**DESCRIPTION OF PROJECT**

This App is an Online Marketplace that provides a means for employers, students, and job seekers to showcase their skills by doing projects of their interest. Such projects are posted by registered clients with work descriptions. Thus ,this app works like a intemediator between clients and applicants. Persons can also get bid by submitting their projects on time and by satisfying the requirnments of clients. This project named “Work ‘n’ Appoint” is made by making the use of certain technologies like HTML,CSS, Javascript at front end and Node Js at Backend.

Any Job Seeker who is willing to get the job can visit this website .Both the Job Seeker and Job Provider will be directed to the main screen i.e Home Page. Any of them can Login or Signup. Bt in order to Login ,one must have its account. If the person is Job seeker, then he will be directed to a page where he/she has to choose skills and signup. If the person is Job Provider, then he will be directed to a page where he has to fill the company related details and make an account. After creating account ,They will be directed to the login page .After login ,Job seeker will be directed to a page where he can view all the jobs ,for which he can apply ,and Job Provider will be directed to a page where he can add new jobs as well as view the existing jobs as well which will be available on the home page.

On the home page only ,there are various technologies mentioned like angular, react etc. If the user clicks on them ,then all the jobs related to a particular technology will be shown on the main screen itself.

## This project/Website basically deals in providing the Jobs to the Job seekers thereby reducing his/her efforts of roaming here and there in search of job, making more efficient, reliable and time saving approach. Also reducing paper work by carrying handy resumes. Candidates; especially the best ones crave information, they want to know about your company and it’s culture so they can make an informed decision and your Career Site is the best place for you to control this flow of information and sell them on your organisation. It also reduces the cost of Hire.

**CHAPTER 3**

**SYSTEM DEVELOPMENT LIFE-CYCLE**

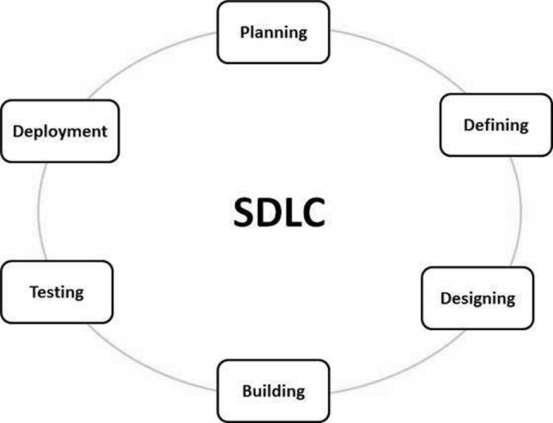
**INTRODUCTION**

Software Development Life Cycle (SDLC) is a process used by the software industry to design, develop and test high quality softwares. The SDLC aims to produce a high-quality software that meets or exceeds customer expectations, reaches completion within times and cost estimates.

* + - SDLC is the acronym of Software Development Life Cycle.
    - It is also called as Software Development Process.
    - ISO/IEC 12207 is an international standard for software life-cycle processes. It aims to be the standard that defines all the tasks required for developing and maintaining software.

### What is SDLC?

SDLC is a process followed for a software project, within a software organization. It consists of a detailed plan describing how to develop, maintain, replace and alter or enhance specific software. The life cycle defines a methodology for improving the quality of software and the overall development process.

Figure 4.1:Stages of SDLC

A typical Software Development Life Cycle consists of the following stages − Stage 1: Planning and Requirement Analysis

Requirement analysis is the most important and fundamental stage in SDLC. It is performed by the senior members of the team with inputs from the customer, the sales department, market surveys and domain experts in the industry. This information is then used to plan the basic project approach and to conduct product feasibility study in the economical, operational and technical areas.

Planning for the quality assurance requirements and identification of the risks associated with the project is also done in the planning stage. The outcome of the technical feasibility study is to define the various technical approaches that can be followed to implement the project successfully with minimum risks.

Stage 2: Defining Requirements

Once the requirement analysis is done the next step is to clearly define and document the product requirements and get them approved from the customer or the market analysts. This is done through an **SRS (Software Requirement Specification)** document which consists of all the product requirements to be designed and developed during the project life cycle.

Stage 3: Designing the Product Architecture

SRS is the reference for product architects to come out with the best architecture for the product to be developed. Based on the requirements specified in SRS, usually more than one design approach for the product architecture is proposed and documented in a DDS - Design Document Specification.

This DDS is reviewed by all the important stakeholders and based on various parameters as risk assessment, product robustness, design modularity, budget and time constraints, the best design approach is selected for the product.

A design approach clearly defines all the architectural modules of the product along with its communication and data flow representation with the external and third party modules (if any). The internal design of all the modules of the proposed architecture should be clearly defined with the minutest of the details in DDS.

Stage 4: Building or Developing the Product

In this stage of SDLC the actual development starts and the product is built. The programming code is generated as per DDS during this stage.

If the design is performed in a detailed and organized manner, code generation can be accomplished without much hassle. Developers must follow the coding guidelines defined by their organization and programming tools like compilers, interpreters, debuggers, etc. are used to generate the code. Different high level programming languages such as C, C++, Pascal, Java and PHP are used for coding. The programming language is chosen with respect to the type of software being developed.

Stage 5: Testing the Product

This stage is usually a subset of all the stages as in the modern SDLC models, the testing activities are mostly involved in all the stages of SDLC. However, this stage refers to the testing only stage of the product where product defects are reported, tracked, fixed and retested, until the product reaches the quality standards defined in the SRS.

Stage 6: Deployment in the Market and Maintenance

Once the product is tested and ready to be deployed it is released formally in the appropriate market. Sometimes product deployment happens in stages as per the business strategy of that organization. The product may first be released in a limited segment and tested in the real business environment (UAT- User acceptance testing). Then based on the feedback, the product may be released as it is or with suggested enhancements in the targeting market segment. After the product is released in the market, its maintenance is done for the existing customer base.

### SDLC Models

There are various software development life cycle models defined and designed which are followed during the software development process. These models are also referred as Software Development Process Models". Each process model follows a Series of steps unique to its type to ensure success in the process of software development.

Following are the most important and popular SDLC models followed in the industry –

* + - * Waterfall Model
      * Iterative Model
      * Spiral Model
      * V-Model
      * Big Bang Model

Other related methodologies are Agile Model, RAD Model, Rapid Application Development and Prototyping Models.

### FEASIBILITY STUDY

A feasibility study is an analysis that takes all of a project's relevant factors into account—including economic, technical, legal, and scheduling considerations—to ascertain the likelihood of completing the project successfully. Project managers use feasibility studies to discern the pros and cons of undertaking a project before they invest a lot of time and money into it. Feasibility studies also can provide a company's management with crucial information that could prevent the company from entering blindly into risky businesses.

Feasibility studies are important to business development. They can allow a business to address where and how it will operate. They can also identify potential obstacles that may impede its operations and recognize the amount of funding it will need to get the business up and running. Feasibility studies aim for marketing strategies that could help convince investors or banks that investing in a particular project or business is a wise choice.

A feasibility analysis evaluates the project’s potential for success; therefore, perceived objectivity is an essential factor in the credibility of the study for potential investors and lending institutions. There are three types of feasibility study—separate areas that a feasibility study examines, described below.

**Technological Feasibility**

This assessment focuses on the technical resources available to the organization. It helps organizations determine whether the technical resources meet capacity and whether the technical team is capable of converting the ideas into working systems. Technical feasibility also involves evaluation of the hardware, software, and other technical requirements of the proposed system.

Economical Feasibility

This assessment typically involves a cost/ benefits analysis of the project, helping organizations determine the viability, cost, and benefits associated with a project before financial resources are allocated.

It also serves as an independent project assessment and enhances project credibility—helping decision-makers determine the positive economic benefits to the organization that the proposed project will provide.

**Behavioral Feasibility**

This assessment involves undertaking a study to analyze and determine whether—and how well—the organization’s needs can be met by completing the project. Operational feasibility studies also examine how a project plan satisfies the requirements identified in the requirements analysis phase of system development.

**CHAPTER 4**

**SOFTWARE REQUIRNMENT ANALYSIS AND SPECIFICATIONS**

**SRS** stands for Software Requirements Specification, which is a document that fully describes the expected behavior of a software system. Functional requirements are documented in an SRS, as are non- functional requirements such as performance goals and descriptions of quality attributes.

