

A Report on
Resume Builder application
Submitted for partial fulfillment of award of
DEGREE
OF
BACHELOR OF COMPUTER APPLICATIONS

Submitted By
Nishant Singh: 230201020
Amisha Pal: 230201004
Ankush Singh: 230201005
Shahid: 230201030

Under the supervision of
Ashutosh Mishra
Destination
(Haridwar University, Roorkee)



DEPARTMENT OF BCA
Haridwar University, Roorkee
Batch: 2023 - 2026

Certificate

This is to Certify that Nishant Singh has carried out the project work presented in this report entitled Resume Builder for the award of Bachelor Of Computer Applications from Haridwar University under my supervision. The report embodies result of original work and studies carried out by Student himself and the contents of the report do not form the basis for the award of any other degree to the candidate or to anybody else.

Date: 26-Mar-2025

Ashutosh Mishra

Destination: Haridwar University

Acknowledgement

Nishant Singh: - 230201020

Amisha Pal: - 230201004

Ankush Singh: - 230201005

Shahid: - 230201030

We would like to express our sincere gratitude to **BRAVERS** for their invaluable guidance and support throughout the development of this Resume Builder Web Application. Their insights and expertise have been instrumental in shaping our project.

We also extend our appreciation to our peers and family for their continuous encouragement and motivation. Their support has played a crucial role in helping us stay focused and committed to our goals.

Lastly, we acknowledge the resources, tools, and technologies that enabled us to successfully build this application. This project would not have been possible without the knowledge and support we received along the way.

Abstract

We have developed a [Resume Builder Web Application](#) that simplifies the process of creating professional, high quality, ATS friendly resumes. This application allows users to input their details, choose from various templates, and generate a well-structured resume effortlessly. It is designed to be user friendly, ensuring that anyone, (Student, jobseekers, working professional) can craft a polished and ATS friendly impressive resume within minutes.

Signature: Nishant

Table of Contents

1. Introduction	(Page 7-10)
1.1 Overview & Problem Statement	
1.2 Purpose	
1.3 Scope	
1.4 Tools Used	
1.5 Methodology Used	
1.6 Technology Used	
2. System Analysis	(Page 11-12)
2.1 Identification of Need	
2.2 Preliminary Investigation	
2.3 Feasibility Study	
3. Means of Project	(Page 13-13)
3.1 Hardware Requirements	
3.2 Software Requirements	
4. Overall Description	(Page 14-16)
4.1 Product Perspective	
4.2 Software Interface	
4.3 Hardware Interface	
4.4 Communication Interface	
4.5 Constraints	
4.6 ER Diagram	
4.7 Data Flow Diagrams	
4.8 Database Design	
5. Specific Diagrams	(Page 17-19)
5.1 Class Diagram	

- 5.2 Use-Case Diagram
- 5.3 Activity Diagrams
- 5.4 Sequence Diagrams

6. Product Features (Page 20-24)

- 6.1 Screen Shots
 - 6.1.1 Forms
 - 6.1.2 Reports

7. Coding (Page 25-51)

8. Test Cases and Test Results (Page 52-52)

9. Conclusion (Page 53-53)

10. References (Page 56-56)

11. Future Scope of the Project (Page 54-55)

- 11.1 Further Enhancement

12. Bibliography (Page 56-56)

INTRODUCTION

1.1 Overview & Problem Statement

A resume is an essential document for job seekers, as it provides a summary of their qualifications, skills, and experiences. However, many individuals struggle with formatting and structuring their resumes effectively. The Resume Builder Web Application aims to simplify this process by providing users with a user-friendly interface to create, customize, and download their resumes in various templates. The application ensures that users generate professional and ATS-compliant resumes without requiring technical expertise.

1.2 Purpose

The primary purpose of this project is to develop a web-based platform that allows users to create resumes quickly and efficiently. It will offer pre-designed templates, an easy-to-use form interface, and automated PDF generation, ensuring convenience and accessibility for job seekers across different industries.

1.3 Scope

- User-friendly interface for inputting resume details
- Multiple templates for different job profiles
- HTML/CSS based template which make it more customizable and user friendly
- Automatic high quality PDF generation
- Secure data transmission Frontend to Backend
- No data storage system to support user privacy
- Highly prioritizing User comfort and provide high quality templates that no one other platforms provide in free tier.

1.4 Tools Used

- **Frontend:** HTML, CSS, JavaScript, React.js
- **Backend:** Flask (Python)
- **NLP Filtering/Generation:** DeepSeek-R1, SpaCy, fstring
- **PDF Generation/Zippping:** WeasyPrint, zipfile

1.5 Methodology Used

The project follows an Agile Development Model, which enables continuous iteration, feedback, and adaptability throughout the development lifecycle. This methodology ensures a collaborative and incremental approach, allowing for flexibility in response to changing requirements and user needs. The key phases involved are:

1. Requirement Analysis

This phase involves understanding the project goals, identifying user needs, and defining key application features. Activities include:

- Conducting stakeholder interviews and surveys.
- Analyzing business requirements and technical constraints.
- Defining functional and non-functional requirements.
- Establishing project scope and feasibility.
- Identifying the need for a **user-friendly interface** for resume input, multiple templates, and **secure data transmission**.

2. Design & Prototyping

Once requirements are gathered, the design phase begins to conceptualize the user experience and system architecture. This includes:

- Creating UI/UX wireframes to visualize the interface and user interactions.
- Developing low-fidelity and high-fidelity prototypes for usability testing.
- Designing database schema and structuring data relationships (if needed for temporary storage).
- Planning system architecture, ensuring **no data storage for user privacy**.
- Designing **HTML/CSS-based resume templates** to enhance customization.

3. Development

This phase focuses on implementing the application using Agile methodologies such as Scrum or Kanban. The development process involves:

- Developing backend services for **secure data transmission without storing user information**.
- Implementing frontend components based on UI/UX designs.
- Creating multiple **resume templates tailored for different job profiles**.
- Integrating **automatic high-quality PDF generation**.
- Ensuring proper documentation and version control.

4. Testing

Comprehensive testing ensures reliability, performance, and security. Various testing techniques include:

- **Unit Testing:** Verifying individual components for expected behavior.

- **Integration Testing:** Ensuring smooth interaction between frontend and backend.
- **Functional Testing:** Validating that resume templates and PDF generation work as intended.
- **User Testing:** Gathering feedback from users to refine the experience.
- **Performance & Security Testing:** Ensuring **fast, secure, and reliable PDF generation**.

5. Deployment & Maintenance

After successful testing, the application is deployed for user access. This phase includes:

- Configuring the production environment and hosting the application.
- Implementing Continuous Integration and Continuous Deployment (CI/CD) pipelines.
- Monitoring system performance and resolving potential issues.
- Releasing updates and new templates based on user feedback.
- Providing ongoing maintenance, security patches, and performance optimizations.

1.6 Technology Used

- **Frontend:** React.js for a dynamic UI
- **Backend:** Flask for handling API requests
- **Database:** PostgreSQL for storing user data securely
- **NLP:** DeepSeek-R1 for generating descriptions, SpaCy for text filtering
- **PDF Generation:** WeasyPrint for high-quality output
- **Security:** HTTPS encryption for secure data transmission

System Analysis

2.1 Identification of Need

Many job seekers struggle with designing a well-structured and visually appealing resume due to a lack of technical skills or design expertise. Existing resume builders often provide limited customization options, require subscriptions, or do not prioritize user privacy. This application addresses these gaps by offering a user-friendly, customizable, and secure resume-building experience with high-quality templates available for free.

2.2 Preliminary Investigation

A thorough analysis of existing resume builders revealed several limitations, such as:

- Lack of customization options in free-tier services.
- Poor users experience due to complex interfaces.
- Requirement of account creation and personal data storage.
- High subscription costs for accessing premium features.

To overcome these issues, our application focuses on providing a simple, intuitive interface, fully customizable HTML/CSS templates, and high-quality PDF generation without data storage, ensuring user privacy and accessibility.

2.3 Feasibility Study

Technical Feasibility

- Utilizes HTML, CSS, JavaScript, and backend frameworks for secure data transmission.
- Compatible across multiple devices and platforms.
- Implements PDF generation libraries for structured resumes.

Economic Feasibility

- Minimal infrastructure lowers costs.
- No cloud storage needed, reducing maintenance expenses.
- Free-tier accessibility for a broad user base.

Operational Feasibility

- Simple UI makes it easy to use for all users.
- Multiple resume templates for diverse job profiles.
- No sign-up required, ensuring a seamless experience.
- Fast, secure, and privacy-focused resume generation.

Means of Project

3.1 Hardware Requirements

The application is lightweight and optimized for minimal resource consumption.

- Minimum: 4GB RAM, Intel Core i3 or equivalent.
- Recommended: 8GB RAM, Intel Core i5 or higher for faster performance.
- Performance: The browser-based nature of the application ensures it consumes only 1.8MB of RAM and requires negligible CPU time, making it ideal for low-end systems.

3.2 Software Requirements

- Operating System: Windows, Linux, or macOS.
- Development Tools: VS Code, Postman for API testing.
- Database: PostgreSQL (if required for optional backend storage).
- Libraries & Frameworks:
 - Frontend: React.js for a dynamic user interface.
 - Backend: Flask for handling requests securely.
 - PDF Generation: WeasyPrint for high-quality document export.
 - NLP Processing: SpaCy for intelligent text formatting (if needed in future updates).

