Project Progress Report on

CV GENERATING WEB APP

Submitted in fulfillment of the requirement for the award of the degree of

BACHELOR OF COMPUTER APPLICATION

Submitted By:

Ankit Kumar Thakur **2102144**

Under the Guidance of

Mr. Aditya Joshi

Assistant Professor

Department of Computer Application

Department of Computer Application

Graphic Era (Deemed to be University)

Dehradun, Uttarakhand

Jun-2024

CANDIDATE'S DECLARATION

I hereby certify that the work which is being presented in the Project progress report entitled **"CV Generating web app"** in partial fulfillment of the requirements for the award of the Degree of Bachelor of Computer Applications in the Department of Computer Application of the Graphic Era (Deemed to be University), Dehradun shall be carried out by the undersigned under the supervision of **Mr. Aditya Joshi, Assistant Professor,** Department of Computer Application, Graphic Era (Deemed to be University), Dehradun.

Ankit Kumar Thakur 2102144 Sign.

The above mentioned student shall be working under the supervision of the undersigned on the **"CV Generating web app"**

Supervisor **Head of the Department**

Examination

Name of the Examiners: **Signature with Date**

1.

2.

Work Satisfactory: Yes/ No

1.2 Problem Statement

The problem Statement for the present work can be stated as follows:

As stated above that, in the highly competitive job market, creating a well-structured and visually appealing Curriculum Vitae (CV) is essential for job seekers to effectively showcase their skills, qualifications, and experience. Despite the importance of a professional CV, many individuals struggle with the task due to a lack of design skills, unfamiliarity with industry standards, and the time-consuming nature of the process. This often results in CVs that fail to capture the attention of employers, diminishing the job seekers' chances of securing interviews and job offers.

Figure 1.1 Representation of common CV getting rejected

Traditional methods of CV creation, such as using word processors or relying on outdated templates, do not adequately address these challenges. They often require significant manual effort to format and customize, leading to inconsistencies and errors. Additionally, these methods do not provide real-time feedback or suggestions for improvement, further complicating the task for users who are unsure of best practices in CV writing.[3]

4.4 MODULES AND LIBRARIES

The Modules that we would be using in building the model are:

1. **HTML-** HTML, or HyperText Markup Language, is the standard language used to create and design documents on the web. It structures web content by using elements and tags to define the layout and format of text, images, links, and other multimedia. HTML is a cornerstone technology of the World Wide Web, alongside CSS (Cascading Style Sheets) and JavaScript.

HTML can embed programs written in a scripting language such as JavaScript, which affects the behavior and content of web pages. The inclusion of CSS defines the look and layout of content. [10]

Elements and Tags: HTML uses “elements” to mark up the structure and to define content. Elements are denoted by “tags,” which label pieces of content such as “heading,” “paragraph,” “table,” and so on.

Attributes: Elements can have attributes that define additional characteristics or provide metadata. For example, the href attribute of an a tag sets the URL for a hyperlink.

Document Object Model (DOM): The DOM represents the document as a tree structure where each node is an object representing a part of the document.

HTML5: The latest evolution of HTML, providing new elements and APIs for more complex web applications.

Semantic Elements: Elements like <article>, <footer>, <header>, and <section> provide better document structure and are used to define the different parts of a web page.

2. **CSS-** CSS, or Cascading Style Sheets, is a stylesheet language used to describe the presentation and design of HTML documents. It controls the layout, colors, fonts, and overall visual appearance of web pages, allowing developers to separate content (HTML) from presentation (CSS)

CSS is designed to enable the separation of content and presentation, including layout, colors, and fonts. This separation can improve content accessibility;[further explanation needed] 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; and enable the .css file to be cached to improve the page load speed between the pages that share the file and its formatting. [11]

Key Features of CSS:

- **Selectors:**

  CSS selectors are used to target HTML elements that you want to style. Common types of selectors include:

  - Element Selector: Targets elements by their tag name (e.g., p { }).
  - Class Selector: Targets elements by their class attribute (e.g., .classname { }).
  - ID Selector: Targets elements by their id attribute (e.g., #idname { }).

- Attribute Selector: Targets elements by their attributes (e.g., [type="text"] { }).

- **Properties and Values:**

  CSS uses properties and values to define the styles for elements. For example:

```css
p {
    color: blue;
    font-size: 16px;
    margin: 10px;
}
```

Figure 4.1.2 Describing CSS property

In this example, color, font-size, and margin are properties, and blue, 16px, and 10px are their respective values.

- **Cascading and Specificity:**

The term "cascading" in CSS means that styles can cascade from one stylesheet to another, allowing multiple stylesheets to be combined. CSS follows a specific order of precedence:

- Inline styles (inside an HTML element) have the highest precedence.
- Internal styles (within a <style> tag in the HTML document) have medium precedence.
- External styles (linked from an external CSS file) have the lowest precedence.
- Specificity determines which styles are applied when there are conflicting rules. Inline styles have the highest specificity, followed by IDs, classes, and finally element selectors.

3. **JavaScript-** JavaScript is a versatile, high-level programming language that is widely used for creating interactive and dynamic web pages. Initially developed by Netscape in 1995, JavaScript has evolved into one of the core technologies of web development, alongside HTML and CSS. It enables developers to enhance the user experience by allowing web pages to respond to user actions, manipulate the DOM (Document Object Model), and communicate asynchronously with servers.

JavaScript (/ˈdʒɑːvəskrɪpt/), often abbreviated as JS, is a programming language and core technology of the Web, alongside HTML and CSS. 99% of websites use JavaScript on the client side for webpage behavior. JavaScript is a high-level, often just-in-time compiled language that conforms to the ECMAScript standard.[12] It has dynamic typing, prototype-based object-orientation, and first-class functions. It is multi-paradigm, supporting 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).[13]

A majority of websites use it and all major web browsers have a dedicated JavaScript engine to execute it. JavaScript supports event-driven, functional, and imperative programming styles, and includes APIs for working with text, arrays, dates, regular expressions, and basic manipulation of the Document Object Model (DOM).

- **Event Handling:**

  JavaScript handles user interactions through events. Event listeners can be added to elements to execute code in response to user actions like clicks, key presses, and form submissions.

- **Asynchronous Programming:**

  Asynchronous operations are crucial for web applications, allowing tasks like data fetching, timers, and I/O operations to run without blocking the main thread. Promises and async/await syntax simplify handling asynchronous code.

- **Modules:**

  ES6 introduced modules, allowing code to be split into reusable pieces. The import and export keywords enable the use of modules, promoting better organization and maintainability of code.