Software requirements specification establishes the basis for an agreement between customers and contractors or suppliers on how the software product should function (in a market-driven project, these roles may be played by the marketing and development divisions). Software requirements specification is a rigorous assessment of requirements before the more specific system design stages, and its goal is to reduce later redesign. It should also provide a realistic basis for estimating product costs, risks, and schedules. Used appropriately, software requirements specifications can help prevent software project failure The Software Requirements Specification (SRS) is a communication tool between users and software designers. The specific goals of the SRS are:

* + - * Facilitating reviews
      * Describing the scope of work
      * Providing a reference to software designers (i.e. navigation aids, document structure)
      * Providing a framework for testing primary and secondary use cases Inc[luding features](https://en.wikipedia.org/wiki/Software_feature) to customer requirements
      * Providing a platform for ongoing refinement (via incomplete specs or questions)

An SRS should address, among other things:

* **Functionality of the software:** What the software will do
* **External interfaces:** How the given software will interact with hardware, other software(s) and assumptions on these entities
* **Required performance levels:** Required performance levels such as response rate, recovery rate etc. of the software
* **Quality attributes:** The non-functional factors that are used to evaluate the performance of the software, such as security, safety, portability etc
* **Design constraints:** Any operating system limitations (e.g.: the stock exchange software will only run on Windows), implementation language etc that will affect or limit the design of the software.

A good SRS defines the how Software System will interact with all internal modules, hardware, communication with other programs and human user interactions with wide range of real life scenarios.

Using the *Software requirements specification* (SRS) document on QA lead, managers creates test plan. It is very important that testers must be cleared with every detail specified in this document in order to avoid faults in test cases and its expected results.

It is highly recommended to review or test SRS documents before start writing test cases and making any plan for testing. Let’s see how to test SRS and the important point to keep in mind while testing it.

### PURPOSE OF SRS

The purpose of the document is to collect and analyze all assorted ideas that have come up to define the system, its requirements with respect to consumers. Also, we shall predict and sort out how we hope this product will be used in order to gain a better understanding of the project, outline concepts that may be developed later, and document ideas that are being considered, but may be discarded as the product develops.

In short, the purpose of this SRS document is to provide a detailed overview of our software product, its parameters and goals. This document describes the project's target audience and its user interface, hardware and software requirements. It defines how our client, team and audience see the product and its functionality. Nonetheless, it helps any designer and developer to assist in software delivery lifecycle (SDLC) processes.

Software requirements specification(SRS) is important for developers because it minimizes the amount of time and effort developers have to expend to achieve desired software goals. It thus reduces development cost. This also benefits the client company because the lesser the development cost, the lesser the developers will charge from the client. And, if composed properly, an SRS ensures that there is less possibility of future redesigns as there is less chance of mistake on the part of developers as they have a clear idea on the functionalities and externalities of the software. It also helps clear any communication problems between the client and the developer. Furthermore, an SRS serves to form a foundation of mutual agreement between the client and the developer (supplier). It also serves as the document to verify the testing processes. Importance of SRS:-

* A software requirements specification is the basis for your entire project. It lays the framework that every team involved in development will follow.
* It’s used to provide critical information to multiple teams — development, quality assurance, operations, and maintenance. This keeps everyone on the same page.
* Using the SRS helps to e nsure requirements are fulfilled. And it can also help you make decisions about your product’s lifecycle — for instance, when to retire a feature.

**CHAPTER 5**

**DEVELOPMENT ENVIRONMENT**

**INTRODUCTION TO HTML:**

**HTML** stands for Hyper Text Markup Language. It is used to design web pages using markup language. HTML is the combination of Hypertext and Markup language. Hypertext defines the link between the web pages. Markup language is used to define the text document within tag which defines the structure of web pages. This language is used to annotate (make notes for the computer) text so that a machine can understand it and manipulate text accordingly. Most markup languages (e.g. HTML) are human readable. Language uses tags to define what manipulation has to be done on the text.HTML is a markup language used by the browser to manipulate text, images and other content, in order to display it in the required format. HTML was created by Tim Berners-Lee in 1991. The first ever version of HTML was HTML 1.0, but the first standard version was HTML 2.0, published in 1999.The basic structure of an HTML page is laid out below. It contain the essential building-block elements (i.e. doctype declaration, html, head, title, and body elements) upon which all webpages are created.

* It is easy to learn and easy to use.
* It is platform independent
* Images, video and audio can be added to a web page.
* Hypertext can be added to text.
* It is a markup language.

An HTML document can be created using any text editor . Save the text file using **.html** or **.htm**. Once saved as an HTML document, the file can be opened as a webpage in the browser.

Initially, the HTML was released in the year of 1993 developed by Tim Berners-Lee in 1990. Currently, the **WHATWG** community working on the development of the HTML. The current HTML5 version gains so much popularity because of newly added features in it. All the websites skeleton are made of HTML and each browser performing on that and makes that visible and user-friendly to the user.

* **Easy to understand:** It is the easiest language you can say, very easy to grasp this language and easy to develop.
* **Flexibility:** This language is so much flexible that you can create whatever you want, a flexible way to design web pages along with the text.
* **Linkable:** You can make linkable text like users can connect from one page to another page or website through these characteristics.
* **Limitless features:** You can add videos, gifs, pictures or sound anything you want that will make the website more attractive and understandable.

Each HTML page consists of a set of **tags** (also called **elements**), which you can refer to as the building blocks of web pages. They create a hierarchy that structures the content into sections, paragraphs, headings, and other content blocks.

Most HTML elements have an opening and a closing that use the ***<tag></tag>*** syntax.

Since the first days, HTML has gone through an incredible evolution. W3C constantly publish new versions and updates, while historical milestones get dedicated names as well.HTML4 (these days commonly referred to as “HTML”) was published in 1999, while the latest major version came out in 2014. Named **HTML5**, the update has introduced many new features to the language.

One of the most anticipated features of HTML5 is native support for audio and video embedding. Instead of using Flash player, we can simply embed videos and audio files to our web pages using the new ***<audio> </audio>***  and  ***<video> </video>***  tags.

It also includes in-built support for scalable vector graphics (SVG) and Math ML for mathematical and scientific formulas.HTML5 introduced a few semantic improvements as well. The new semantic tags inform browsers about the meaning of content, which benefits both readers and search engines. The most popular semantic tags are ***<article></article>***, ***<section></section>***, ***<aside></aside>***, ***<header></header>***, and ***<footer></footer>***. To find a more unique differences, consider checking our [**in-depth HTML and HTML5 comparison**](https://www.hostinger.com/tutorials/difference-between-html-and-html5).

While HTML is a powerful language, it isn’t enough to build a professional and fully responsive website. We can only use it to add text elements and create the structure of the content.

However, HTML works extremely well with two other frontend languages: [CSS (Cascading Style Sheets)](https://www.hostinger.com/tutorials/css-cheat-sheet), and [JavaScript](https://www.hostinger.com/tutorials/what-is-javascript). Together, they can achieve rich user experience and implement advanced functions.

* CSS is responsible for styling s such as background, colors, layouts, spacing, and animations.
* JavaScript lets you add dynamic functionality such as sliders, pop-ups, and photo galleries.

Think of HTML as a naked person, CSS as the clothing, and JavaScript as movements and manner.

## So…What is HTML?

HTML is the main markup language of the web.

It runs natively in every browser and is maintained by the World Wide Web Consortium.

You can use it to create the content structure of websites and web applications. It’s the lowest level of frontend technologies, that serves as the basis for styling you can add with CSS and functionality you can implement using JavaScript.

HTML is not a programming language, meaning it doesn’t have the ability to create dynamic functionality. Instead, it makes it possible to organize and format documents, similarly to Microsoft Word. When working with HTML, we use simple code structures (tags and attributes) to mark up a website page.

Hypertext means a text that contains references (links) to other texts that viewers can access immediately. He published the first version of HTML in 1991, consisting of 18 HTML tags. Since then, each new version of the HTML language came with new tags and attributes (tag modifiers) to the markup.

According to Mozilla Developer Network’s [HTML Element Reference](https://developer.mozilla.org/en-US/docs/Web/HTML/Element), currently, there are 140 HTML tags, although some of them are already obsolete (not supported by modern browsers).

Due to a quick rise in popularity, HTML is now considered an official web standard. The HTML specifications are maintained and developed by the World Wide Web Consortium (W3C). You can check out the latest state of the language anytime on [W3C’s website](https://www.w3.org/).

The biggest upgrade of the language was the introduction of **HTML5** in 2014. It added several new semantic tags to the markup, that reveal the meaning of their own content, such as **<article>**, **<header>**, and **<footer>**.

HTML documents are files that end with a .***html*** or **.htm**extension. You can view then using any web browser (such as Google Chrome, Safari, or Mozilla Firefox). The browser reads the HTML file and renders its content so that internet users can view it.

Usually, the average [website includes several different HTML pages](https://www.hostinger.com/tutorials/must-have-pages-for-business-websites). For instance: home pages, about pages, contact pages would all have separate HTML documents.

Each HTML page consists of a set of **tags** (also called **elements**), which you can refer to as the building blocks of web pages. They create a hierarchy that structures the content into sections, paragraphs, headings, and other content blocks.