Overall Description

4.1 Product Perspective

The application is a **lightweight, browser-based resume builder** that eliminates the need for installation. It allows users to **create, customize, and download resumes in real-time** without storing any personal data. Unlike many competitors, it prioritizes **user privacy, high-quality templates, and accessibility in the free tier**.

4.2 Software Interface

- **Frontend:** Built with **React.js** for a dynamic and interactive user experience.
- **Backend:** Powered by **Flask API**, ensuring secure and efficient processing of resume data.
- **Database:** **PostgreSQL** (if required for storing pre-defined templates or additional functionalities).

4.3 Hardware Interface

- Designed to run smoothly on standard **desktop and laptop hardware**.
- Consumes minimal resources (**~1.8MB RAM, negligible CPU usage**) due to its **browser-based architecture**.

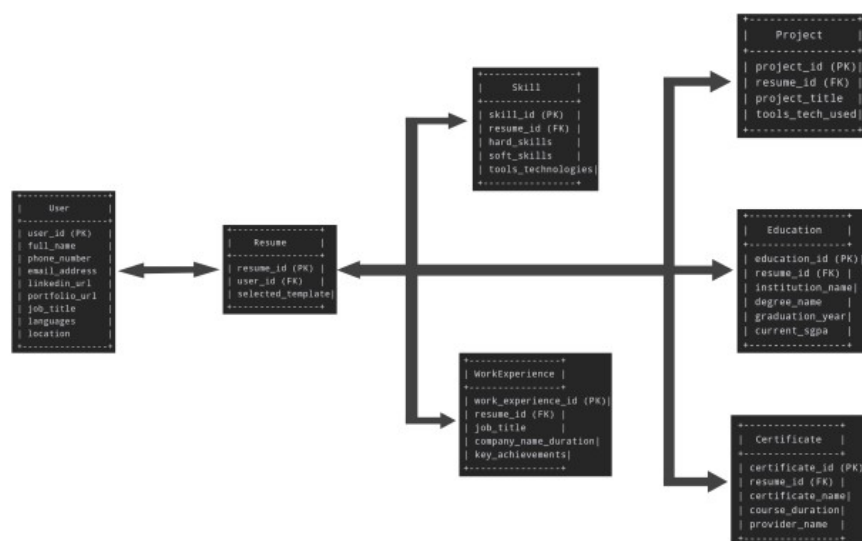
4.4 Communication Interface

- Utilizes **RESTful APIs** for seamless interaction between the frontend and backend.
- Ensures **secure data transmission** without storing user inputs.
- PDF generation handled dynamically through backend libraries like **WeasyPrint**.

4.5 Constraints

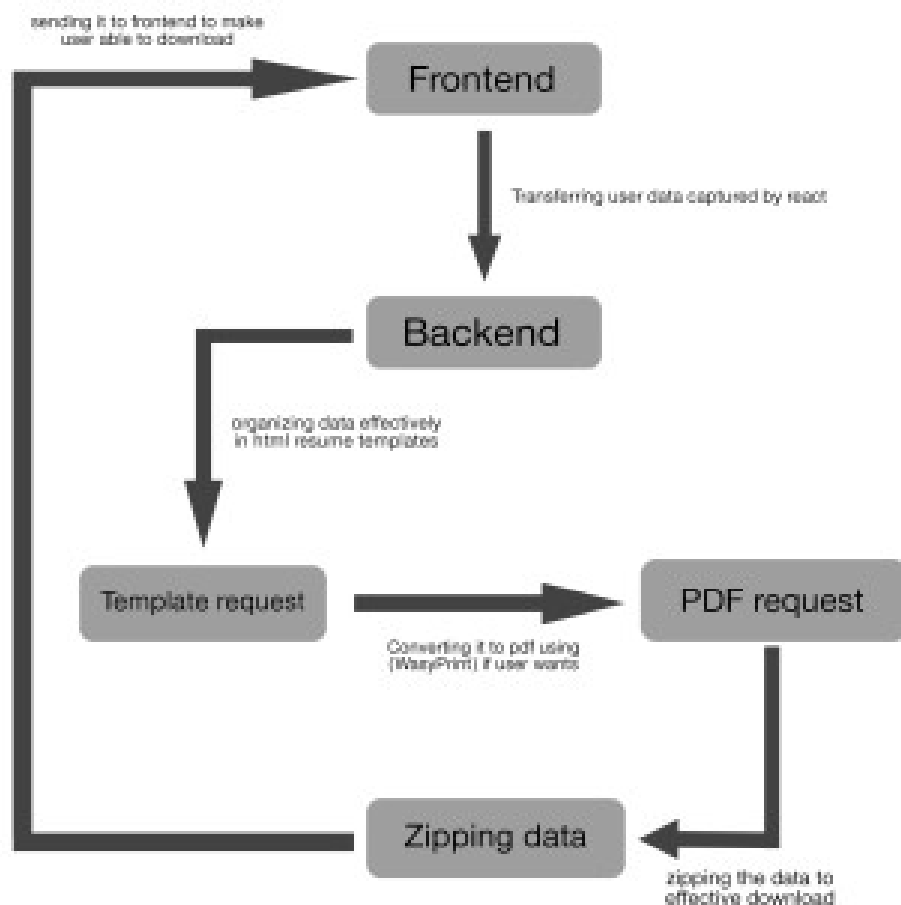
- **Internet Connection Required:** The application is web-based and requires an active connection.
- **Limited Templates (Initial Version):** Future updates will introduce more professionally designed templates.
- **No Data Persistence:** Prioritizes privacy by not storing user data after session completion.

4.6 Entity-Relationship (E-R) Diagram:



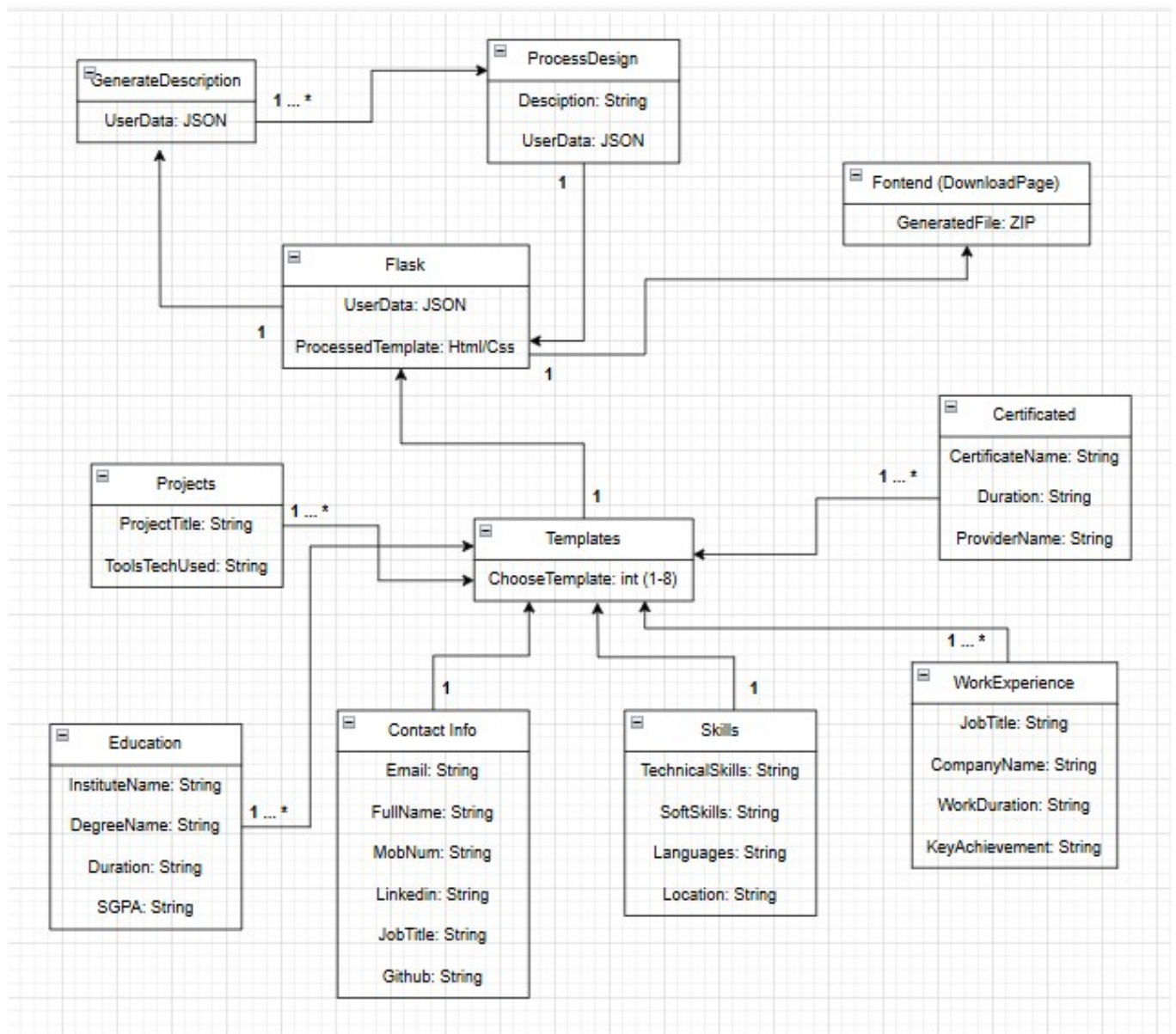
4.7 Data Flow Diagrams (DFD):

A **Data Flow Diagram (DFD)** illustrates how user data moves between the frontend and backend. The frontend captures user details and sends them to the backend, which processes the data, generates HTML, CSS, and PDF files, and compresses them into a ZIP archive. The backend then returns the ZIP file to the frontend, providing a download link for the user.