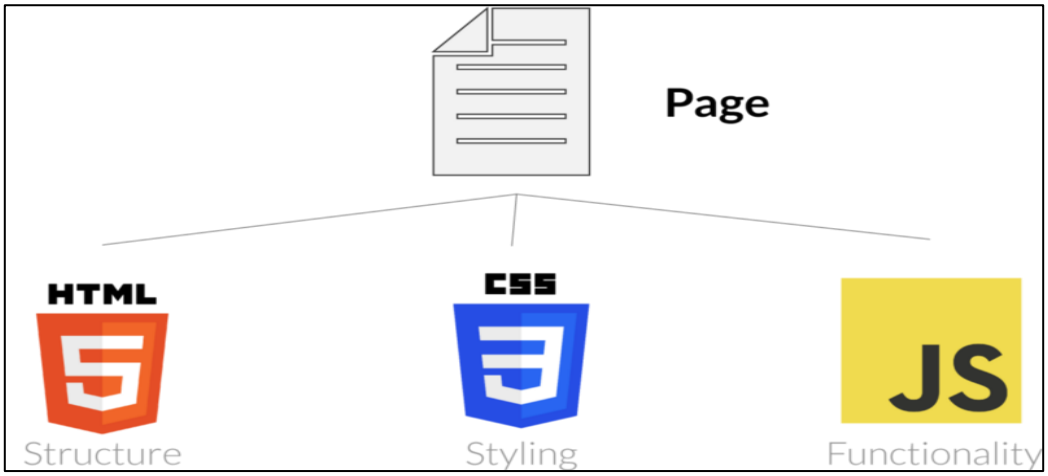


Fig 4.1.4 html,css and js.

4. **BootsStrap** - Bootstrap is a popular open-source front-end framework that simplifies the development of responsive and visually appealing web applications. Developed by Twitter and released in 2011, Bootstrap provides a collection of CSS and JavaScript tools designed to make the process of creating web pages faster and more efficient. It is widely used for its ease of use, flexibility, and ability to create consistent layouts across different browsers and devices.

Bootstrap (formerly Twitter Bootstrap) is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains HTML, CSS and (optionally) JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.

As of May 2023, Bootstrap is the 17th most starred project (4th most starred library) on GitHub, with over 164,000 stars.[14] According to W3Techs, Bootstrap is used by 19.2% of all websites.

Bootstrap is an HTML, CSS and JS library that focuses on simplifying the development of informative web pages (as opposed to web applications). 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.

4.9 Model Architecture:

The Architecture and proposed flowchart for building the Web Application is shown in form of flowchart diagram below.

This Basic circuit Diagram shows the flow data and exact execution of Web Application:

The basic blocks show the modules and data used and arrow reflect the flow of process →

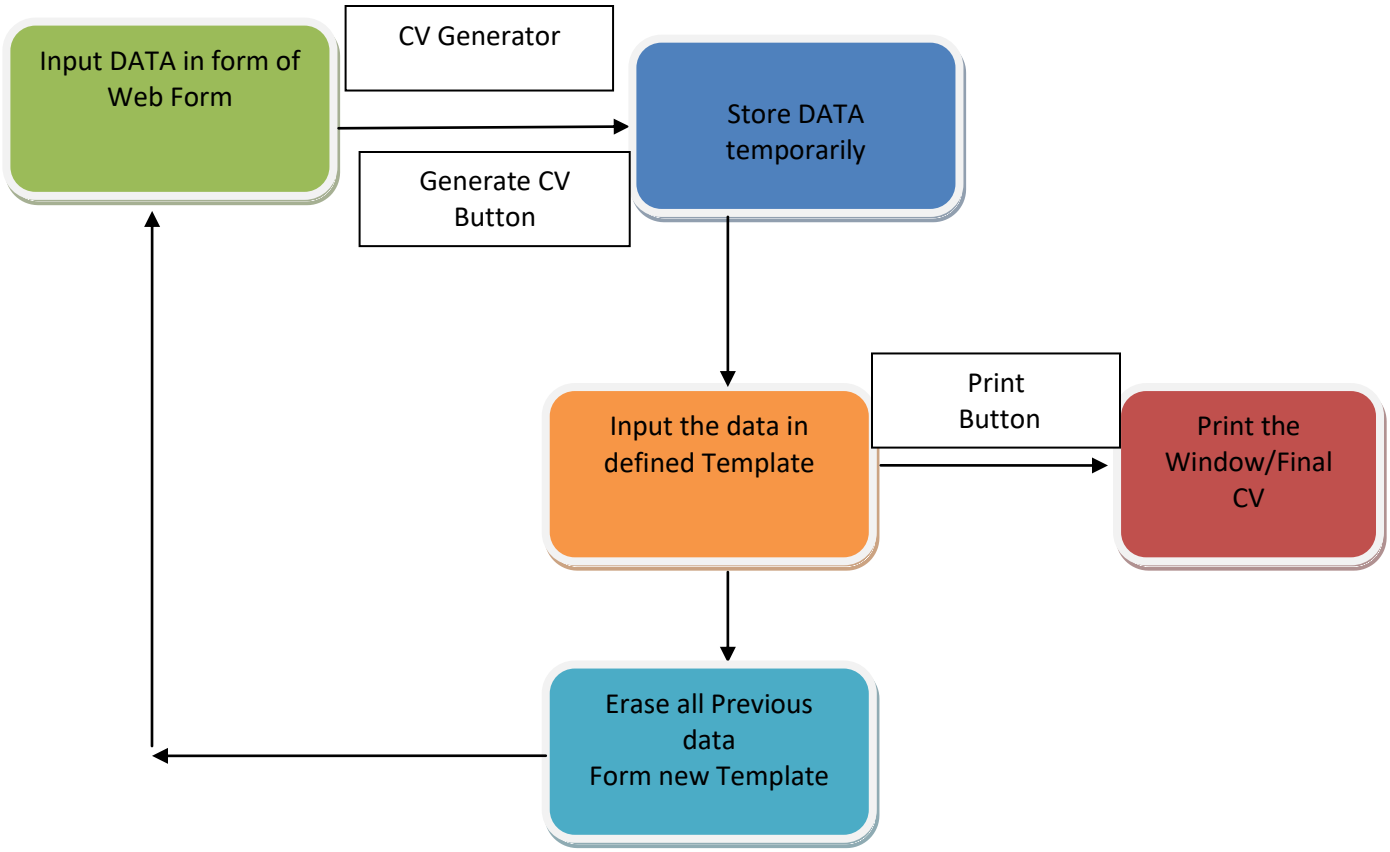


Figure 4.9.1 Flow Diagram for CV Generator

- Additional Sections: Grants, fellowships, certifications, skills, languages, and any other relevant information.

The Resumes are concise and targeted, suitable for most job applications, focusing on relevant skills and experiences. CVs, on the other hand, are detailed and comprehensive, ideal for academic, research, and scientific positions, providing an in-depth look at your entire career. Understanding these differences ensures that you present your qualifications in the best possible light, depending on the job or position you are applying for.