Most HTML elements have an opening and a closing that use the ***<tag></tag>*** syntax.

Berners-Lee considered HTML to be an application of SGML. It was formally defined as such by the [Internet Engineering Task Force](https://en.wikipedia.org/wiki/Internet_Engineering_Task_Force) (IETF) with the mid-1993 publication of the first proposal for an HTML specification, the "Hypertext Markup Language (HTML)" Internet Draft by Berners-Lee and [Dan Connolly](https://en.wikipedia.org/wiki/Dan_Connolly_(computer_scientist)), which included an SGML [Document type definition](https://en.wikipedia.org/wiki/Document_type_definition) to define the grammar. The draft expired after six months, but was notable for its acknowledgment of the [NCSA Mosaic](https://en.wikipedia.org/wiki/Mosaic_(web_browser)) browser's custom tag for embedding in-line images, reflecting the IETF's philosophy of basing standards on successful prototypes. Similarly, [Dave Raggett](https://en.wikipedia.org/wiki/Dave_Raggett)’s competing Internet-Draft, "HTML+ (Hypertext Markup Format)", from late 1993, suggested standardizing already-implemented features like tables and fill-out forms.

After the HTML and HTML+ drafts expired in early 1994, the IETF created an HTML Working Group, which in 1995 completed "HTML 2.0", the first HTML specification intended to be treated as a standard against which future implementations should be based.

**INTRODUCTION TO CSS:**

**C**ascading **S**tyle **S**heets, fondly referred to as **CSS**, is a simply designed language intended to simplify

the process of making web pages presentable. CSS allows you to apply styles to web pages. More

importantly, CSS enables you to do this independent of the HTML that makes up each web page.

CSS is easy to learn and understood but it provides powerful control over the presentation of an

HTML document.

* **CSS saves time :**You can write CSS once and reuse same sheet in multiple HTML pages.
* **Easy Maintainence :**To make a global change simply change the style, and all elements in all the webpages will be updated automatically.
* **Search Engines :**CSS is considered as clean coding technique, which means search engines won’t have to struggle to “read” its content.
* **Superior styles to HTML :**CSS has a much wider array of attributes than HTML, so you can give a far better look to your HTML page in comparison to HTML attributes.
* **Offline Browsing :**CSS can store web applications locally with the help of offline catche. Using of this we can view offline websites.

Cascading Style Sheets level 1 (CSS1) came out as a recommendation in December 1996. This version describes the CSS language as well as a simple visual formatting model for all the HTML tags.CSS2 became a W3C recommendation in May 1998 and builds on CSS1. This version adds support for media-specific style sheets .

Basically CSS gives the outer cover on any HTML elements. If you consider HTML as a skeleton of the web-page then the CSS will be the skin of the skeleton. The Internet media type (MIME type) of CSS is text/CSS. The CSS was developed by the World Wide Web Consortium (W3C) in the year of 1996. The CSS can be applied to HTML documents in different ways.

In CSS, there are lots of selectors (ID selectors, Class Selectors, etc.) that will be helpful to perform specific tasks. It is easy to maintain, changing in a single place will affect globally in your web site. No need to change every specific place.

In the [Introduction to HTML](https://developer.mozilla.org/en-US/docs/Learn/HTML/Introduction_to_HTML) module we covered what HTML is, and how it is used to mark up documents. These documents will be readable in a web browser. Headings will look larger than regular text, paragraphs break onto a new line and have space between them. Links are colored and underlined to distinguish them from the rest of the text. What you are seeing is the browser's default styles — very basic styles that the browser applies to HTML to make sure it will be basically readable even if no explicit styling is specified by the author of the page.

As we have mentioned before, CSS is a language for specifying how documents are presented to users — how they are styled, laid out, etc.

A **document** is usually a text file structured using a markup language — [HTML](https://developer.mozilla.org/en-US/docs/Glossary/HTML) is the most common markup language, but you may also come across other markup languages such as [SVG](https://developer.mozilla.org/en-US/docs/Glossary/SVG) or [XML](https://developer.mozilla.org/en-US/docs/Glossary/XML).

**Presenting** a document to a user means converting it into a form usable by your audience. [Browsers](https://developer.mozilla.org/en-US/docs/Glossary/browser), like [Firefox](https://developer.mozilla.org/en-US/docs/Glossary/Mozilla_Firefox), [Chrome](https://developer.mozilla.org/en-US/docs/Glossary/Google_Chrome), or [Edge](https://developer.mozilla.org/en-US/docs/Glossary/Microsoft_Edge) , are designed to present documents visually, for example, on a computer screen, projector or printer.

CSS can be used for very basic document text styling — for example changing the [color](https://developer.mozilla.org/en-US/docs/Web/CSS/color_value) and [size](https://developer.mozilla.org/en-US/docs/Web/CSS/font-size) of headings and links. It can be used to create layout — for example [turning a single column of text into a layout](https://developer.mozilla.org/en-US/docs/Web/CSS/Layout_cookbook/Column_layouts) with a main content area and a sidebar for related information. It can even be used for effects such as [animation](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Animations). Have a look at the links in this paragraph for specific examples.

CSS is a rule-based language — you define rules specifying groups of styles that should be applied to particular elements or groups of elements on your web page. For example "I want the main heading on my page to be shown as large red text."

As there are so many things that you could style using CSS, the language is broken down into modules. You'll see reference to these modules as you explore MDN and many of the documentation pages are organized around a particular module. For example, you could take a look at the MDN reference to the [Backgrounds and Borders](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Backgrounds_and_Borders) module to find out what its purpose is, and what different properties and other features it contains. You will also find links to the CSS Specification that defines the technology (see below).

At this stage you don't need to worry too much about how CSS is structured, however it can make it easier to find information if, for example, you are aware that a certain property is likely to be found among other similar things and are therefore probably in the same specification.

For a specific example, let's go back to the Backgrounds and Borders module — you might think that it makes logical sense for the [background-color](https://developer.mozilla.org/en-US/docs/Web/CSS/background-color) and [border-color](https://developer.mozilla.org/en-US/docs/Web/CSS/border-color) properties to be defined in this module. And you'd be right.

All web standards technologies (HTML, CSS, JavaScript, etc.) are defined in giant documents called specifications (or simply "specs"), which are published by standards organizations (such as the [W3C](https://developer.mozilla.org/en-US/docs/Glossary/W3C), [WHATWG](https://developer.mozilla.org/en-US/docs/Glossary/WHATWG), [ECMA](https://developer.mozilla.org/en-US/docs/Glossary/ECMA), or [Khronos](https://developer.mozilla.org/en-US/docs/Glossary/Khronos)) and define precisely how those technologies are supposed to behave.

CSS is no different — it is developed by a group within the W3C called the [CSS Working Group](https://www.w3.org/Style/CSS/). This group is made of representatives of browser vendors and other companies who have an interest in CSS. There are also other people, known as invited experts, who act as independent voices; they are not linked to a member organization.

New CSS features are developed, or specified, by the CSS Working Group. Sometimes because a particular browser is interested in having some capability, other times because web designers and developers are asking for a feature, and sometimes because the Working Group itself has identified a requirement. CSS is constantly developing, with new features coming available. However, a key thing about CSS is that everyone works very hard to never change things in a way that would break old websites. A website built in 2000, using the limited CSS available then, should still be usable in a browser today!

As a newcomer to CSS, it is likely that you will find the CSS specs overwhelming — they are intended for engineers to use to implement support for the features in user agents, not for web developers to read to understand CSS. Many experienced developers would much rather refer to MDN documentation or other tutorials. It is however worth knowing that they exist, understanding the relationship between the CSS you are using, browser support (see below), and the specs.

Once CSS has been specified then it is only useful for us in developing web pages if one or more browsers have implemented it. This means that the code has been written to turn the instruction in our CSS file into something that can be output to the screen. We'll look at this process more in the lesson [How CSS works](https://developer.mozilla.org/en-US/docs/Learn/CSS/First_steps/How_CSS_works). It is unusual for all browsers to implement a feature at the same time, and so there is usually a gap where you can use some part of CSS in some browsers and not in others. For this reason, being able to check implementation status is useful. On each property page on MDN you can see the status of the property you are interested in, so you can tell if you will be able to use it on a website.

When a browser displays a document, it must combine the document's content with its style information. It processes the document in a number of stages, which we've listed below. Bear in mind that this is a very simplified version of what happens when a browser loads a webpage, and that different browsers will handle the process in different ways. But this is roughly what happens.