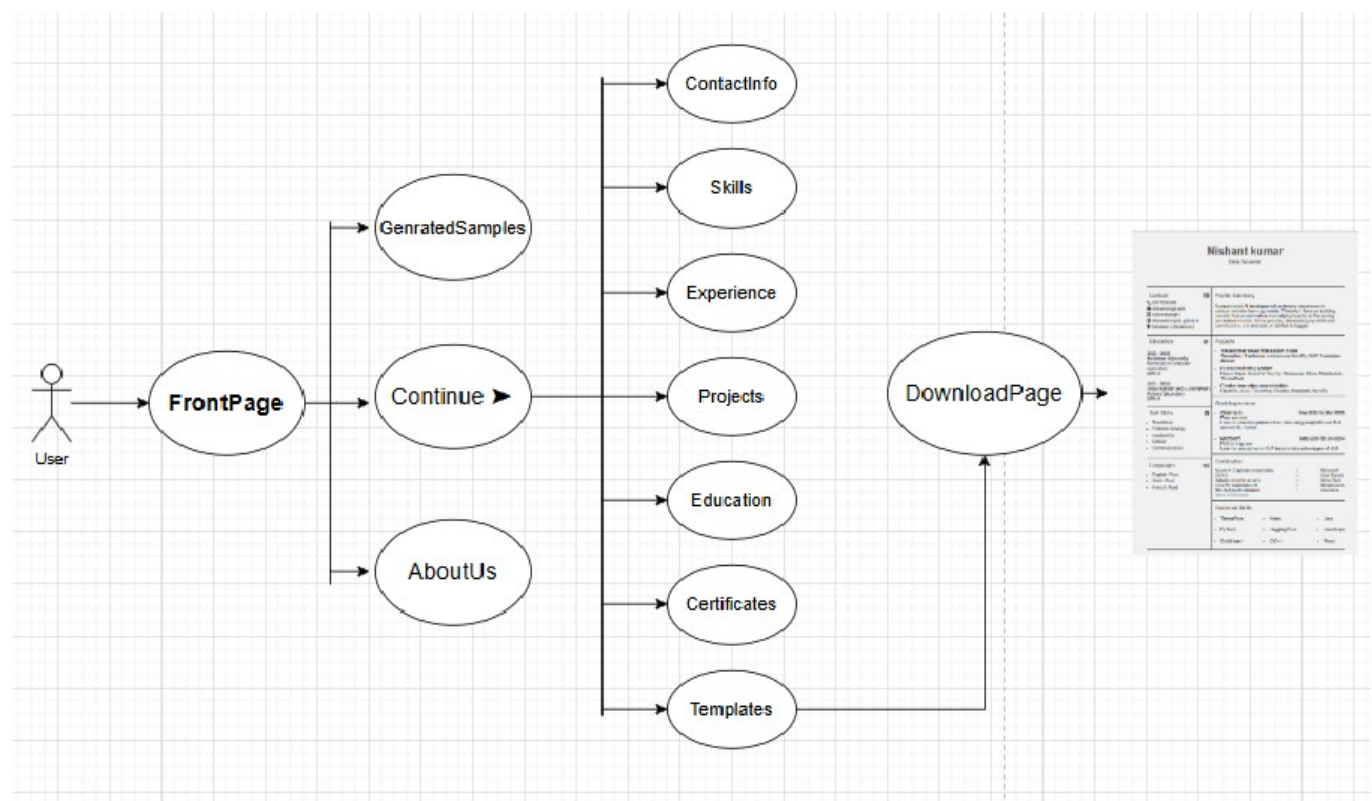
Specific Diagrams

5.1 Class Diagram



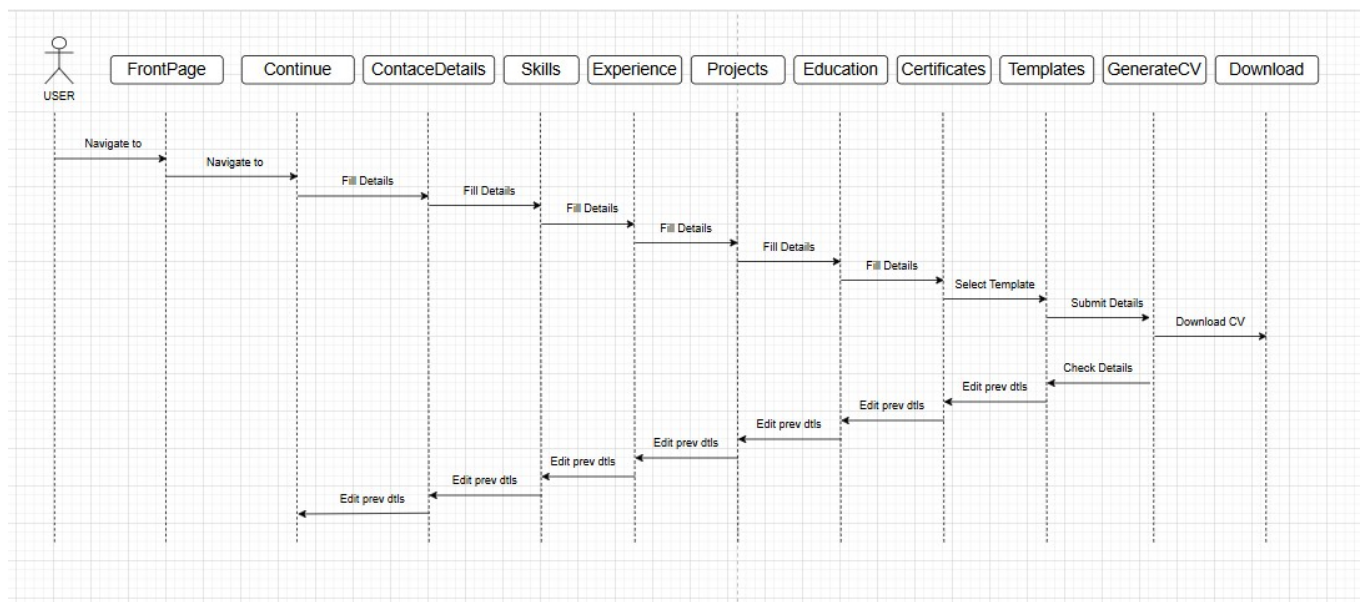
5.2 Use-Case/Activity Diagram:

This use case diagram illustrates the user journey on a resume generation website. It starts with the user accessing the **"FrontPage"** and provides options like **"GeneratedSamples"** and **"AboutUs."** By selecting **"Continue,"** the user progresses through steps to input essential details such as **"ContactInfo,"** **"Skills,"** **"Experience,"** **"Projects,"** **"Education,"** **"Certificates,"** and **"Templates."** Finally, all steps converge at the **"DownloadPage,"** where the user can download their completed resume.



5.4 Sequence Diagrams

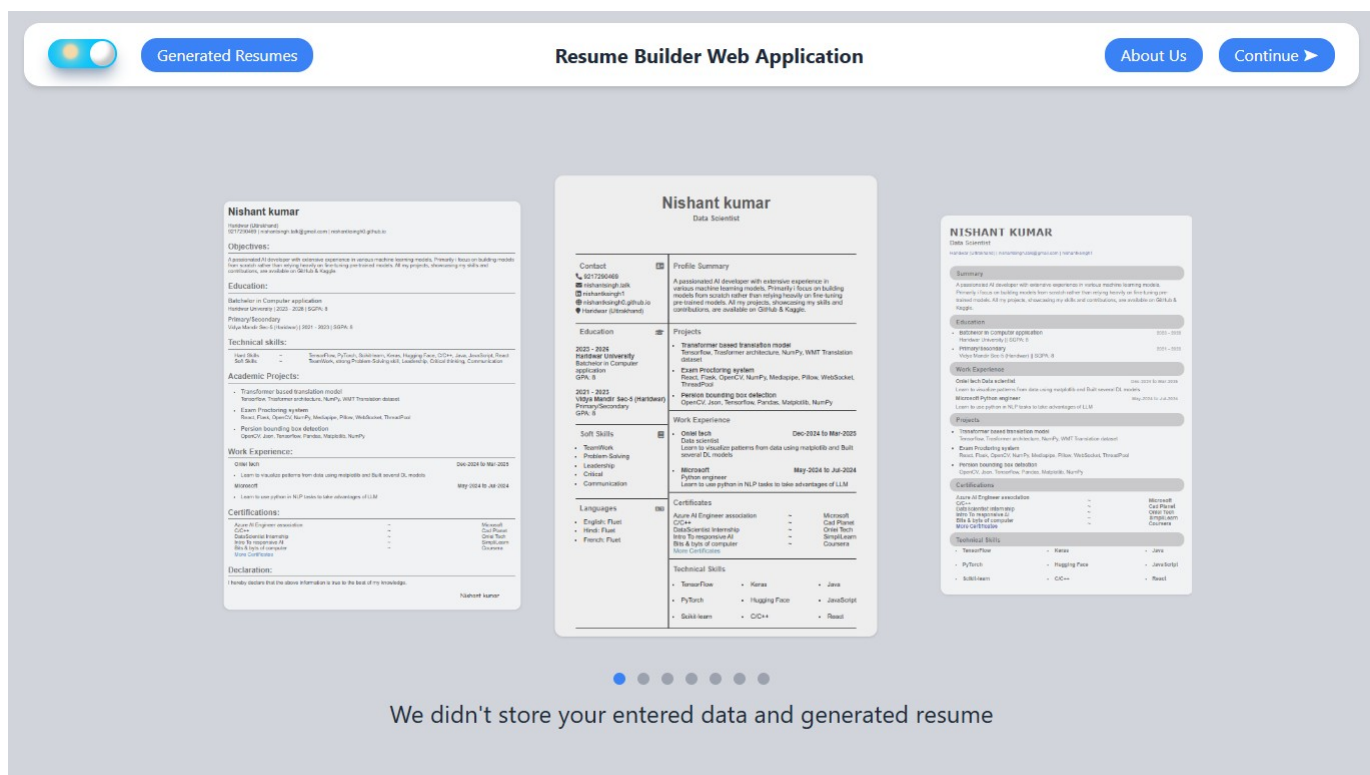
The sequence diagram outlines the step-by-step interaction flow for creating and downloading a CV on an online platform. It begins with the user accessing the **FrontPage** and progresses through key stages like filling out **ContactDetails**, **Skills**, **Experience**, **Projects**, **Education**, and **Certificates**. The user then selects a **Template**, generates the CV, and reviews the details before reaching the final step, **Download**, where the completed CV is obtained. The diagram also includes options for editing details at various stages, ensuring flexibility and accuracy throughout the process. It provides a clear visual representation of the workflow.