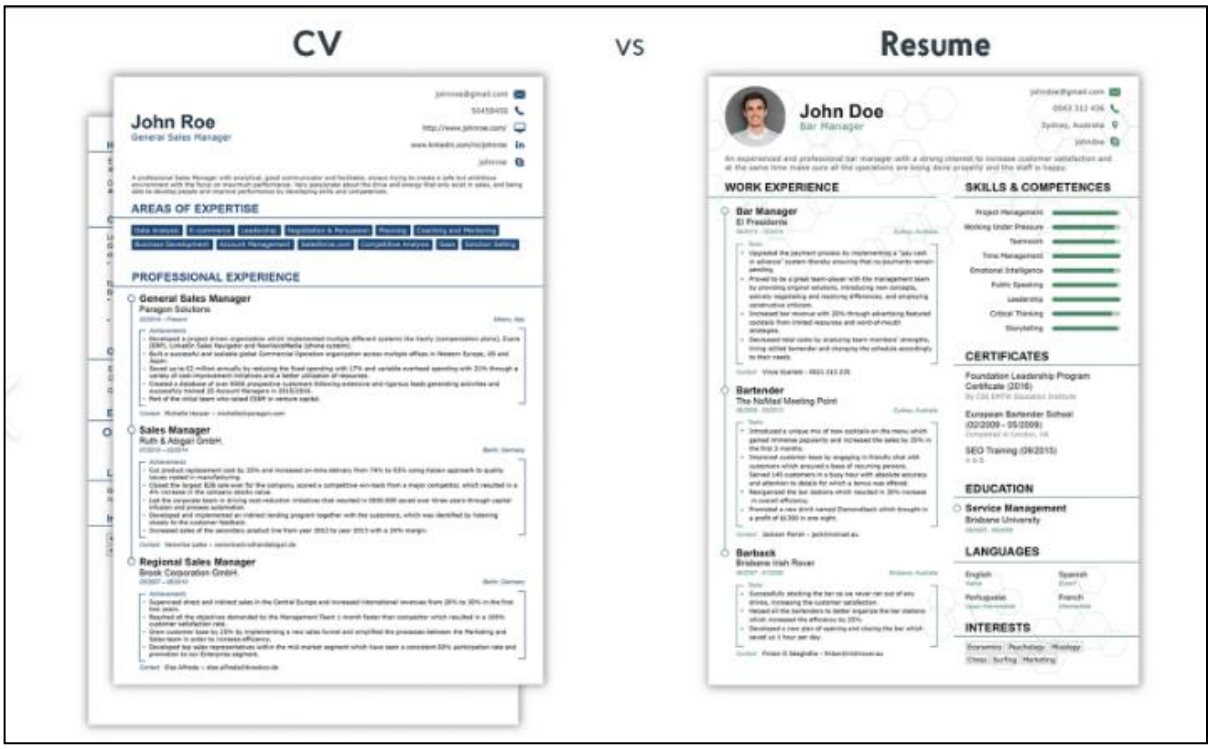


Figure4.10.1 Visual Difference between CV and Resume

available for documents and introduces markup and application programming interfaces (APIs) for complex web applications.

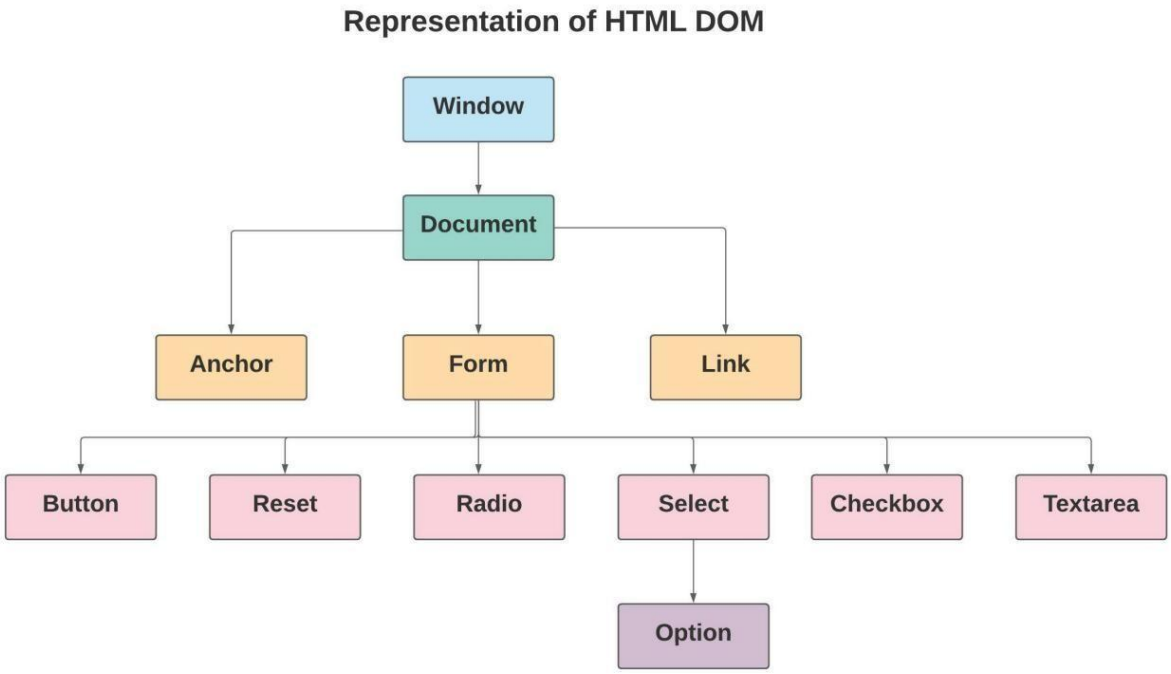


Fig 5.1 HTML Tree

5.1.2 CSS3

To style the captivating world of “CV generator,” we’ve utilized CSS3, the latest standard for cascading style sheets. CSS3 offers advanced styling capabilities with minimal effort, such as rounded corners, shadows, gradients, transitions, and animations, which are essential for creating a visually appealing interface. With media queries, we’ve ensured that “CV generator” is responsive and provides an optimal viewing experience across a range of devices, from desktops to mobile phones.