1. The browser loads the HTML (e.g. receives it from the network).
2. It converts the [HTML](https://developer.mozilla.org/en-US/docs/Glossary/HTML) into a [DOM](https://developer.mozilla.org/en-US/docs/Glossary/DOM) (Document Object Model). The DOM represents the document in the computer's memory. The DOM is explained in a bit more detail in the next section.
3. The browser then fetches most of the resources that are linked to by the HTML document, such as embedded images and videos ... and linked CSS! JavaScript is handled a bit later on in the process, and we won't talk about it here to keep things simpler.
4. The browser parses the fetched CSS, and sorts the different rules by their selector types into different "buckets", e.g. element, class, ID, and so on. Based on the selectors it finds, it works out which rules should be applied to which nodes in the DOM, and attaches style to them as required (this intermediate step is called a render tree).
5. The render tree is laid out in the structure it should appear in after the rules have been applied to it.
6. The visual display of the page is shown on the screen (this stage is called painting).

The browser will parse the HTML and create a DOM from it, then parse the CSS. Since the only rule available in the CSS has a span selector, the browser will be able to sort the CSS very quickly! It will apply that rule to each one of the three <span>s, then paint the final visual representation to the screen.

In addition, many people are not using the latest version of a browser. Given that CSS is being developed all the time, and is therefore ahead of what browsers can recognise, you might wonder what happens if a browser encounters a CSS selector or declaration it doesn't recognise.

The answer is that it does nothing, and just moves on to the next bit of CSS!

If a browser is parsing your rules, and encounters a property or value that it doesn't understand, it ignores it and moves on to the next declaration. It will do this if you have made an error and misspelled a property or value, or if the property or value is just too new and the browser doesn't yet support it.

Similarly, if a browser encounters a selector that it doesn't understand, it will just ignore the whole rule and move on to the next one.

In the example below I have used the British English spelling for color, which makes that property invalid as it is not recognised. So my paragraph has not been colored blue. All of the other CSS have been applied however; only the invalid line is ignored.

**INTRODUCTION TO JAVASCRIPT:**

JavaScript was initially created to “make web pages alive”.The programs in this language are called scripts. They can be written right in a web page’s HTML and run automatically as the page loads.Scripts are provided and executed as plain text. They don’t need special preparation or compilation to run.Today, JavaScript can execute not only in the browser, but also on the server, or actually on any device that has a special program called [the JavaScript engine](https://en.wikipedia.org/wiki/JavaScript_engine).The browser has an embedded engine sometimes called a “JavaScript virtual machine”.

Different engines have different “codenames”. For example:

* [V8](https://en.wikipedia.org/wiki/V8_(JavaScript_engine)) – in Chrome and Opera.
* [SpiderMonkey](https://en.wikipedia.org/wiki/SpiderMonkey) – in Firefox.
* …There are other codenames like “Chakra” for IE, “ChakraCore” for Microsoft Edge, “Nitro” and “SquirrelFish” for Safari, etc.

The terms above are good to remember because they are used in developer articles on the internet. We’ll use them too. For instance, if “a feature X is supported by V8”, then it probably works in Chrome and Opera.JavaScript is the only browser technology that combines these three things. That’s what makes JavaScript unique. That’s why it’s the most widespread tool for creating browser interfaces. That said, JavaScript also allows to create servers, mobile applications etc JavaScript is standardized at [Ecma International](https://www.ecma-international.org/) — the European association for standardizing information communication systems (ECMA was formerly an acronym for the European Computer Manufacturers Association) to deliver a standardized, international programming language based on JavaScript. This standardized version of JavaScript, called ECMA Script, behaves the same way in all applications that support the standard. Companies can use the open standard language to develop their implementation of JavaScript. The ECMA Script standard is documented in the ECMA-262 specification. See [New in JavaScript](https://developer.mozilla.org/en-US/docs/Web/JavaScript/New_in_JavaScript) to learn more about different versions of JavaScript and ECMA Script specification editions. The ECMA-262 standard is also approved by the [ISO](https://www.iso.org/home.html) (International Organization for Standardization) as ISO-16262. You can also find the specification on [the Ecma International website](https://www.ecma-international.org/publications/standards/Ecma-262.htm). The ECMA Script specification does not describe the Document Object Model (DOM), which is standardized by the [World Wide Web Consortium (W3C)](https://www.w3.org/) and/or [WHATWG (Web Hypertext Application Technology Working Group)](https://whatwg.org/). The DOM defines the way in which HTML document objects are exposed to your script. To get a better idea about the different technologies that are used when programming with JavaScript,

consult the article [JavaScript technologies overview](https://developer.mozilla.org/en-US/docs/Web/JavaScript/JavaScript_technologies_overview).

In JavaScript, most things are objects, from core JavaScript features like strings and arrays to the browser APIs built on top of JavaScript. You can even create your own objects to encapsulate related functions and variables into efficient packages. The object-oriented nature of JavaScript is important to understand if you want to go further with your knowledge of the language and write more efficient code, therefore we've provided this module to help you. Here we teach object theory and syntax in detail, look at how to create your own objects, and explain what JSON data is and how to work with it.

Being a scripting language, **JavaScript cannot run on its own. In fact, the browser is responsible for running JavaScript code**. When a user requests an HTML page with JavaScript in it, the script is sent to the browser and it is up to the browser to execute it. The main advantage of JavaScript is that **all modern web browsers support** JavaScript. So, you do not have to worry about whether your site visitor uses Internet Explorer, Google Chrome, Firefox or any other browser. JavaScript will be supported. Also, JavaScript **runs on any operating system** including Windows,[Linux](https://www.guru99.com/unix-linux-tutorial.html)or Mac. Thus, JavaScript overcomes the main disadvantages of[VBScript](https://www.guru99.com/vbscript-tutorials-for-beginners.html)(Now deprecated) which is limited to just IE and Windows.

Client-side JavaScript is the most common form of the language. The script should be included in or referenced by an HTML document for the code to be interpreted by the browser.

It means that a web page need not be a static HTML, but can include programs that interact with the user, control the browser, and dynamically create HTML content.

The JavaScript client-side mechanism provides many advantages over traditional CGI server-side scripts. For example, you might use JavaScript to check if the user has entered a valid e-mail address in a form field.

The JavaScript code is executed when the user submits the form, and only if all the entries are valid, they would be submitted to the Web Server.

JavaScript can be used to trap user-initiated events such as button clicks, link navigation, and other actions that the user initiates explicitly or implicitly.

The merits of using JavaScript are −

* **Less server interaction** − You can validate user input before sending the page off to the server. This saves server traffic, which means less load on your server.
* **Immediate feedback to the visitors** − They don't have to wait for a page reload to see if they have forgotten to enter something.

**Increased interactivity** − You can create interfaces that react when the user hovers over them with a mouse or activates them via the keyboard.

* **Richer interfaces** − You can use JavaScript to include such items as drag-and-drop components and sliders to give a Rich Interface to your site visitors.

One of major strengths of JavaScript is that it does not require expensive development tools. You can start with a simple text editor such as Notepad. Since it is an interpreted language inside the context of a web browser, you don't even need to buy a compiler.To make our life simpler, various vendors have come up with very nice JavaScript editing tools. Some of them are listed here −

* **Microsoft FrontPage** − Microsoft has developed a popular HTML editor called FrontPage. FrontPage also provides web developers with a number of JavaScript tools to assist in the creation of interactive websites.
* **Macromedia Dreamweaver MX** − Macromedia Dreamweaver MX is a very popular HTML and JavaScript editor in the professional web development crowd. It provides several handy prebuilt JavaScript components, integrates well with databases, and conforms to new standards such as XHTML and XML.
* **Macromedia HomeSite 5** − HomeSite 5 is a well-liked HTML and JavaScript editor from Macromedia that can be used to manage personal websites effectively.
* The ECMAScript Edition 5 standard will be the first update to be released in over four years. JavaScript 2.0 conforms to Edition 5 of the ECMAScript standard, and the difference between the two is extremely minor.
* The specification for JavaScript 2.0 can be found on the following site: <http://www.ecmascript.org/>

Today, Netscape's JavaScript and Microsoft's JScript conform to the ECMAScript standard, although both the languages still support the features that are not a part of the standard.

JavaScript is a dynamic scripting language supporting [prototype based](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Details_of_the_Object_Model#Class-Based_vs._Prototype-Based_Languages) object construction. The basic syntax is intentionally similar to both Java and C++ to reduce the number of new concepts required to learn the language. Language constructs, such as if statements, for and while loops, and switch and try ... catch blocks function the same as in these languages (or nearly so).JavaScript can function as both a [procedural](https://en.wikipedia.org/wiki/Procedural_programming) and an [object oriented language](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Introduction_to_Object-Oriented_JavaScript). Objects are created programmatically in JavaScript, by attaching methods and properties to otherwise empty objects **at run time**, as opposed to the syntactic class definitions common in compiled languages like C++ and Java.

Once an object has been constructed it can be used as a blueprint (or prototype) for creating similar objects. JavaScript's dynamic capabilities include runtime object construction, variable parameter lists, function variables, dynamic script creation (via [eval](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/eval)), object introspection (via for ... in), and source code recovery (JavaScript programs can decompile function bodies back into their source text).