Product Features

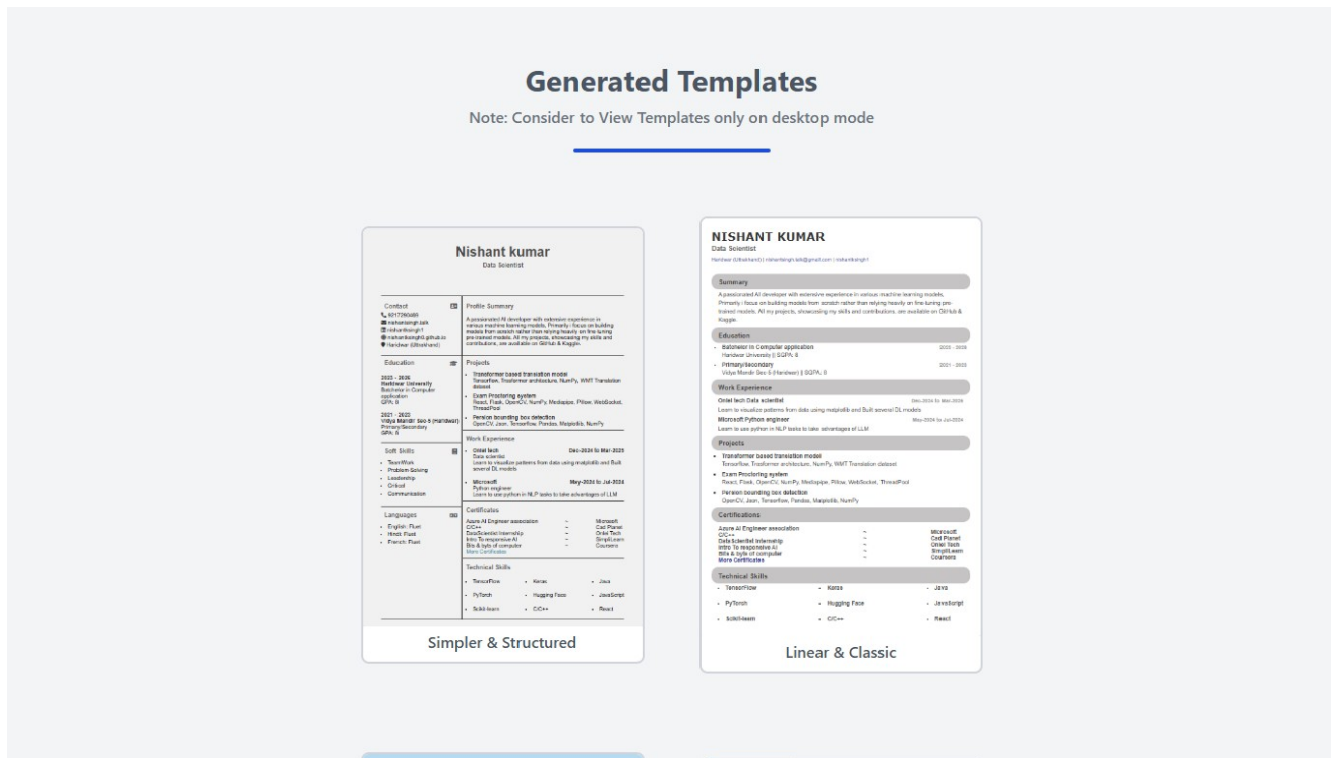
6.1 Screenshots

• Front Page:



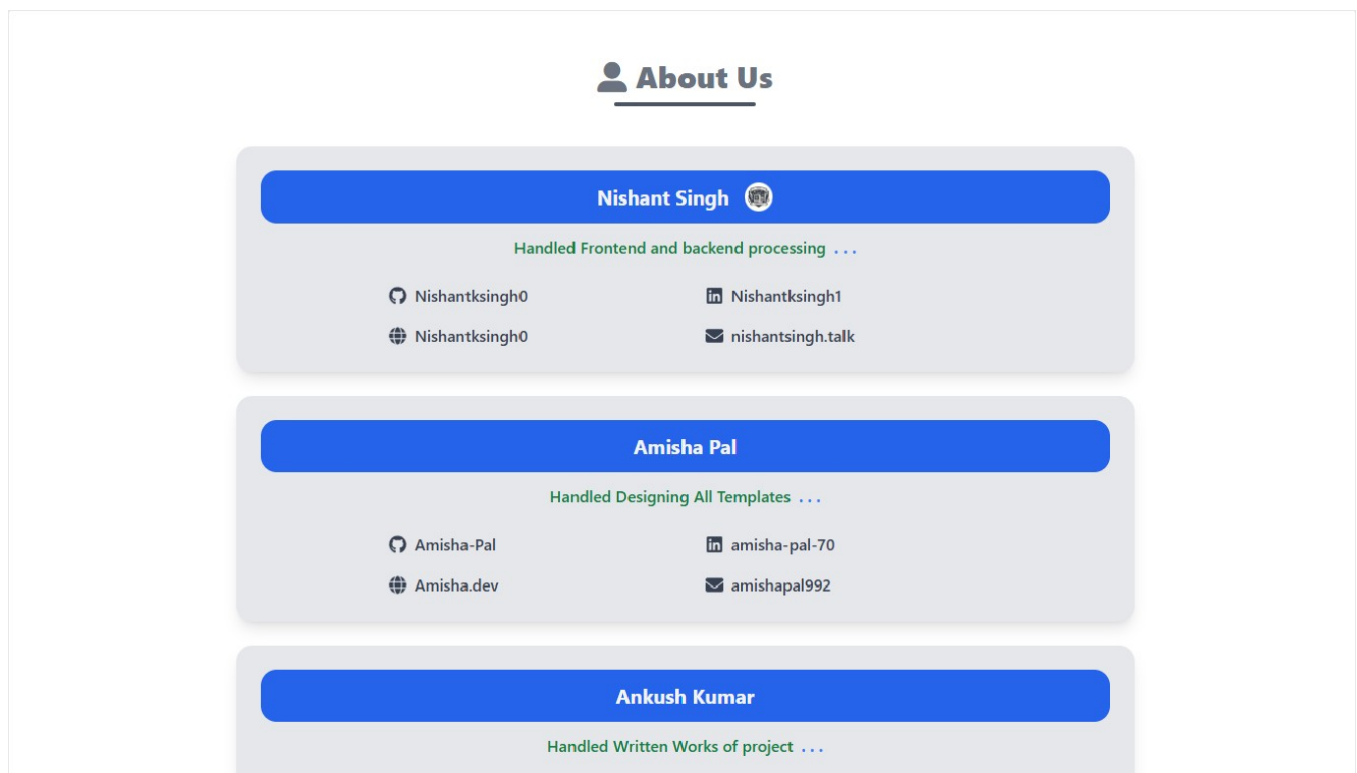
The front page includes a **Dark/Light mode switch**, a **Generated Templates** navigation button, an **About Us** button, and a **Continue** button for users to fill in their details. In the center, there's a **sliding view of generated resume templates**, and at the bottom, a **feature showcasing writing animation** for an interactive experience.

• Generated Templates Page:



The **Generated Templates** page displays images of resume templates created by the website. Below each template, there are two buttons: **"View Template"** to preview it in the browser and **"View HTML/CSS Code"** to access its source code. The page provides an easy way for users to explore and customize different resume designs.