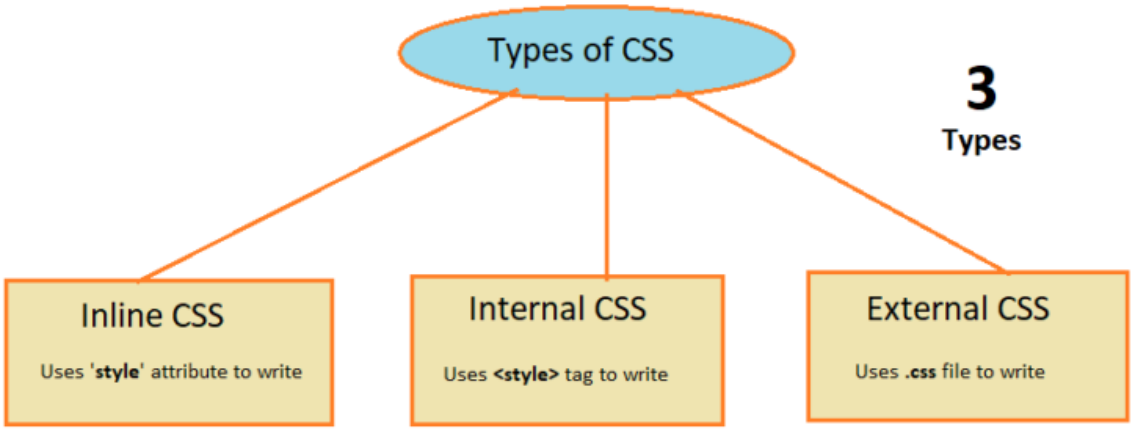


Fig 5.2 CSS types

CSS, or Cascading Style Sheets, is the language for describing the presentation of Web pages, including colors, layout, and fonts, allowing one to adapt the presentation to different types of devices, such as large screens, small screens, or printers. CSS is independent of HTML and can be used with any XML-based markup language. The separation of HTML from CSS makes it easier to maintain sites, share style sheets across pages, and tailor pages to different environments. This is referred to as the separation of structure (or: content) from presentation. CSS has a simple syntax and uses a number of English keywords to specify the names of various style properties. Each rule or rule-set consists of one or more selectors and a declaration block. In CSS, selectors declare which part of the markup a style applies to by matching tags and attributes in the markup itself. Elements in the document tree are matched against selectors when the user agent (browser) applies the style sheet to the document tree. Selectors may apply to all elements of a specific type, or only those elements that match a

5.1.3 JavaScript

JavaScript breathes life into "CV generator," turning static pages into a dynamic and interactive experience. It's a multi-paradigm language that supports event-driven, functional, and imperative programming styles, enabling us to implement complex features like the "Guess the Pokémon" game and real-time updates in the Template. JavaScript's versatility allows us to handle both client-side and server-side scripting, making "CV generator" an engaging platform for trainers worldwide.

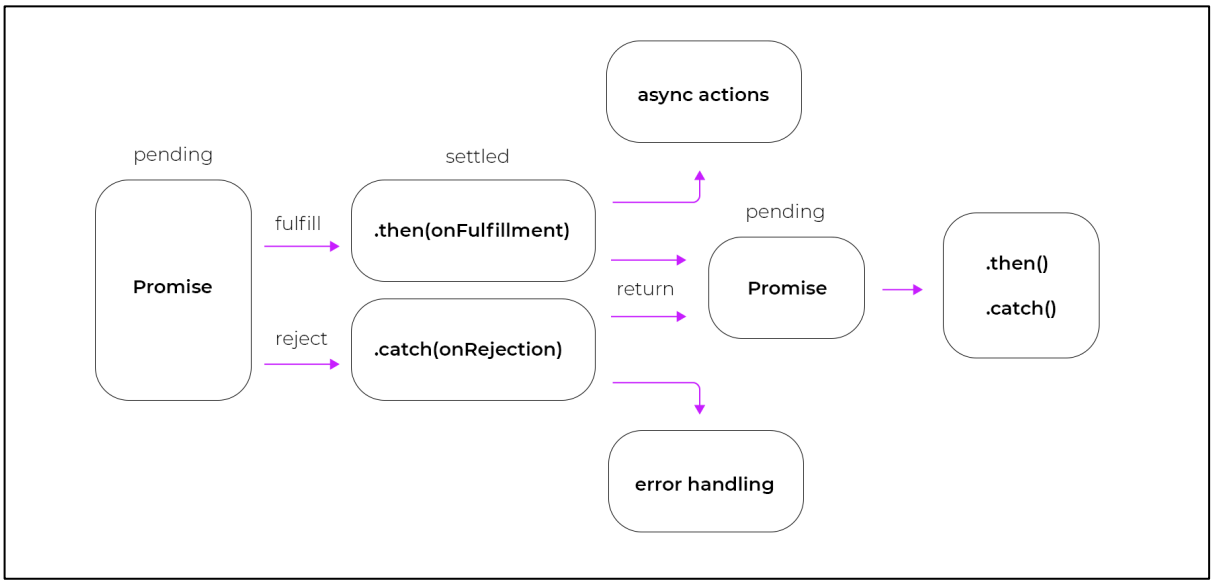


Fig 5.3 Async-await in javascipt

JavaScript, often abbreviated as JS, is a high-level, interpreted scripting language that conforms to the ECMAScript specification. JavaScript has become one of the core technologies of the World Wide Web, alongside HTML and CSS, and is supported by all modern web browsers without the need for plugins. Its capabilities allow developers to build dynamic and interactive web applications, making it an essential part of web development. JavaScript enables client-side script to interact with the user, control the browser, communicate asynchronously, and alter the document content that is displayed. As a multi-paradigm language, JavaScript supports event-driven, functional, and imperative programming styles, and it has APIs for working with text, arrays, dates, regular expressions,

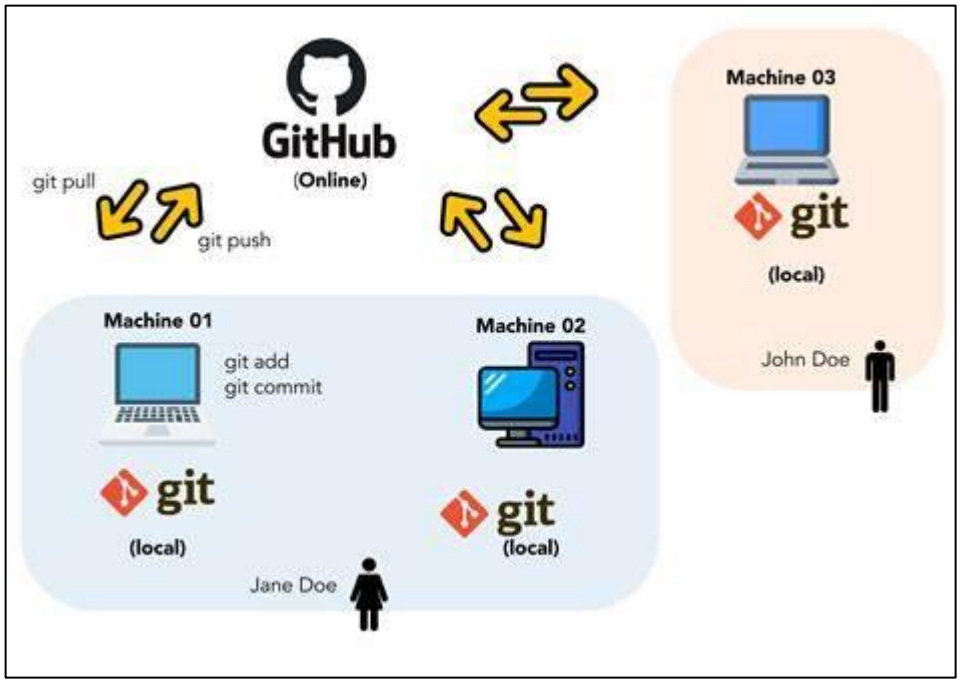


Fig 5.4 Git and Github Flow

5.1.5 GitHub:

GitHub acts as our repository hosting service, leveraging Git's version control and adding its own features. It provides a web-based graphical interface and access control, along with several collaboration features such as bug tracking, feature requests, task management, and wikis for every project. GitHub has been instrumental in "CV generator" development, allowing us to manage our code, track issues, and collaborate with contributors from around the globe.