The Mozilla project provides two JavaScript implementations. The first **ever** JavaScript was created by Brendan Eich at Netscape, and has since been updated to conform to ECMA-262 Edition 5 and later versions. This engine, code named [SpiderMonkey](https://developer.mozilla.org/en-US/docs/Mozilla/Projects/SpiderMonkey), is implemented in C/C++. The [Rhino](https://developer.mozilla.org/en-US/docs/Rhino) engine, created primarily by Norris Boyd (also at Netscape) is a JavaScript implementation written in Java. Like SpiderMonkey, Rhino is ECMA-262 Edition 5 compliant.Several major runtime optimizations such as TraceMonkey (Firefox 3.5), JägerMonkey (Firefox 4) and IonMonkey were added to the SpiderMonkey JavaScript engine over time. Work is always ongoing to improve JavaScript execution performance.

**INTRODUCTION TO NODE.JS:**

Node.js is an open source and cross-platform runtime environment for executing [JavaScript](https://www.geeksforgeeks.org/JavaScript-tutorial/) code outside of a browser. You need to remember that **Node JS is not a framework and it’s not a programming language**. Most of the people are confused and understand it’s a framework or a programming language. We often use Node.js for building back-end services like APIs like Web App or Mobile App. It’s used in production by large companies such as Paypal, Uber , Netflix, Wallmart and so on.

There are other programming languages also which we can use to build back-end services so what makes Node.js different I am going to explain.

1. It’s easy to get started and can be used for prototyping and agile development
2. It provide fast and highly scalable services
3. It uses JavaScript everywhere so it’s easy for a JavaScript programmer to build back-end services using Node.js
4. Source code more cleaner and consistent.
5. Large ecosystem for open source library.
6. It has Asynchronous or Non blocking nature.

**BENEFITS:**

1. **Easy Scalability:** Developers prefer to use Node.js because it is easily scale the application in both horizontal and vertical direction. We can also add extra resources during the scalability of application.
2. **Real time web apps:** If you are building a web app you can also use PHP and it will take the same amount of time when you use Node.js, But if I am talking about building chat apps or gaming apps Node.js is much more preferable because of faster synchronization. Also, event loop avoid HTTP overload for Node.js development.
3. **Fast Suite:** NodeJs runs on the V8 engine developed by Google. Event loop in NodeJs handles all asynchronous operation so NodeJs acts like a fast suite and all the operations can be done quickly like reading or writing in the database, network connection or file system
4. **Easy to learn and code:** NodeJs is easy to learn and code because it uses JavaScript. If you are a front-end developer and have a good grasp on JavaScript you can easily learn and build the application on NodeJS
5. **Advantage of Caching:**It provides the caching of single module. Whenever there is any request for the first module, it gets cached in the application memory so you don’t need to re-execute the code.
6. **Data Streaming:** In NodeJs HTTP request and response are considered as two separate events.

Node.js is similar in design to, and influenced by, systems like Ruby's [Event Machine](https://github.com/eventmachine/eventmachine) and Python's [Twisted](https://twistedmatrix.com/trac/). Node.js takes the event model a bit further. It presents an [event loop](https://nodejs.org/en/docs/guides/event-loop-timers-and-nexttick/) as a runtime construct instead of as a library. In other systems, there is always a blocking call to start the event-loop. Typically, behavior is defined through callbacks at the beginning of a script, and at the end a server is started through a blocking call like EventMachine::run(). In Node.js, there is no such start-the-event-loop call. Node.js simply enters the event loop after executing the input script. Node.js exits the event loop when there are no more callbacks to perform. This behavior is like browser JavaScript — the event loop is hidden from the user.

HTTP is a first-class citizen in Node.js, designed with streaming and low latency in mind. This makes Node.js well suited for the foundation of a web library or framework.

Node.js being designed without threads doesn't mean you can't take advantage of multiple cores in your environment. Child processes can be spawned by using our [child\_process.fork()](https://nodejs.org/api/child_process.html#child_process_child_process_fork_modulepath_args_options) API, and are designed to be easy to communicate with. Built upon that same interface is the [cluster](https://nodejs.org/api/cluster.html) module, which allows you to share sockets between processes to enable load balancing over your cores.

Node.js is an epitome of an exceptionally customizable and scalable tech. The server engine utilizes an event-based, non-blocking I/O model. This makes the adaptation of Javascript easier to the machine language providing execution of the code super fast. Thanks to Javascript and Node.js, the code operates faster in server-to-client direction. This enhances the performance ability of the web applications to the next level. To be more precise, web application development in Node.js ensures a steady and secure non-blocking I/O model, simplifying the code beautifully.

Node.js runs over Google’s  V8 Javascript engine, where web applications are event-based in an asynchronous manner. Node.js platform uses a “*single-threaded event loop*.”

So, how exactly does Node.js handle concurrent requests along with a single-threaded model?  Well! “*Multi-threaded request-response*” architecture is an event loop which is much slower and unable to handle multiple concurrent threads at a time.

## Node.js Architecture

Unlike the traditional web-serving technique, where each request creates new thread cramping up the system RAM, Node.js operates on a single thread. This enables it to support thousands of concurrent connection handling event loops.

### NPM: Node Package Manager While we are trying to understand the basics of Node.js, we definitely don’t want to miss out on its built-in support for package management using npm. A popular package library – npm is the most prized possession of Node.js community. It contains millions of downloadable libraries according to the specific requirement. These massive libraries are absolutely free of cost with its registry. With each passing day, these libraries are getting bigger rapidly, making the Node.js community stronger.

Open-source developers from every continent use npm to share or borrow packages, and many enterprises utilize them for their private developments too. Find the entire list of packaged modules on [npm website](https://npmjs.com/)

The latest Node’s version 11.0 has some remarkable features. Node.js 11.0 uses native Node, making it even more stable. More error codes and interval time has enhanced better performance and improvised diagnosis. This has made the geeky life of the developers more delightful by deploying extraordinary experiences for users.

Beat your competition with Node.js based business solutions. Let me fire some quickies for you:

* Backend for Social Media Networking
* Single-page Application
* Chat Application
* Data Streaming
* IoT Application

### Chatbots

### Node.js offers exclusive advanced functionalities for real-time chatbots and chat applications. The key features such as multi-user applications, intensive data, heavy traffic, across all devices are very popular in chatbots. Node.js hits the sweet spot, for it runs across different devices and covers all paradigms in chat and chatbot apps. Node.js makes it super easy to execute push notifications and server-side event loops popularly used in IMs and real-time apps.

### Data Streaming

Giants like Netflix welcomed Node.js to serve 190 countries and 100+ million hours of streaming across 120 million users and counting. Node.js has a definite edge on this as the processing files at a super-fast speed, and feather-light encodes and uploads. This has been encashed upon by many online fashion e-stores and applications with massive data streaming videos.

### IoT Application Development IoT has drastically become popular over the recent decade and ever since then Node.js is one of the most preferred solutions for organizations looking to build public and private IoT solution systems.

### The very core advantage of Node.js is its brilliant ability to process seamlessly concurrent requests with thousands of events released by billions of devices on its network.Thanks to Node.js event-driven server architecture and asynchronous processing relevant heavy I/O operations on IoT networks. Furthermore, Node.js operates on writable and readable channels and streams, which is why it is the most suitable platform for IoT app development.

WHY CONSIDERING NODE?

**Scalability:**  
Node.js applications are easily scalable, vertically and horizontal. While vertical scaling lets you add more resources to the current nodes, horizontal scaling enables you to add new nodes quicker. Furthermore, Node.js applications do not require a large block in the entire development process, for it works along with a set of microservices and modules. It’s as simple, easy, and fits perfectly for startups that are looking to grow.

**High Performance:**  
Node.js runs on Google V8 JS engine that complies with Javascript code directly into the machine code. This makes it resource efficient as well as accelerates the framework’s speed. In fact, Google themselves invest heavily in their search engine to consistently improve their performance.

**Large Community Support:**  
Established in 2015, Node.js Foundation is an impressive group of founders from leading enterprises such as IBM, Microsoft, Fidelity, and SAP. An enormous number of software developers and many active Node.js fans are constantly contributing to the ever-evolving and nurturing Node.js community to support the fellow developers. Additionally, Node.js is supported on Github which makes it super easy and fast to develop and support the framework.

**Boosts Development Speed:**  
Node.js allows you developers to accelerate the web app development process with its lightweight and varied tools. The npm registry offers numerous solutions, modules, and libraries to create web apps by acquiring more value and investing in a lesser effort. Node.js, along with npm libraries, helps you reduce bugs and decrease the size of your web application with its reusable templates. This optimization directly influences the time to market.