- **About Page:**



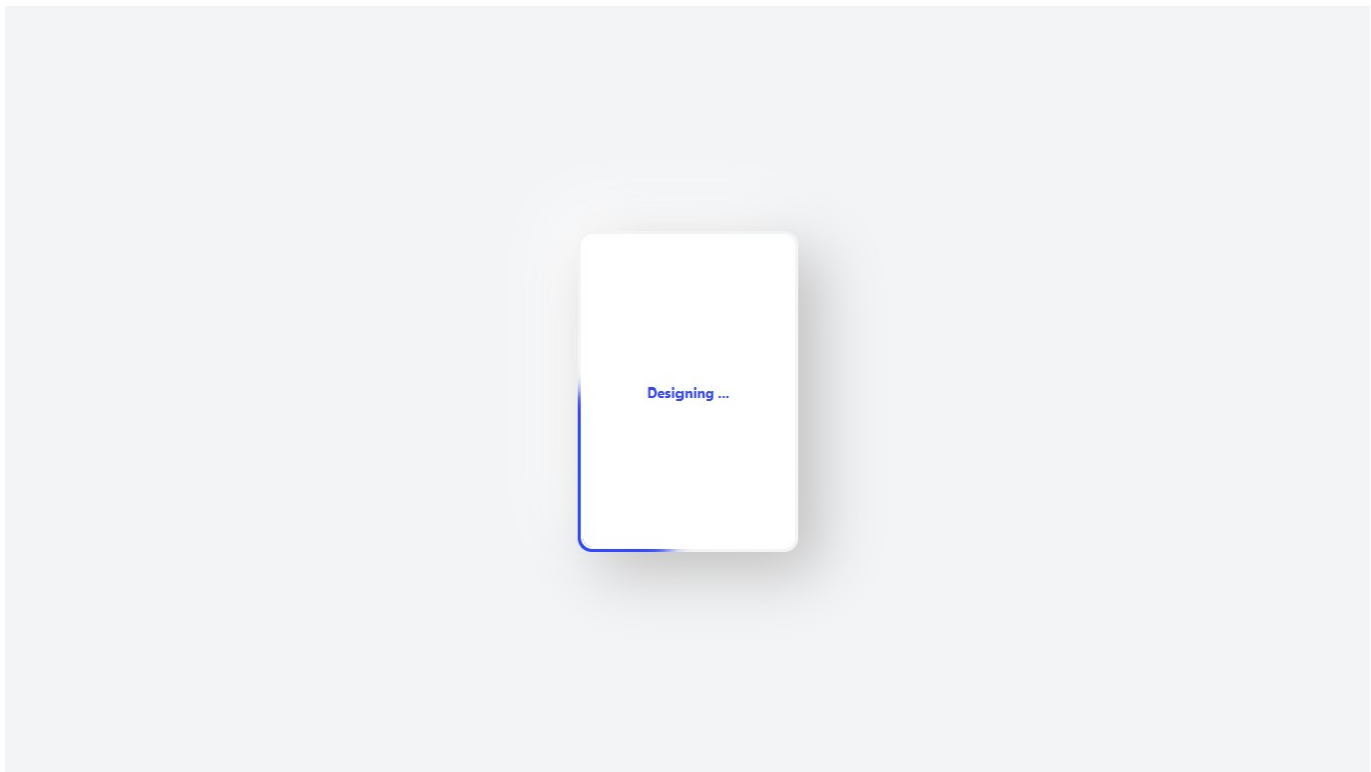
The **About Us** page introduces the team members who contributed to building the Resume Builder Web Application. It includes their **GitHub, LinkedIn, portfolio links, and email addresses**, allowing users to connect with them. This page highlights the team's efforts and expertise in developing the platform.

- **Get Info page:**

The screenshot displays the 'BRAVERS' application interface. On the left, a vertical sidebar contains a list of steps: 1 Contact Info, 2 Skills, 3 Work Experience, 4 Projects, 5 Education, 6 Certificates, and 7 Template. The 'Contact Info' step is highlighted. Below this list is a button labeled 'Example Processing'. The main content area is titled 'Contact Information' and contains several input fields: 'Full Name' (filled with 'xyz'), 'Phone Number' (with a placeholder 'Enter 10-digit phone number'), 'Email Address' (filled with 'xyz231@gmail.com'), 'LinkedIn UserName' (filled with 'xyz231'), 'Portfolio URL / Github UserName' (with a placeholder 'Personal portfolio URL if have else add GitHub UserName'), and 'Job Title' (filled with 'Data Scientist'). A blue 'Next >' button is located at the bottom right of the form.

The **GetInfo** page allows users to fill in their details to generate a resume. It includes sections for **Contact Info, Skills, Education, Work Experience, Projects, Certifications, and Profile Summary**. A clean and structured form ensures easy input, helping users create a professional resume effortlessly.

- **Result page:**



The **Result Page** generates the user's resume based on the provided information. It displays a **preview of the generated CV** with options to **download it as ZIP (PDF or their HTML/CSS code.)** This page ensures a seamless experience for users to obtain their professionally formatted resumes.

CODE (Frontend)

FrontPage:

```
import React, { useContext, useEffect } from "react";

import Typed from "typed.js";

import Examplepages from './Examplepage.jsx'

import { useNavigate } from 'react-router-dom';

import { ThemeContext } from "../ThemeContext.jsx";

import { MdDarkMode } from "react-icons/md";

import Switch from "../Switch.jsx";

const features=[

    "Build your resume easily",

    "No need to worry about what to add what to not",

    "Get high rated, ATS freindly resumes",

    "Different variants of resume templates available",

    "We carefully selected strong resume data inputs from users after a high research.",

    "Download your resume in pdf or web formate",

    "Use AI to create a perfect summary for your resume",

    "Completely free to use",

    "We didn't store your entered data and generated resumes (for user privacy)",

    "Light weight application. just take 1.8MB of RAM and approximately 0 CPU time"

]

const FrontPage = () => { const navigate = useNavigate();

const handleContinue = () => { navigate('/GetInfo'); };

const handleViewTemplates = () => { navigate('/ViewTemplates'); };
```

```

const handleAboutUs = () => { navigate('/AboutUs'); };

const { isDark, setIsDark } = useContext(ThemeContext);

const handleTheme = () => { setIsDark((prev) => !prev); };

useEffect(() => { const typedMobile = new Typed("#mobile-typing-text", { strings: features,
loop: true, typeSpeed: 35, backSpeed: 25, backDelay: 500, cursorChar: " ", });

const typed = new Typed("#desktop-typing-text", {
  strings: features,
  loop: true,
  typeSpeed: 35,
  backSpeed: 25,
  backDelay: 500,
  cursorChar: " "
});

return () => {
  typedMobile.destroy();
  typed.destroy();
};
}, []);return (

  <button
    className="px-4 py-[5px] bg-blue-500 text-white rounded-2xl
hover:bg-blue-700 hover:scale-105"
    onClick={handleViewTemplates}
    title="View Generated resume samples"
  >
    Generated Resumes
  </button>
</div>

  <h1 className="text-xl font-bold text-gray-800 dark:text-white no-
underline">
    <a href="https://github.com/NishantkSingh0/Resume-Builder"
title="Address to GitHub repository of this project" target="_blank">
      <span className="lg:hidden">Resume Builder</span>
      <span className="hidden lg:block">Resume Builder Web
Application</span>
    </a>
  </h1>
  <div className="flex space-x-4">
    <button className="px-4 py-[5px] bg-blue-500 text-white rounded-2xl
hover:bg-blue-700 hover:scale-105" onClick={handleAboutUs}>
      About Us
    </button>

```

```

        <button className="px-4 py-[5px] bg-blue-500 text-white rounded-2xl
hover:bg-blue-700 hover:scale-105" onClick={handleContinue}>
            Continue
        </button>
    </div>
</div>
<div className="hidden md:flex flex-col items-center justify-center
flex-1 mt-3 mb-3">
    <Examplepages />
    <span id="desktop-typing-text" className="hidden md:inline-block text-
xl md:text-2xl text-gray-800 h-6 dark:text-white"></span>
</div>

```

```

{ /* Mobile View */}
<div className="flex md:hidden justify-between items-center w-full h-14
px-6 py-3 bg-white shadow-md mt-6 rounded-3xl dark:bg-slate-800">
    <button
        className="text-2xl mr-8 mt-1"
        title="The Dark/Light mode will be chosen randomly on each refresh,
allowing users to experience both modes. You can also set it to your
preferred mode."
        onClick={handleTheme}>
        <Switch/>
    </button>

```

```

    <div className="md:hidden flex space-x-4">
        { /* <button
            className="px-4 py-[5px] bg-blue-500 text-white rounded-2xl
hover:bg-blue-700 hover:scale-105"
            onClick={handleViewTemplates}
            title="View Generated resume samples"
        >
            Generated
        </button> */}
        <button className=" px-4 py-[3px] bg-blue-500 text-white rounded-
full hover:bg-blue-700" onClick={handleContinue}>
            Continue
        </button>

```

```

    </div>
</div>

```

```

<div className="md:hidden flex justify-center flex-col items-center
flex-grow">
    <h1 className="text-2xl md:hidden sm:text-3xl font-bold mb-3

```

```

dark:text-white">
    Resume Builder
  </h1>
  <p className="text-sm sm:text-lg text-gray-500 font-semibold mb-4">
    {/* Here the mobile span are with desktop id */}
    <span id="mobile-typing-text" className="md:hidden text-xl sm:text-
2xl text-gray-800 h-6 mb-4 dark:text-white"></span>
  </p>
  <a
    onClick={handleViewTemplates}
    className="cursor-pointer text-gray-600 dark:text-gray-300/80 font-
bold"
  >
    Generated Templates
  </a>
</div>

```

```

<div className="md:hidden absolute bottom-4 font-bold text-gray-800
left-1/2 -translate-x-1/2 text-center dark:text-white/80">