5.2.6 Visual Studio Code (VSCode): Our development environment of choice for "CV generator" is VSCode, a powerful and lightweight code editor that's free and open-source. It offers features like IntelliSense for smart completions, built-in Git commands, debugging tools, and an extensive marketplace for extensions. VSCode's flexibility and customization options have made it an indispensable tool for writing, testing, and debugging our game's code. Visual Studio Code (VS Code) is a lightweight yet powerful source code editor that runs on Windows, macOS, and Linux. It combines the simplicity

of a code editor with robust developer tooling, making it an ideal choice for both beginners and seasoned developers. Here's why VS Code stands out.

Fig 5.5 VS Code logo

Fast and Efficient: VS Code's lightning-fast source code editor ensures a frictionless edit-build-debug cycle. It gets out of your way, allowing you to focus on your ideas rather than environment setup.

Language Support: With built-in support for hundreds of languages, VS Code provides syntax highlighting, bracket matching, auto-indentation, and snippets. It adapts to your coding needs seamlessly.

IntelliSense and Code Navigation: IntelliSense offers smart code completion, rich semantic understanding, and navigation. You'll find it easier to explore and understand your codebase.

Debugging Made Easy: VS Code includes an interactive debugger. You can step through source code, inspect variables, view call stacks, and execute commands in the console—all within the editor.

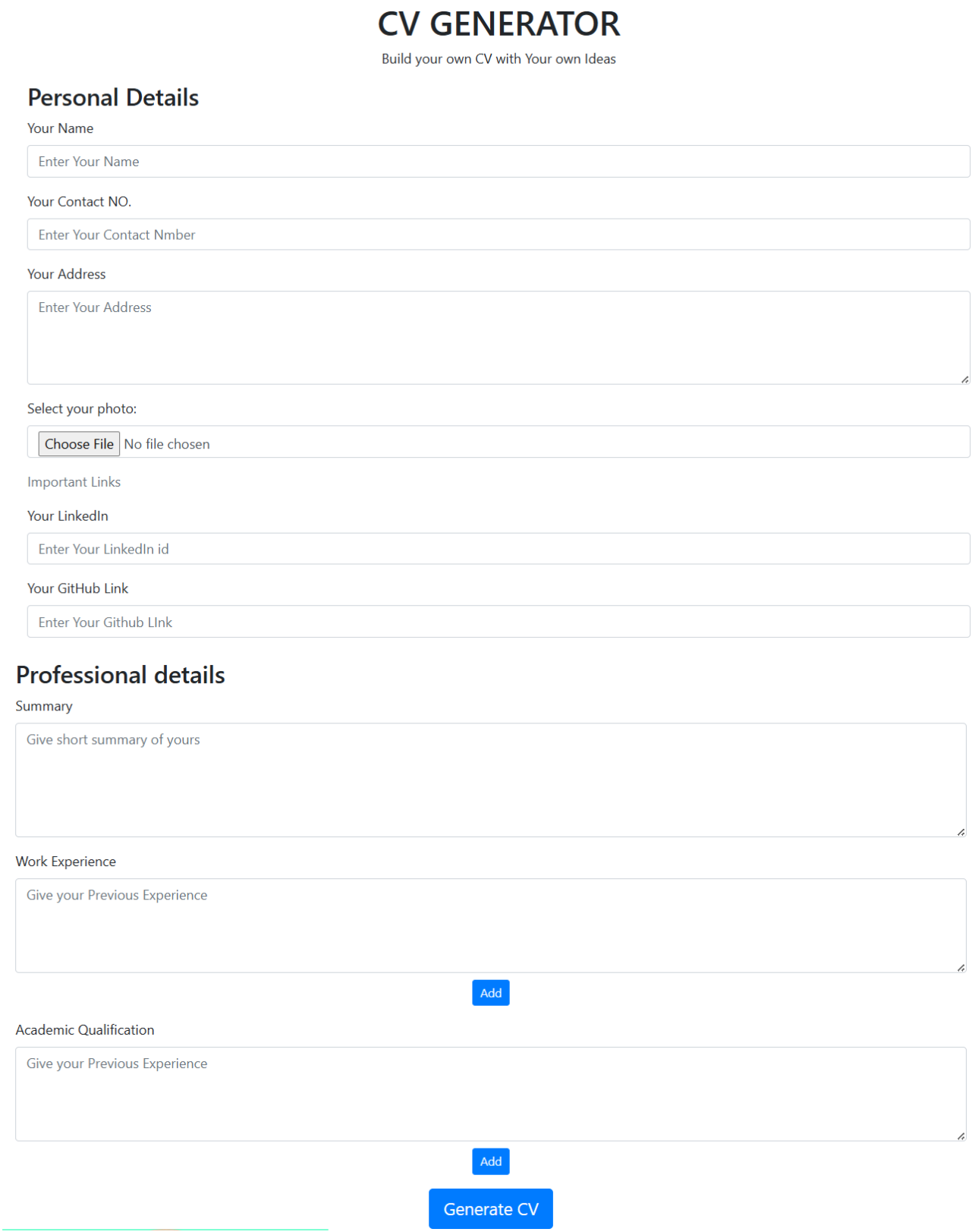
A. Building a simple HTML form that will take input from user:

```
<body>
    <!-- Starting HTML body -->
    <div class="container" id="cv-form">

        <h1 class="text-center my-2">CV GENERATOR</h1>
        <p class="text-center ">Build your own CV with Your own Ideas</p>

        <div class="row-md-6">
            <div class="clo-md-6">
                <!-- first col -->
                <h3>Personal Details</h3>

                <div class="form-group">
                    <label for="nameFiled">Your Name</label>
                    <input type="text" id="nameField" placeholder="Enter Your Name" class="form-control">
                </div>

                <div class="form-group mt-2">
                    <label for="contactFiled">Your Contact NO.</label>
                    <input type="text" id="contactField" placeholder="Enter Your Contact Nmber" class="form-control">
                </div>

                <div class="form-group mt-2">
                    <label for="addressFiled">Your Address</label>
                    <textarea id="addressField" rows="4" placeholder="Enter Your Address"
                        class="form-control"></textarea>
                </div>

                <div class="form-group mt-3">
                    <label for="">Select your photo:</label>
                    <input id="imgField" type="file" class="form-control" />
                </div>

                <p class="text-secondary my-3">Important Links</p>

                <div class="form-group mt-2">
                    <label for="ldFiled">Your LinkedIn</label>
                    <input type="text" id="ldField" placeholder="Enter Your LinkedIn id" class="form-control">
                </div>

                <div class="form-group mt-2">
                    <label for="ghFiled">Your GitHub Link</label>
                    <input type="text" id="ghField" placeholder="Enter Your Github LInk" class="form-control">
```

```
        </div>
        <div class="clo-md-6">
            <!-- second col -->
            <h3>Professional details</h3>

            <div class="form-group mt-2">
                <label for="sumFiled">Summary</label>
                <textarea rows="5" id="sumField" placeholder="Give short summary of yours"
                    class="form-control"></textarea>
            </div>

            <div class="form-group mt-2" id="we">
                <label for="">Work Experience</label>
                <textarea rows="4" placeholder="Give your Previous Experience"
                    class="form-control weField"></textarea>
                <div class="container text-center mt-2" id="weAddButton">
                    <button onclick="addNewWEField()" class="btn btn-primary btn-sm">Add</button>
                </div>
            </div>

            <div class="form-group mt-2" id="aq">
                <label for="">Academic Qualification</label>
                <textarea rows="4" placeholder="Give your Previous Experience"
                    class="form-control eqField"></textarea>
                <div class="container text-center mt-2" id="aqAddButton">
                    <button onclick="addNewAQField()" class="btn btn-primary btn-sm">Add</button>
                </div>
            </div>

        </div>

    </div>
    <div class="container text-center mt-3">
        <button onclick="generateCV()" class="btn btn-primary btn-lg">Generate CV</button>
    </div>
</div>
```