**INTRODUCTION TO VISUAL STUDIO:**

**Visual Studio Code** is an IDE developed by Microsoft for [Windows,](https://en.wikipedia.org/wiki/Windows) [Linux](https://en.wikipedia.org/wiki/Linux) and macOS. It includes support for debugging, embedded Git control and GitHub, syntax highlighting, intelligent code completion, snippets, and code refactoring. It is highly customizable, allowing users to change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality. The source code is free and open source and released under the permissive MIT License.

Visual Studio Code is based on Electron, a framework which is used to deploy Node.js applications for the desktop running on the Blink layout engine. Although it uses the Electron framework, the software does not use Atom and instead employs the same editor component (codenamed "Monaco") used in Azure DevOps (formerly called Visual Studio Online and Visual Studio Team Services).

In the Stack Overflow 2019 Developer Survey, Visual Studio Code was ranked the most popular developer environment tool, with 50.7% of 87,317 respondents claiming to use it.

**Evolution of Visual Studio:**

The first version of VS(Visual Studio) was released in 1997, named as Visual Studio 97 having version number 5.0. The latest version of Visual Studio is 15.0 which was released on March 7, 2017. It is also termed as Visual Studio 2017. The supported .Net Framework Versions in latest Visual Studio is 3.5 to 4.7. Java was supported in old versions of Visual Studio but in the latest version doesn’t provide any support for Java language.

There are 3 editions of Microsoft Visual Studio as follows:

**1. Community:** It is a **free** version which is announced in 2014. All other editions are paid. This contains the features similar to Professional edition. Using this edition, any individual developer can develop their own free or paid apps like .Net applications, Web applications and many more. In an enterprise organization, this edition has some limitations. For example, if your organization have more than 250 PCs and having annual revenue greater than $1 Million(US Dollars) then you are not permitted to use this edition. In a non-enterprise organization, up to five users can use this edition. Its main purpose is to provide the Ecosystem(Access to thousands of extensions) and Languages(You can code in C#, VB, F#, C++, HTML, JavaScript, Python, etc.) support.

**2. Professional:** It is the commercial edition of Visual Studio. It comes in Visual Studio 2010 and later versions. It provides the support for XML and XSLT editing and includes the tool like Server Explorer and integration with Microsoft SQL Server. Microsoft provides a free trial of this edition and after the trial period, the user has to pay to continue using it. Its main purpose is to provide Flexibility(Professional developer tools for building any application type), Productivity(Powerful features such as CodeLens improve your team’s productivity), Collaboration(Agile project planning tools, charts, etc.) and Subscriber benefits like Microsoft software, plus Azure, Pluralsight, etc.

At its heart, Visual Studio Code features a lightning fast source code editor, perfect for day-to-day use. With support for hundreds of languages, VS Code helps you be instantly productive with syntax highlighting, bracket-matching, auto-indentation, box-selection, snippets, and more. Intuitive keyboard shortcuts, easy customization and community-contributed keyboard shortcut mappings let you navigate your code with ease.

For serious coding, you'll often benefit from tools with more code understanding than just blocks of text. Visual Studio Code includes built-in support for IntelliSense code completion, rich semantic code understanding and navigation, and code refactoring.

And when the coding gets tough, the tough get debugging. Debugging is often the one feature that developers miss most in a leaner coding experience, so we made it happen. Visual Studio Code includes an interactive debugger, so you can step through source code, inspect variables, view call stacks, and execute commands in the console. VS Code also integrates with build and scripting tools to perform common tasks making everyday workflows faster.

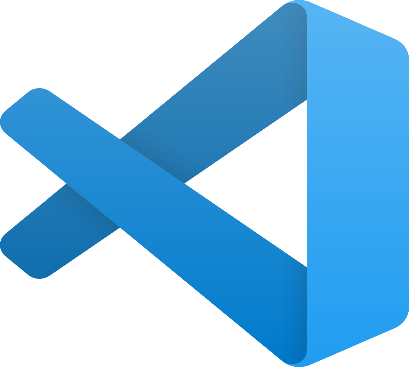


Figure 6.1 Visual Studio Code

**CHAPTER 6**

**TESTING AND DEBUGGING**

Testing is a process of finding bugs or errors in a software product that is done manually by tester or can be automated. Debugging is a process of fixing the bugs found in testing phase. Programmer or developer is responsible for debugging and it can't be automated.

Testing means verifying correct behavior. Testing can be done at all stages of module development: requirements analysis, interface design, algorithm design, implementation, and integration with other modules. In the following, attention will be directed at implementation testing. Implementation testing is not restricted to execution testing. An implementation can also be tested using correctness proofs, code tracing, and peer reviews, as described below.

Debugging is a cyclic activity involving execution testing and code correction. The testing that is done during debugging has a different aim than final module testing. Final module testing aims to demonstrate correctness, whereas testing during debugging is primarily aimed at locating errors. This difference has a significant effect on the choice of testing strategies.

Table 9.1 Testing and Debugging

|  |  |
| --- | --- |
| **Testing** | **Debugging** |
| Performed by testers | Performed by developer or development team |
| Can be done manually or Automatically | Can only be done manually |
| Can be predefined when starting testing. The test result could be  Predicted | Start with unknown conditions and it is hard to predict the result |
| Find the programming failure | Demonstrate that it**’**s only an unattended small mistake |
| Could be done automatically by using automation testing tools | Automatic debugging of software is still a dream of programmers |

**UNIT TESTING** is a level of software testing where individual units/ components of a software are tested. The purpose is to validate that each unit of the software performs as designed. A unit is the smallest testable part of any software. It usually has one or a few inputs and usually a single output. In procedural programming, a unit may be an individual program, function, procedure, etc. In object- oriented programming, the smallest unit is a method, which may belong to a base/ super class, abstract class or derived/ child class. (Some treat a module of an application as a unit. This is to be discouraged as there will probably be many individual units within that module.) Unit testing frameworks, drivers, stubs, and mock/ fake objects are used to assist in unit testing.

#### Unit Testing Method

It is performed by using the White Box Testing method.

#### When is it performed?

Unit Testing is the first level of software testing and is performed prior to Integration Testing.

#### Who performs it?

It is normally performed by software developers themselves or their peers. In rare cases, it may also be performed by independent software testers.

#### Unit Testing Tasks

Unit Test Plan:-

* + - * Prepare
      * Review
      * Rework
      * Baseline

Unit Test Cases/Scripts

* + - * Prepare
      * Review
      * Rework
      * Baseline

Unit Test

* + - * Perform

**INTEGRATION TESTING** is a level of software testing where individual units are combined and tested as a group. The purpose of this level of testing is to expose faults in the interaction between integrated units. Test drivers and test stubs are used to assist in Integration Testing.

Integration Testing is the second level of testing performed after Unit Testing and before System Testing. Developers themselves or independent testers perform Integration Testing.

#### Method

Any of Black Box Testing, White Box Testing and Gray Box Testing methods can be used. Normally, method depends on your definition of ‘unit’.

#### Tasks

Integration Test Plan:-

* + - * Prepare
      * Review
      * Rework
      * Baseline

Integration Test Cases/Scripts:-

* + - * Prepare
      * Review
      * Rework
      * Baseline

Integration Test:-

* + - * Perform

#### Analogy

During the process of manufacturing a ballpoint pen, the cap, the body, the tail and clip, the ink cartridge and the ballpoint are produced separately and unit tested separately. When two or more units are ready, they are assembled and Integration Testing is performed. For example, whether the cap fits into the body or not.

**SYSTEM TESTING** is a level of software testing where a complete and integrated software is tested. The purpose of this test is to evaluate the system’s compliance with the specified requirements.

System Testing is the third level of software testing performed after Integration Testing and before Acceptance Testing. Normally, independent Testers perform System Testing.

#### Method

Usually, Black Box Testing method is used.

#### Tasks

System Test Plan:-

* + - * Prepare
      * Review
      * Rework
      * Baseline

System Test Cases

* + - * Prepare
      * Review
      * Rework
      * Baseline

System Test

* + - * Perform

#### Analogy

During the process of manufacturing a ballpoint pen, the cap, the body, the tail, the ink cartridge and the ballpoint are produced separately and unit tested separately. When two or more units are ready, they are assembled and Integration Testing is performed. When the complete pen is integrated, System Testing is performed.

**ACCEPTANCE TESTING** is a level of software testing where a system is tested for acceptability. The purpose of this test is to evaluate the system’s compliance with the business requirements and assess whether it is acceptable for delivery. Acceptance Testing is the fourth and last level of software testing performed after System Testing and before making the system available for actual use.