```

About Us

```

{/* <p className="text-sm sm:text-lg text-gray-500 font-semibold mb-4">
Here the desktop span are with mobile id
  <span id="mobile-typing-text" className="hidden md:inline-block text-
xl md:text-2xl text-gray-800 h-6 mb-3 dark:text-white"></span>
  </p> */}
</div>

```

```

); });

```

```

export default FrontPage;

```

• Generated Template Page:

```
export default function ViewTemplates() {
  const items = [
    {
      img: "/Resume-Builder/Temp/temp1.png",
      title: "Simpler & Structured",
      codeLink: "https://github.com/NishantkSingh0/Templates/blob/main/T1.html",
      templateLink: "https://nishantksingh0.github.io/Templates/T1",
    },
    {
      img: "/Resume-Builder/Temp/temp2.png",
      title: "Linear & Classic",
      codeLink: "https://github.com/NishantkSingh0/Templates/blob/main/T2.html",
      templateLink: "https://NishantkSingh0.github.io/Templates/T2",
    },
    {
      img: "/Resume-Builder/Temp/temp3.png",
      title: "Colourful & Attractive",
      codeLink: "https://github.com/NishantkSingh0/Templates/blob/main/T3.html",
      templateLink: "https://NishantkSingh0.github.io/Templates/T3",
    },
    {
      img: "/Resume-Builder/Temp/temp4.png",
      title: "Colourful & Highly Designed",
      codeLink: "https://github.com/NishantkSingh0/Templates/blob/main/T4.html",
      templateLink: "https://NishantkSingh0.github.io/Templates/T4",
    },
    {
      img: "/Resume-Builder/Temp/temp5.png",
      title: "Simpler & Linear",
      codeLink: "https://github.com/NishantkSingh0/Templates/blob/main/T5.html",
      templateLink: "https://NishantkSingh0.github.io/Templates/T5",
    },
    // {
    //   img: "/Resume-Builder/Temp/temp6.png",
    //   title: "Designed & Attractive",
    //   codeLink: "#",
    //   templateLink: "#",
    // },
    {
      img: "/Resume-Builder/Temp/temp7.png",
      title: "Highly Simpler & Classic",
      codeLink: "https://github.com/NishantkSingh0/Templates/blob/main/T7.html",
    }
  ]
}
```

```

    templateLink: "https://NishantkSingh0.github.io/Templates/T7",
  },
];

return (
<div className="flex flex-col items-center justify-center min-h-screen bg-gray-100 dark:bg-slate-900 p-4">
  <h3 className="mt-10 mb-2 text-3xl text-gray-600 dark:text-slate-200 font-bold">Generated Templates</h3>
  <h5 className="mb-4 text-sm md:text-base font-semibold text-gray-500">Note: Consider to View Templates only on desktop mode</h5>
  <div className="w-[200px] h-1 bg-blue-700 mb-16 mx-auto mt-1 rounded dark:bg-blue-500"></div>

  { /* Change grid-cols-1 to grid-cols-2 for mobile */ }
  <div className="grid grid-cols-2 gap-14 sm:grid-cols-2 md:grid-cols-2 max-w-5xl mx-auto place-items-center">
    {items.map((item, index) => (
      <div key={index} className="group relative mb-6 bg-white dark:bg-slate-700 hover:shadow-2xl hover:scale-105 transition-transform border-2 dark:shadow-[0_-4px_10px_rgba(0,0,0,0.1)] border-gray-300 dark:border-gray-700 dark:shadow-gray-800 dark:hover:shadow-gray-700 rounded-lg overflow-hidden w-40 sm:w-44 md:w-48 lg:w-64 xl:w-72 flex flex-col items-center">
        { /* Adjust image size */ }
        <img src={item.img} alt={item.title} className="w-full h-auto object-cover dark:opacity-80 dark:brightness-80 dark:contrast-90" />
        <div className="absolute bottom-0 left-0 right-0 bg-white bg-opacity-90 dark:bg-slate-700 p-4 rounded-md opacity-0 group-hover:opacity-100 transition-transform flex flex-col items-center">
          <div className="flex space-x-4">
            <div className="flex space-x-2">
              <a href={item.codeLink} className="text-gray-50 text-xs bg-green-600 hover:bg-green-700 px-2 py-1 rounded-md md:hidden" target="_blank">
                Code
              </a>
              <a href={item.templateLink} className="text-gray-50 text-xs bg-green-600 hover:bg-green-700 px-2 py-1 rounded-md md:hidden" target="_blank">
                Resume
              </a>
              <a href={item.codeLink} className="text-gray-50 text-xs bg-green-600 hover:bg-green-700 px-2 py-1 rounded-md hidden md:block" target="_blank">
                View Code
              </a>
              <a href={item.templateLink} className="text-gray-50 text-xs bg-green-600 hover:bg-green-700 px-2 py-1 rounded-md hidden md:block" target="_blank">
                View Resume
              </a>
            </div>
          </div>
        </div>
      </div>
    ))}
  </div>

```

</div>

</div>

</div>

<div className="font-semibold text-gray-600 dark:text-gray-200 text-xs pb-2 pt-1 md:text-base"> {item.title} </div>

</div>

))}

</div>

</div>

);

}

• About Page:

```
import { FaGithub, FaLinkedin, FaEnvelope, FaGlobe } from "react-icons/fa";
import ExpandButton from './ExpandButton.jsx'

const developers = [
  {
    name: "Nishant Singh",
    // "Developed and integrated frontend and backend systems, Proposed innovative project
    ideas with structured workflows, Researched high-rated resumes for optimization, Analyzed
    ATS algorithms for effective resume selection."
    role: "Handled Frontend and backend processing",
    github: "Nishantksingh0",
    gitLink: "https://github.com/NishantkSingh0",
    linkedin: "Nishantksingh1",
    linLink: "https://www.linkedin.com/in/nishantksingh1",
    portfolio: "Nishantksingh0",
    prtLink: "https://nishantksingh0.github.io",
    email: "nishantsingh.talk",
    emailLink: "mailto:nishantsingh.talk@gmail.com"
  },
  {
    name: "Amisha Pal",
    role: "Handled Designing All Templates",
    github: "Amisha-Pal",
    gitLink: "https://github.com/Amisha-Pal",
    linkedin: "amisha-pal-70",
    linLink: "https://www.linkedin.com/in/amisha-pal-703493328/",
    portfolio: "Amisha.dev",
    prtLink: "",
    email: "amishapal992",
    emailLink: "mailto:amishapal9927@gmail.com"
  },
  {
    name: "Ankush Kumar",
    role: "Handled Written Works of project",
    github: "Ankush2011",
    gitLink: "https://github.com/Ankush201109",
    linkedin: "Ankush-kumar-5",
    linLink: "https://www.linkedin.com/in/ankush-kumar-505318231",
    portfolio: "Ankush.dev",
    prtLink: "",
    email: "akbahot2004",
    emailLink: "mailto:akbahot2004@gmail.com"
  }
]
```



```

    },
    // {
    //   name: "Nisha Mandal",
    //   role: "Enhanced UI by adding Nightmode's and Other essential components",
    //   github: "NishaKumari92",
    //   gitLink: "https://github.com/NishaKumari9279",
    //   linkedin: "nisha-kumari-59",
    //   linLink: "https://www.linkedin.com/in/nisha-kumari-598611344?utm_source=share&utm_campaign=share_via&utm_content=profile&utm_medium=android_app",
    //   portfolio: "Nisha.dev",
    //   prtLink: "",
    //   email: "nishakumari92",
    //   emailLink: "mailto:nishakumari927979@gmail.com"
    // },
  ];

```

```

function AboutUs() {
  return (
    <div className="min-h-screen text-gray-100 py-12 px-6 flex flex-col items-center dark:bg-slate-900">
      <div>
        <h1 className="text-2xl font-extrabold text-center mb-2 text-gray-500 md:text-3xl dark:text-amber-300"><i className="fas fa-user-alt"/> About Us</h1>
        <div className="w-[80%] h-1 bg-gray-600 mx-auto mb-10 mt-1 rounded dark:bg-amber-400"></div>
      </div>
      <div className="w-full max-w-4xl flex flex-col space-y-6">
        {developers.map((dev, index) => (
          <div
            key={index}
            className="w-full bg-gray-200 rounded-2xl p-6 text-center shadow-lg hover:shadow-2xl transition-shadow duration-300 dark:bg-slate-800"
            style={{ minWidth: "33%" }}
          >
            <div className="flex items-center bg-blue-600 text-xl font-bold p-3 rounded-2xl mb-3">
              <div className="flex items-center justify-center flex-1">
                {dev.name}
                {dev.name === "Nishant Singh" && <ExpandButton
                  content="/Resume-Builder/BinaryBunch.jpeg"
                  message="Have secured 3rd rank in University 24hr Hackathon (TechWizard) on"
                  linkText="Proctoring System"
                  linkUrl="https://github.com/NishantkSingh0/Proctoring-System"
                />}
              </div>
            </div>
          </div>
        ))}
      </div>
    </div>
  );
}