Figure 5.6 HTML form for input

OUTPUT:


CV GENERATOR
Build your own CV with Your own Ideas
Personal Details
Your Name
Enter Your Name
Your Contact NO.
Enter Your Contact Nmber
Your Address
Enter Your Address
Select your photo:
Choose File
No file chosen
Important Links
Your LinkedIn
Enter Your LinkedIn id
Your GitHub Link
Enter Your Github LInk
Professional details
Summary
Give short summary of yours
Work Experience
Give your Previous Experience
Add
Academic Qualification
Give your Previous Experience
Add
Generate CV

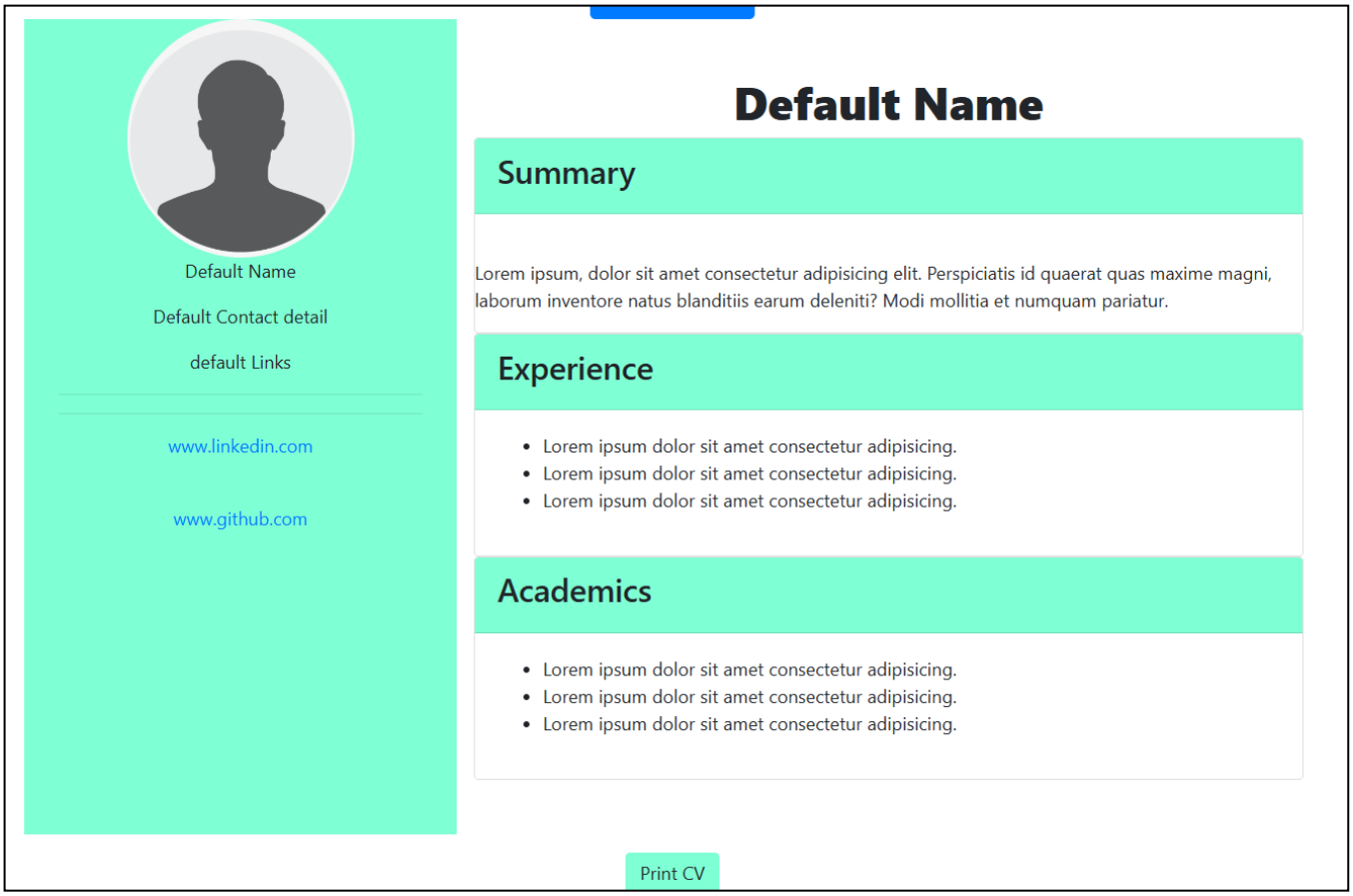
B.HTML scripting for CV template:

```
<!-- print Template -->
<div class="container" id="cv-template">
    <div class="row">
        <div class="col-md-4 text-center background">
            <!-- first col -->
            <img src="img/ank.jpeg" alt="" id="imgT" class="img-fluid myimg" />
            <div class="container">
                <p id="nameT1">Default Name</p>
                <p id="contactT">Default Contact detail</p>
                <p id="addressT">default Links</p>
                <hr>
                <hr>
                <p><a id="ldT" href="#1">www.linkedin.com</a></p><br>
                <p><a id="ghT" href="#1">www.github.com</a></p>
            </div>
        </div>
        <div class="col-md-8 py-5">
            <!-- second col -->
            <h1 id="nameT2" class="text-center" style="font-weight: 900;">Default Name</h1>

            <!-- summary -->
            <div class="card nt-4">
                <div class="card-header background">
                    <h3>Summary</h3>
                </div>
                <div class="card-body"> </div>
                <p id="summaryT">Lorem ipsum, dolor sit amet consectetur adipisicing elit. Perspiciatis id quaerat
                    quas maxime
                    magni, laborum inventore natus blanditiis earum deleniti? Modi mollitia et numquam pariatur.
                </p>
            </div>

            <!-- Experienxe -->
            <div class="card nt-4">
                <div class="card-header background">
                    <h3>Experience</h3>
                </div>
                <div class="card-body">
                    <ul id="weT">
                        <li>Lorem ipsum dolor sit amet consectetur adipisicing.</li>
                        <li>Lorem ipsum dolor sit amet consectetur adipisicing.</li>
                        <li>Lorem ipsum dolor sit amet consectetur adipisicing.</li>
                    </ul>
                </div>
            </div>
```

```
            <!-- academic Qulaification -->
            <div class="card nt-4">
                <div class="card-header background">
                    <h3>Academics</h3>
                </div>
                <div class="card-body">
                    <ul id="aqT">
                        <li>Lorem ipsum dolor sit amet consectetur adipisicing.</li>
                        <li>Lorem ipsum dolor sit amet consectetur adipisicing.</li>
                        <li>Lorem ipsum dolor sit amet consectetur adipisicing.</li>
                    </ul>
                </div>
            </div>
        </div>

        <div class="container mt-3 text-center">
            <button onclick="printCV()" class="btn background">
                Print CV
            </button>
        </div>
    </div>

</div>
</div>
```