* *Internal Acceptance Testing* (Also known as Alpha Testing) is performed by members of the organization that developed the software but who are not directly involved in the project (Development or Testing). Usually, it is the members of Product Management, Sales and/or Customer Support.
* *External Acceptance Testing* is performed by people who are not employees of the organization that developed the software.
  + *Customer Acceptance Testing* is performed by the customers of the organization that developed the software. They are the ones who asked the organization to develop the software. [This is in the case of the software not being owned by the organization that developed it.]
  + *User Acceptance Testing* (Also known as Beta Testing) is performed by the end users of the software. They can be the customers themselves or the customers’ customers.

#### Method

Usually, Black Box Testing method is used in Acceptance Testing. Testing does not normally follow a strict procedure and is not scripted but is rather ad-hoc.

#### Tasks

* + - * Acceptance Test Plan
        + Prepare
        + Review
        + Rework
        + Baseline
      * Acceptance Test Cases/Checklist
        + Prepare
        + Review
        + Rework

**CHAPTER 7**

**CODING AND IMPLEMENTATION**

### Implementation and Conversion Plan

All the designing and alignment is implemented with the help of code editor i.e. Visual Studio Code by creating a projects and defining the components , building them in their component modules.

**CODE:**

<html>

<head>

<link rel="preconnect" href="https://fonts.gstatic.com">

<link href="https://fonts.googleapis.com/css2?family=Slabo+27px&display=swap" rel="stylesheet">

<link rel="preconnect" href="https://fonts.gstatic.com">

<link href="https://fonts.googleapis.com/css2?family=Advent+Pro&family=Slabo+27px&display=swap" rel="stylesheet">