```

```

<p className="text-green-700 mb-6 font-semibold dark:text-blue-300">{dev.role}
{dev.name === "Nishant Singh" && <ExpandButton
  content=" . . . "
  message=[
    "Developed and integrated frontend and backend systems.",
    "Proposed innovative project ideas with structured workflows.",
    "Researched high-rated resumes for optimization.",
    "Analyzed ATS algorithms for effective resume selection."
  ]
}/>
{dev.name === "Amisha Pal" && <ExpandButton
  content=" . . . "
  message=[
    "Designed all templates in HTML/CSS",
    "Researched designs and innovation ideas",
    "Gived optimization idea for various modules",
    "Researched other Resume builder websites and suggested improvemnets"
  ]
}/>
{dev.name === "Ankush Kumar" && <ExpandButton
  content=" . . . "
  message=[
    "Managed All written works (Synopsis, Project file)",
    "Designed PPT for presentation",
    "Researched other Resume builder websites and suggested improvemnets"
  ]
}/>
</p>

```

```

<div className="flex flex-col items-center space-y-4 px-4 md:px-32">
  <div className="flex flex-col md:flex-row justify-between w-full max-w-2xl space-y-4 md:space-y-0 md:space-x-10">
    <div className="flex items-center space-x-2 text-gray-700 font-semibold w-full md:w-1/2 justify-center md:justify-start dark:text-slate-300">
      <FaGithub className="text-lg" />
      <a href={dev.gitLink} onClick={(e) => { if (!dev.gitLink) e.preventDefault(); }}
        title={dev.gitLink} target="_blank" rel="noopener noreferrer"
        className="hover:underline">{dev.github}</a>
    </div>
    <div className="flex items-center space-x-2 text-gray-700 font-semibold w-full md:w-1/2 justify-center md:justify-start dark:text-slate-300">
      <FaLinkedin className="text-lg" />
      <a href={dev.linLink} onClick={(e) => { if (!dev.linLink) e.preventDefault(); }}
        title={dev.linLink} target="_blank" rel="noopener noreferrer"
        className="hover:underline">{dev.linkedin}</a>
    </div>
  </div>
</div>

```

<div className="flex flex-col md:flex-row justify-between w-full max-w-2xl space-y-4 md:space-y-0 md:space-x-10">

<div className="flex items-center space-x-2 text-gray-700 font-semibold w-full md:w-1/2 justify-center md:justify-start dark:text-slate-300">

<FaGlobe className="text-lg" />

```
<a href={dev.prtLink} onClick={(e) => { if (!dev.prtLink) e.preventDefault(); }}
title={dev.prtLink} target="_blank" rel="noopener noreferrer"
className="hover:underline">{dev.portfolio}</a>
```

```
<div className="flex items-center space-x-2 text-gray-700 font-semibold w-full md:w-1/2 justify-center md:justify-start dark:text-slate-300">
```

<FaEnvelope className="text-lg" />

```
<a href={dev.emailLink} onClick={ (e) => { if (!dev.emailLink)
e.preventDefault(); } } title={dev.emailLink} target="_blank" rel="noopener noreferrer"
className="hover:underline">{dev.email}</a>
```

$$))\}$$

);

$$\}$$

```
export default AboutUs;
```

CODE (Backend)

- **Template Processing (T1 in GenerateTemplates.py)**

```
def T1(jsonData,desc):
    Education="\n".join((f"""
        <div class="SubSec-title
        TextLight"><b>{edu['graduationYear']}<br>{edu['institutionName']}</b></div>
        {edu['degreeName']} <br>
        GPA: {edu['currentSGPA']} """
    for edu in jsonData['education']))+'<br><br>'

    skills=jsonData['skills']['hardSkills'].split(',')
    eachColumn=len(skills)//3
    column1=eachColumn
    column2=eachColumn
    column3=eachColumn
    if eachColumn%3==1:
        column1+=1
    elif eachColumn%3==2:
        column1+=1
        column2+=1
    HardSkills=f"""<ul>
        {' <br> '.join(("<li>" +skills[skl]+"</li>" for skl in range(column1))))}
    </ul>
    <ul>
        {' <br> '.join(("<li>" +skills[skl]+"</li>" for skl in
range(column1,column2+column1))))}
    </ul>
    <ul>
        {' <br> '.join(("<li>" +skills[skl]+"</li>" for skl in
range(column2+column1,column3+column2+column1))))}
    </ul>"""

    softSkills="\n".join((f"<li>{removespace(skill)}</li>" for skill in
jsonData['skills']['softSkills'].split(',')))

    languages="\n".join((f"<li>{removespace(skill)}: Fluet</li>" for skill in
jsonData['contactInfo']['Languages'].split(',')))

    Projects="\n".join((
f"""
<div class="Ritem">
```

```

    <li> <div class="item-title TextLight">{proj['projectTitle']} </div>
    <div> {proj['toolsTechUsed']} </li>
</div>\n"" for proj in jsonData['projects'])

```

```

WorkExp='\n <br>'.join((f""
<li> <div class="item-title TextLight">{we['companyName']}<div>{we['WorkDuration']}</div>
</div>
{we['jobTitle']} <br>
{we['keyAchievements']}</li>"" for we in jsonData['workExperience']))

```

```

css=""body {
    font-family: Arial, sans-serif;
    margin: 0;
    padding: 0;
    background-color: #d6cece;
    display: flex;
    justify-content: center;
    align-items: center;
    height: 1120px;
}

```

```

.resume {
    width: 800px;
    background: #f1f1f1;
    border-radius: 15px;
    border: 1px solid #ddd;
    padding: 20px;
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
}

```

```

.header {
    text-align: center;
    border-top: 2px solid #353333;
    padding: 80px 0px 80px 0px;
    margin: 0px 40px 0px 40px;
    background-image: url('/Img/Sig.png');
    background-repeat: no-repeat;
    background-size: 250px;
    background-position: center;
    margin-top: 10px;
}

```

```

.header h1 {
    margin: 0;
    font-size: 40px;
    color: #4e4c4c;
}

```

```
.header h2 {  
    margin: 5px 0 0;  
    font-size: 18px;  
    color: #555;  
}
```

```
.SUsection {  
    padding-top: 10px;  
    padding-left: 8px;  
}
```

```
.section,.SUsection {  
    margin-bottom: 15px;  
    border-bottom: 2px solid #000000;  
}
```

```
.content .left .SubSec-title {  
    margin-top: 20px;  
}
```

```
.rotate-90 {  
    transform: rotate(90deg);  
}
```

```
.Contact div {  
    margin-bottom: 4px;  
    margin-left: 8px;  
}
```

```
.Contact {  
    width: 33%;  
}
```

```
.Contact,.Usection {  
    padding-top: 20px;  
}
```

```
.NoneDecorationBlack a {  
    text-decoration: none;  
    color: #000000;  
}
```

```
.colorBlue{  
    color: #277ca3;  
}
```

```

.TextLight{ /* Used for subheadings of the content & contact fasfonts to make it more classic */
  color: #333333;
}

.NoneDecoration a { /* Used in More Certificates to remove text decoration but add blue
color to make it Easy to understand */
  text-decoration: none;
  color: #277ca3;
}

.subcont,.SkillSubCon {
  display: flex;
  justify-content: space-between;
}

.section-title {
  font-size: 18px;
  color: #5e6163;
  padding: 0 10px 5px 10px;
  margin-bottom: 10px;
}

.section{
  padding-left: 8px;
}

.LDsection{
  padding-left: 8px;
}

.content {
  display: flex;
  justify-content: center;
  border-bottom: 2px solid #353333;
  margin: 0 20px 0 20px;
}

.upperContent {
  display: flex;
  justify-content: center;
  margin: 0 20px 0 20px;
  border-top: 2px solid #353333;
  border-bottom: 2px solid #353333;
}

.upperContent .Usection {
  border-left: 4px solid #5e6163;
}

```

```

        width: 67%;
    }

    .content .left {
        width: 33%;
        height: 100%;
        border-radius: 2px;
    }

    .section-title,.content .right .item-title {
        display: flex;
        justify-content: space-between;
        align-items: center;
    }

    .content .right {
        width: 67%;
        height: 100%;
        border-left: 4px solid rgb(109, 106, 106);
    }

    .Litem {
        margin-bottom: 10px;
        padding-right: 20px;
    }

    .Ritem {
        margin-bottom: 10px;
        padding-left: 10px;
    }

    .item-title {
        font-weight: bold;
    }

    ul {
        padding-left: 20px;
        margin: 5px 0;
    }

    ul li {
        margin-bottom: 5px;
    }
}""

updatedTemplate=f""
<!DOCTYPE html>
<html lang="en">

```



```

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Designed by BRAVERS</title>
  <link rel="icon" type="image/png" href="bravers.png" />      <!-- Address you fevicon here -->
  <style>
    {css}
  </style>
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/5.15.3/css/all.min.css">

</head>
<body>
  <div class="resume">
    <div class="header">
      <h1>{jsonData['contactInfo']['fullName']}</h1>
      <h2>{jsonData['contactInfo']['jobTitle']}</h2>
    </div>

    <div class="upperContent">
      <div class="Contact">
        <div class="section-title"><b>Contact</b> <i class="fas fa-address-card"></i></div><br>
        <div class="Litem"><i class="fa fa-phone rotate-90 TextLight"></i>
{jsonData['contactInfo']['phoneNumber']} </div>
        <div class="Litem NoneDecorationBlack"><i class="fas fa-envelope TextLight"></i><a
href="mailto:"+{jsonData['contactInfo']['emailAddress']}>
{jsonData['contactInfo']['emailAddress'].split('@')[0]} </a></div>
        <div class="Litem NoneDecorationBlack"><i class="fab fa-linkedin TextLight"></i><a
href="{https://www.linkedin.com/in/"+jsonData['contactInfo']['linkedin']} target="_blank">
{jsonData['contactInfo']['linkedin']} </a></div>
        <div class="Litem NoneDecorationBlack"><i class="fas fa-globe TextLight"></i><a
href="{jsonData['contactInfo']['portfolio'] if validators.url(jsonData['contactInfo']['portfolio']) else
'https://github.com/'+jsonData['contactInfo']['portfolio']}" target="_blank">
{jsonData['contactInfo']['portfolio']} </a></div>
        <div class="Litem "><i class="fa fa-map-marker TextLight"></i>
{jsonData['contactInfo']['Location']}</div> <br>
      </div>

      <div class="Usection">
        <div class="section-title"><b>Profile Summary</b></div>
        <p class="Ritem">
          <div class="Litem">
            <p class="Ritem">
              {desc}
            </p>
          </div>
        </p>
      </div>
    </div>
  </div>