Figure 5.7 HTML for CV template

OUTPUT:


Default Name
Default Contact detail
default Links
www.linkedin.com
www.github.com
Default Name
Summary
Lorem ipsum, dolor sit amet consectetur adipisicing elit. Perspiciatis id quaerat quas maxime magni, laborum inventore natus blanditiis earum deleniti? Modi mollitia et numquam pariatur.
Experience
Lorem ipsum dolor sit amet consectetur adipisicing.
Lorem ipsum dolor sit amet consectetur adipisicing.
Lorem ipsum dolor sit amet consectetur adipisicing.
Academics
Lorem ipsum dolor sit amet consectetur adipisicing.
Lorem ipsum dolor sit amet consectetur adipisicing.
Lorem ipsum dolor sit amet consectetur adipisicing.
Print CV

C. Java Script code:

```
// All JS function in here
function addNewWEField(){
    // console.log("Adding new field");

    let newNode=document.createElement("textarea");
    newNode.classList.add("form-control");
    newNode.classList.add("weField");
    newNode.classList.add("mt-2");

    newNode.setAttribute("rows",3);
    newNode.setAttribute("placeholder","Enter Here");

    let weOb = document.getElementById("we");
    let weAddButtonOb = document.getElementById("weAddButton");

    weOb.insertBefore(newNode, weAddButtonOb);

}

function addNewAQField(){

    let newNode=document.createElement("textarea");
    newNode.classList.add("form-control");
    newNode.classList.add("eqField");
    newNode.classList.add("mt-2");

    newNode.setAttribute("rows",3);
    newNode.setAttribute("placeholder","Enter Here");

    let aqOb = document.getElementById("aq");
    let aqAddButtonOb = document.getElementById("aqAddButton");

    aqOb.insertBefore(newNode, aqAddButtonOb);

}
```

```
// generate CV

function generateCV(){
    // console.log("going");
    let nameField=document.getElementById("nameField").value;

    let nameT1 = document.getElementById("nameT1");

    nameT1.innerHTML=nameField;


    document.getElementById("nameT2").innerHTML=nameField;

    document.getElementById("contactT").innerHTML=document.getElementById("contactField").value;

    document.getElementById("addressT").innerHTML=document.getElementById("addressField").value;
    document.getElementById("ldT").innerHTML=document.getElementById("ldField").value;
    document.getElementById("ghT").innerHTML=document.getElementById("ghField").value;

    document.getElementById("summaryT").innerHTML=document.getElementById("sumField").value;

    // Work experienxe
    let wes=document.getElementsByClassName('weField');
    let str="";
    for (let e of wes){
        str=str+`<li> ${e.value} </li>`;
    }

    document.getElementById("weT").innerHTML = str;

    // academic qualifiaction
    let aqs=document.getElementsByClassName('eqField');
    let str1="";
    for(let e of aqs){
        str1+=`<li> ${e.value} </li`;
    }

    document.getElementById("aqT").innerHTML=str1;
```

```
    // code for image

    let file = document.getElementById('imgField').files(0);
    console.log(file);
    let reader = new FileReader();
    reader.readAsDataURL(file);
    console.log(reader.result);


    // set image
    reader.onloadend=function(){

        document.getElementById('imgT').src=reader.result;
    }

    document.getElementById("cv-form").style.display="none";
    document.getElementById("cv-template").style.display="block";

}

// print CV
function printCV(){
    window.print();
}
```

Figure 5.8 JavaScript code for functioning

Chapter 6 Result and Output:

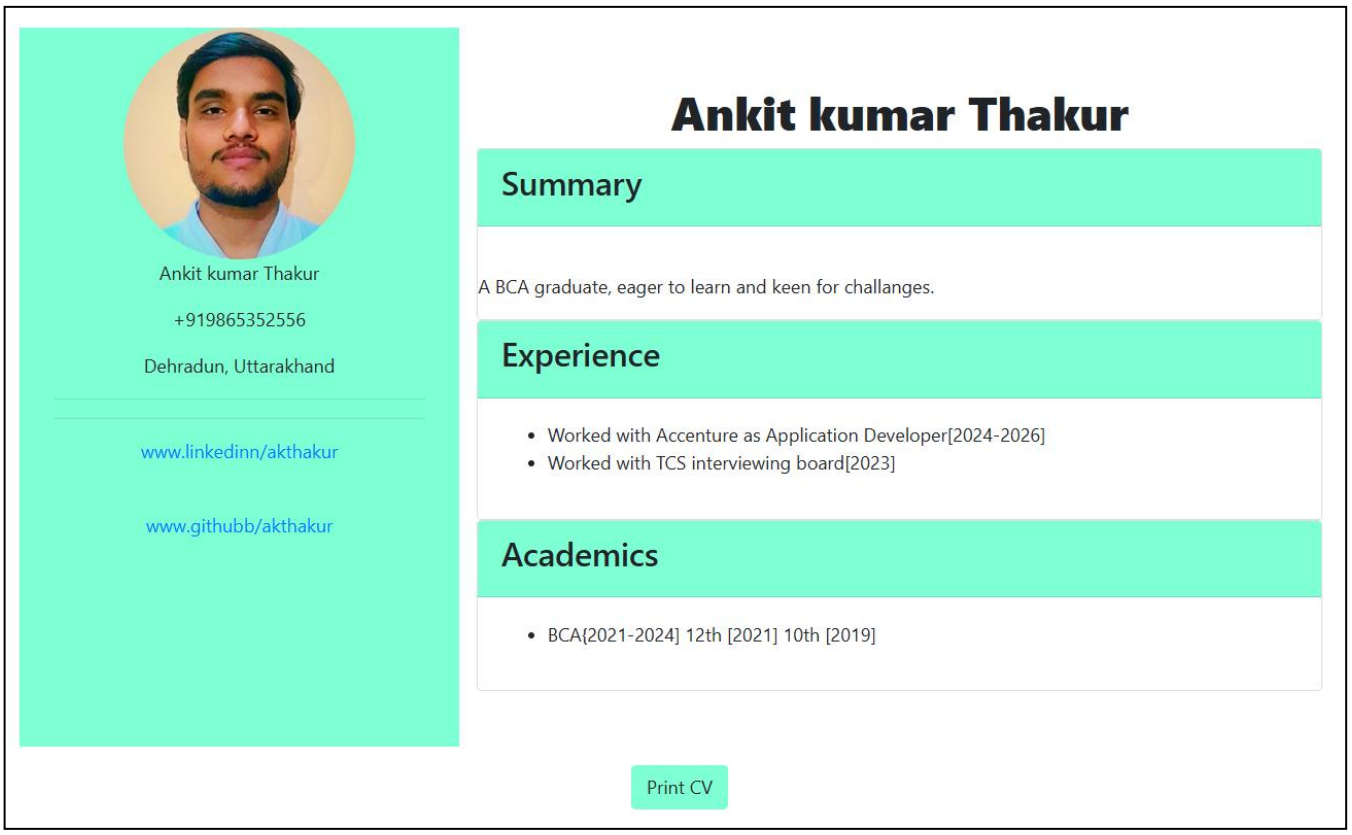
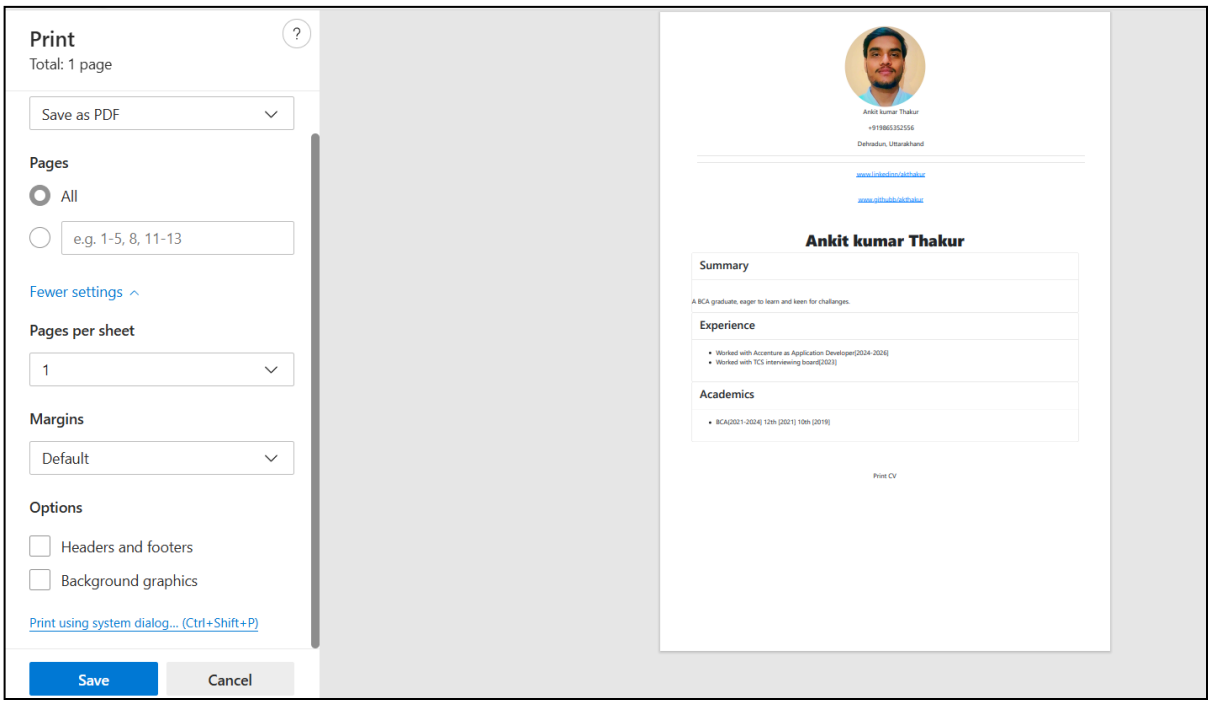
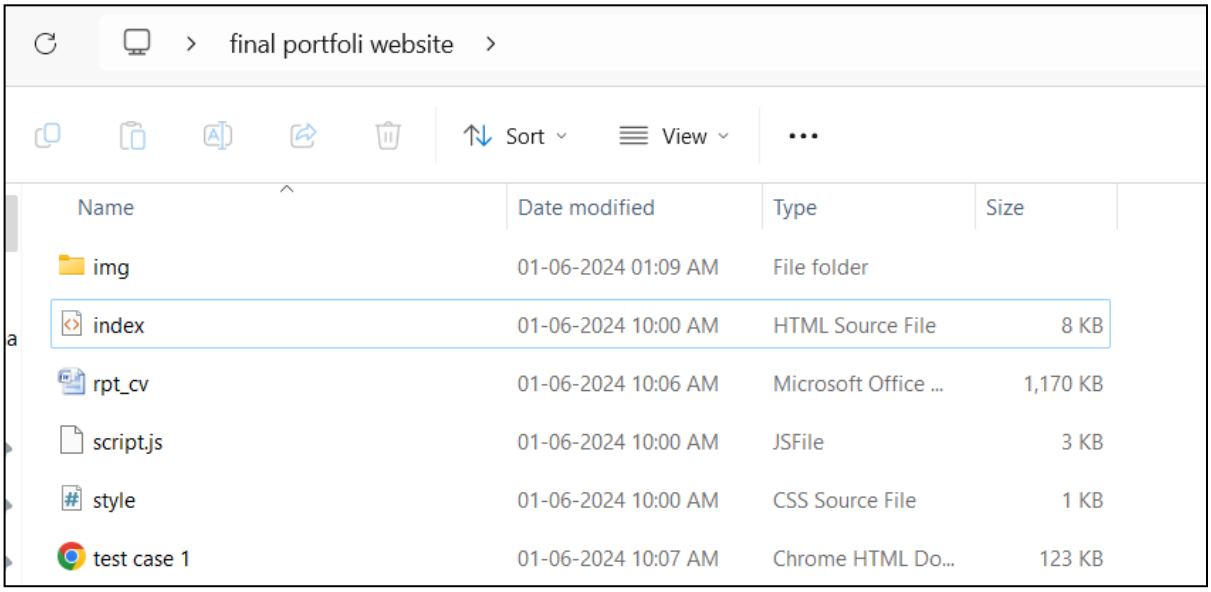


Figure 6 Test case 1

After pressing PrintCV button the following CV is printed/ saved in pdf format



CV is saved in same folder with source code in pdf format.



test case 1.pdf

%20portfoli%20website/test%20case%201.pdf

YouTube Maps Downloads Deloitte Consulting...

Ask Copilot 1 of 1

Ankit kumar Thakur

+919865352556

Dehradun, Uttarakhand

www.linkedinn/akthakur

www.githubb/akthakur

Ankit kumar Thakur

Summary

A BCA graduate, eager to learn and keen for challanges.

Experience

- Worked with Accenture as Application Developer[2024-2026]
- Worked with TCS interviewing board[2023]

Academics

- BCA(2021-2024] 12th [2021] 10th [2019]