<title>My Home Page</title>

<link rel="stylesheet" type="text/css" href="HomePage.css">

</head>

<body>

<div class="navbar">

<div class="logo"><p>WORK'n'APPOINT</p></div>

<ul class="test">

<li class="list">

<button class="btn"><a href="SignIn.html">Login</a></button>

</li>

<li class="list">

<button class="btn">Sign Up</button>

</li>

<li class="list"><img src="person.png" class="t5">

<a href="#footer\_anchor" class="five" id="con">Contact Us</a>

</li>

<li class="list"><img src="about.png" class="t5">

<a href="#about" class="five">About</a>

</li>

<li class="list"><img src="find2.jpg" class="t5">

<a href="#contact" class="five">Find Jobs</a>

</li>

</ul>

</div>

<div class="parent">

<div class="sidebar">

<h2>SKILLS</h2>

<ul>

<li><a href="#">Html</a></li>

<li><a href="#">Cascading style sheets</a></li>

<li><a href="#">Javascript</a></li>

<li><a href="#">Python</a></li>

<li><a href="#">Cloud Computing</a></li>

<li><a href="#">Data Structures</a></li>

<li><a href="#">Machine Learning</a></li>

<li><a href="#">C</a></li>

<li><a href="#">C++</a></li>

<li><a href="#">Java</a></li>

<li><a href="#">Django</a></li>

<li><a href="#">Flutter</a></li>

<li><a href="#">Mean</a></li>

<li><a href="#">Mern Stack</a></li>

<li><a href="#">PHP</a></li>

<li><a href="#">MySql</a></li>

<li><a href="#">AutoCad</a></li>

<li><a href="#">Salesforce</a></li>

<li><a href="#">Blockchains</a></li>

<li><a href="#">Automation</a></li>

</ul>

</div>

<div class="mid">

</div>

<div class="right-sidebar">

</div>

</div>

<!--footer-->

<div class="footer">

<div class="footer-content">

<a name="footer\_anchor"></a>

<div class="footer-section contact-us">

<h2 class="logo-text">Contact Us</h2>

<p class="para">9306228235</p>

<p class="para">kaur.26parneet@gmail.com</p>

</div>

<div class="footer-section links">

<h2 class="logo-text">Links</h2>

<ul class="new">

<a href="#"><li class="list1">Events</li></a>

<a href="#"><li class="list1">Team</li></a>

<a href="#"><li class="list1">Mentors</li></a>

<a href="#"><li class="list1">Terms and Conditions</li></a>

</ul>

</div>

</div>

<div class="footer-bottom">

&copy;worknappoint.com | Designed By Rahul, Pankaj, Parneet

</div>

</div>

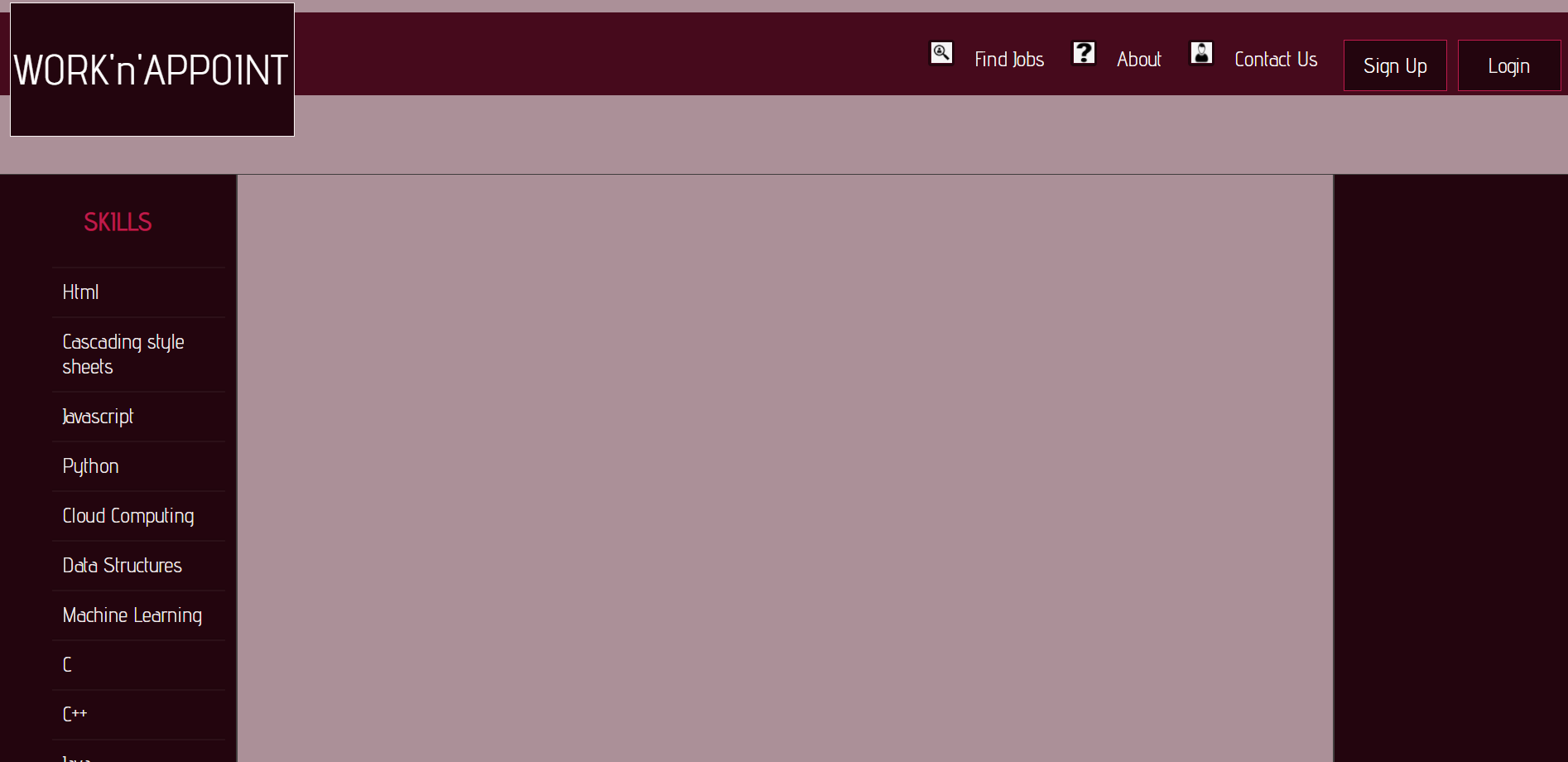
</body>

<script>

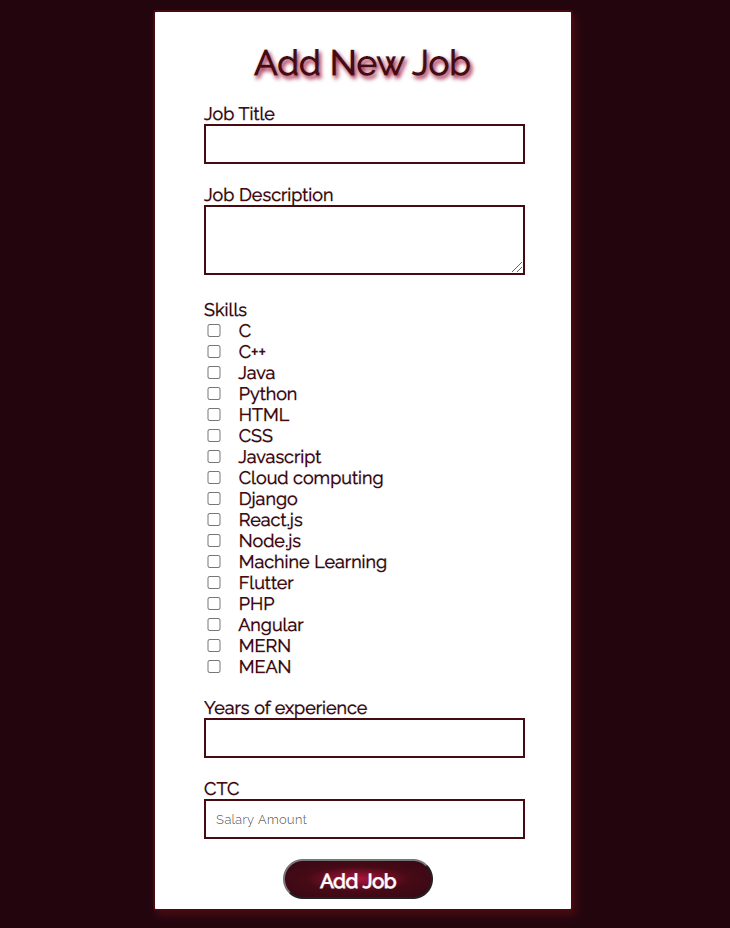
</script>

</html>

**SNAPSHOTS**







## 

## 

## 

## C:\Users\Lenovo\Pictures\1.3.PNG

## C:\Users\Lenovo\Pictures\1.4.PNG

## 

## ADVANTAGES:

## 1. Branding opportunity for employers All successful company should be utilising their career site to project a consistent brand and company image/values to prospective job seekers. In what is now a very competitive, candidate driven market the top talent can be very particular about whom they work for and a companies career site can often serve as a basis for their application decisions. Candidates; especially the best ones crave information, they want to know about your company and it’s culture so they can make an informed decision and your Career Site is the best place for you to control this flow of information and sell them on your organisation.

**2. Reduced cost-of-hire** Online job postings can range from £150 to approximately £500 per job on the more niche job sites plus the additional cost of CV data base search functionality. This is considerably more cost-effective than the 20% of an annual salary fee that many traditional recruiters charge or the costs of newspaper/publication ads for the same reach and time period, however recruiting direct through your company career site can reduce this cost yet again by around 35%.

## 3. Reduced time-to-hire

On line recruitment allows for immediate, real-time interaction and 24×7 hiring and job search activity. Employers can post a job direct to their Career Site in as little as 5 minutes and start receiving CVs in response immediately. Traditional methods where a newspaper ad may take a week to appear and then only appear for a day, or a HR Manager has to wait till month-end to see the benefits of a costly ad in a monthly industry or geographic specific publication can increase the time and cost of recruiting the best talent. Recruiting the top talent through an agency also takes time. From shortlisting your chosen suppliers to meeting them and finally giving one the opportunity to work on a role can take up to to two weeks. Utilising your own Career Site is on average 70% faster than traditional hiring methods as you the employer are in full control. You can contact candidates immediately and directly and you do not have to wait for a middleman to sift through, filter, assess or select the top talent. By being in the driving seat you gain valuable insight into the nature of the marketplace and the competitive landscape for the position. You are also able to ensure a superior match and a better fit for your organisation in the long term.

## 4. Allows for database build-up and tracking.

As previously mentioned the cost of utilising a CV database on any of the nations many job boards can be very costly and is an exercise that will need repeating at every hire.

Your own Career Site will give you, the employer the opportunity to save high profile or particularly attractive CVs to build a priority database of pre-screened top talent for future use. This will in turn reduce cost and time per hire. To asses return on investment you should be monitoring the interaction with every vacancy you post. This can be done on your Career site with metrics that will allow you to track everything from page views to search engine keywords, you need to understand what content is driving the best results for your company in terms of the top candidates and hires.

## 5. Company Research

## Before and after you apply for a job, it's a good idea to research the company. This will help you avoid companies with bad reputations and learn about the company to help you put your best foot forward during an interview. Visit a company's website and go deep inside to learn everything you can about the business to better target your cover letter and prepare for an interview. Visit other websites that provide information on the company, such as employee reviews or business articles about the management, products and sales.

**FUTURE SCOPE:**

Job portals are like the meeting points for the recruiters as well as the job seekers. Most of the jobseekers who're actively seeking new employment opportunities are believed to be registered on multiple job portals. Finding a job opportunity per your choice and qualification through a job portal is relatively easier. But the sourcing and operations process often takes a long time. Many use an employment agency to expedite this process. Recruitment firms usually do not charge a single penny from the candidates. But due to the ongoing Situation, most of the placement agencies are currently out of business. Many placement agencies are bringing their services online. In the short term, we will likely see a wave of new job portals, but just as in most markets, this initial influx will see various winners and losers. This will anyway benefit the job seekers.

The future is and will remain unknown to us , but fact is that, till there is a issue of unemployment in the world of job market, the job portals will exist and will grow in proportionate with demand .

**CONCLUSION:**

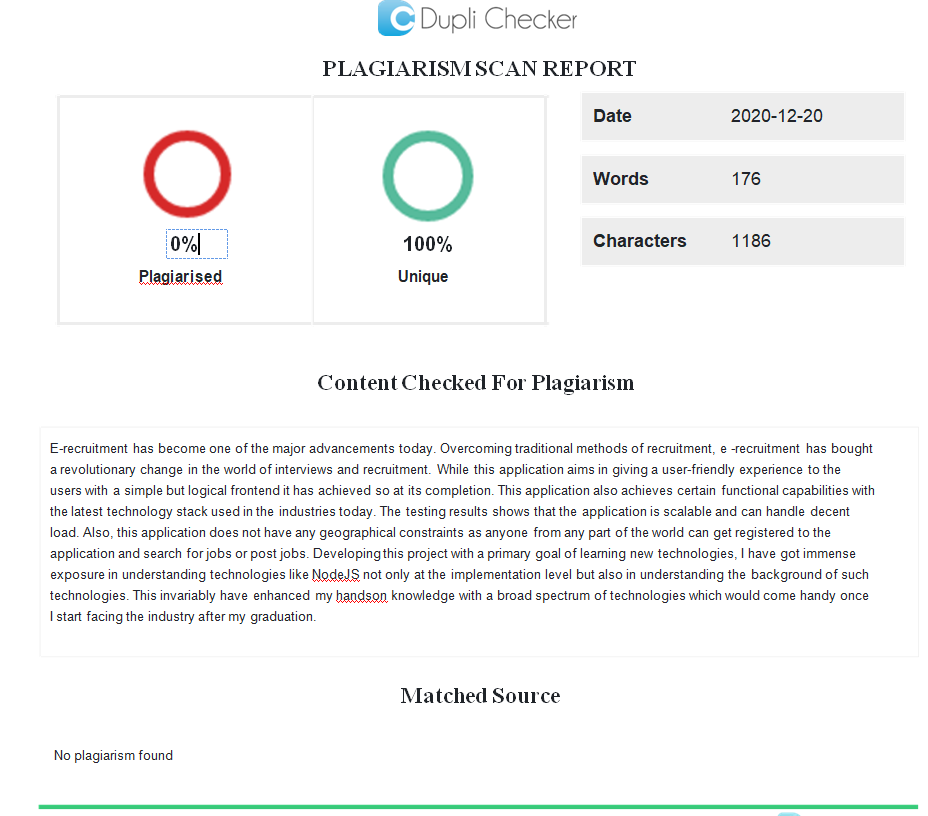
E-recruitment is one of the major advancements of the job industry today. Overcoming traditional methods of recruitment, e -recruitment has bought a revolutionary change in the world of interviews and recruitment. While this application aims in giving a user-friendly experience to the users with a simple but logical frontend it has achieved so at its completion. This application also achieves certain functional capabilities with the latest technology stack used in the industries today. The testing results shows that the application is scalable and can handle decent load. Also, this application does not have any geographical constraints as anyone from any part of the world can get registered to the application and search for jobs or post jobs. Developing this project with a primary goal of learning new technologies, I have got immense exposure in understanding technologies like NodeJS not only at the implementation level but also in understanding the background of such technologies.

This invariably have enhanced my handson knowledge with a broad spectrum of technologies which would come handy once I start facing the industry after my graduation.

**BIBLIOGRAPHY:**

1. <https://javascript.io/>
2. <https://Nodejs.io/>
3. htt[ps://www.duc](http://www.ducatindia.com/)a[tindia.com](http://www.ducatindia.com/)
4. https://colorlib.com › templates

**PLAGERISM:**

****

****

****

****