```

</div>

<div class="content">

<div class="left">

<div class="SUsection">

<div class="section-title">Education <i class="fas fa-graduation-cap"></i></div>
{Education}

</div>

<div class="section">

<div class="section-title">Soft Skills <i class="fa fa-book"></i></div>

{softSkills}

</div>

<div class="LDsection">

<div class="section-title">Languages <i class="fa fa-language"></i></div>

{languages}

</div>

</div>

<div class="right">

<div class="SUsection">

<div class="section-title">Projects</div>

{Projects}

</div>

<div class="section">

<div class="section-title">Work Experience</div>

<div class="Ritem">

{WorkExp}

</div>

</div>

<div class="section">

<div class="section-title">Certificates</div>

<div class="Ritem subcont NoneDecoration">

<div>{'
 '.join((cer['certificateName'] for cer in jsonData['certificates']))}
More Certificates</div>

```

        <div>{' <br> '.join(('~' for _ in range(len(jsonData['certificates']))))}<br></div>
        <div>{' <br> '.join((cer['providerName'] for cer in jsonData['certificates']))}<br></div>
    </div>
</div>

<div class="skills">
    <div class="section-title"><b>Technical Skills</b></div>
    <div class="Ritem subcont">
        {HardSkills}
    </div>
</div>
</div>
</div>
</div>
</body>
</html>""""
    return updatedTemplate

```

- **Flask (Main.py):**

```
from flask import Flask, request, jsonify, send_file
from flask_cors import CORS
from ProcessTemplates import GenerateResume
import zipfile
import os

app = Flask(__name__)
CORS(app, resources={r"/upload": {"origins": "http://localhost:5173"}})

@app.route('/upload', methods=['POST'])
def receive_json():
    try:
        data = request.get_json()
        if not data:
            return jsonify({"error": "No JSON received"}), 400

        print("Data Received")

        Address = GenerateResume(data)

        zip_filename = Address + '.zip'

        # Create a ZIP file
        with zipfile.ZipFile(zip_filename, 'w') as zipf:
            zipf.write(Address + ".txt")
            zipf.write(Address + ".pdf")

        print("sent successfully")
        return send_file(zip_filename, as_attachment=True)

    except Exception as e:
        return jsonify({"error": str(e)}), 500

if __name__ == '__main__':
    app.run(debug=True)
```

- **Process Template:**

```
import GenerateTemplates as gtps
import GenerateText as gtxt
import os
import weasyprint
from weasyprint import HTML

def GenerateResume(JsonData):
    ExampleDesc="A passionate AI developer with extensive experience in various
    machine learning models, Primarily i focus on building models from scratch rather than
    relying heavily on fine-tuning pre-trained models. All my projects, showcasing my skills
    and contributions, are available on GitHub & Kaggle."
    GeneratedText=""
    Address=f"Backend/OnWebCall/T {JsonData['selectedTemplateEx']}.html"
    match JsonData['selectedTemplateEx']:
        case "1":
            GeneratedText=gtps.T1(JsonData)
        case "2":
            GeneratedText=gtps.T2(JsonData)
        case "3":
            GeneratedText=gtps.T3(JsonData)
        case "4":
            GeneratedText=gtps.T4(JsonData)
        case "5":
            GeneratedText=gtps.T5(JsonData)
        case "6":
            GeneratedText=gtps.T1(JsonData)
        case "7":
            GeneratedText=gtps.T7(JsonData)
        case _:
            print("Incorrect template number provided!")

    with open(Address+'.html','w') as f:
        f.write(GeneratedText)

    HTML(Address+".html").write_pdf(Address+".pdf")

    return Address
```

• Example Generated Resume Template:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <title>Designed by BRAVERS</title>
  <style>
    * {
      box-sizing: border-box;
      margin: 0;
      padding: 0;
    }

    .Name{
      font-family: Verdana;
    }

    body {
      font-family: Arial, sans-serif;
      background: #e4e4e4;
      color: #333;
      padding: 20px;
    }

    .resume-container {
      max-width: 850px;
      height: 1150px;
      margin: 0 auto;
      background: #fff;
      border-radius: 8px;
      box-shadow: 0 0 10px rgba(0, 0, 0, 0.05);
      padding: 20px;
    }

    .header {
      margin-bottom: 2rem;
    }

    .header h1 {
      font-size: 2rem;
```

```
letter-spacing: 1px;  
margin-bottom: 0.25rem;  
text-transform: uppercase;  
}
```

```
.header h2 {  
  font-size: 1.2rem;  
  font-weight: normal;  
  color: #666;  
  margin-bottom: 0.75rem;  
}
```

```
.header p {  
  font-size: 0.9rem;  
  color: #666;  
}
```

```
.summary p {  
  line-height: 1.5;  
  margin: 0 20px 0 20px;  
  margin-bottom: 0.5rem;  
}
```

```
.skills ul {  
  margin-left: 10px;  
  padding-left: 0;  
}
```

```
.skills li {  
  margin: 0 20px 0 30px;  
  margin-bottom: 0.5rem;  
}
```

```
.experience-item {  
  margin: 0 20px 10px 0;  
}
```

```
.experience-item h4 {  
  font-size: 1rem;  
  font-weight: bold;  
  margin-bottom: 0.25rem;  
}
```

```
.experience-item span {  
  font-size: 0.9rem;  
  color: #999;  
}
```

```
.experience-item ul {  
  list-style: disc;  
  margin-left: 1.2rem;  
  margin-top: 0.5rem;  
}
```

```
.experience-item li {  
  margin-bottom: 0.5rem;  
}
```

```
.education {  
  margin-left: 10px;  
}
```

```
.education-item {  
  margin: 0 10px 10px 20px;  
}
```

```
.education-item h4 {  
  font-size: 1rem;  
  font-weight: bold;  
  margin-bottom: 0.25rem;  
}
```

```
.education-item span {  
  font-size: 0.9rem;  
  color: #999;  
}
```

```
.Projects-items {  
  margin-bottom: 10px;  
  margin-left: 20px;  
}
```

```
.Projects-items h4 {  
  font-size: 1rem;  
  font-weight: bold;  
  margin-bottom: 0.25rem;
```



```

}

.Projects-items p {
  font-size: medium;
  font-weight: lighter;
}

.Certificats {
  margin: 10px 20px;
}

.SpaceBetween {
  display: flex;
  justify-content: space-between;
  align-items: center;
  font-weight: bolder;
  color: #525151;
}

.fontlight{
  color: #494848;
}

.fontBold {
  font-weight: bolder;
  color: #333;
}

.NoneDecoration {
  text-decoration: none;
  color: #2d3499;
}

.Heading {
  padding: 13px 20px 10px 20px;
  color: #424141;
  border-radius: 20px;
  margin-bottom: 5px;
  background-color: #c5c3c3;
}
</style>
</head>
<body>

```

```
<div class="resume-container">
  <div class="header">
    <h1 class="Name">Nishant kumar</h1>
    <h2 class="fontBold" style="font-weight: 700;">Data Scientist</h2>
    <p style="color: #333;"> <a href="#" class="NoneDecoration">Haridwar
(Uttarakhand)</a> | <a class="NoneDecoration" href="mailto:nishantsingh.talk@gmail.com"
target="_blank">nishantsingh.talk@gmail.com</a> | <a class="NoneDecoration"
href="https://www.linkedin.com/in/nishantksingh1" target="_blank">nishantksingh1</a></p>
  </div>
```

```
  <h3 class="Heading">Summary</h3>
  <div class="summary">
    <p>A passionated AI developer with extensive experience in various machine learning
models, Primarily i focus on building models from scratch rather than relying heavily on fine-
tuning pre-trained models. All my projects, showcasing my skills and contributions, are
available on GitHub & Kaggle.</p>
  </div>
```

```
  <h3 class="Heading">Education</h3>
  <div class="education">
    <div class="education-item">
      <ul>
        <li><h4 class="SpaceBetween">Batchelor in Computer application <span>2023 -
2026</span></h4></li>
        <p>Haridwar University || SGPA: 8</p>
      </ul>
    </div>
    <div class="education-item">
      <ul>
        <li><h4 class="SpaceBetween">Primary/Secondary <span>2021 -
2023</span></h4></li>
        <p>Vidya Mandir Sec-5 (Haridwar) || SGPA: 8</p>
      </ul>
    </div>
  </div>
```

```
  <h3 class="Heading">Work Experience</h3>
  <div class="experience">
    <div class="experience-item">
      <ul>
        <li class="SpaceBetween">Onlei tech Data scientist <span>Dec-2024 to Mar-
2025</span></li>
      </ul>
    </div>
  </div>
```

- Learn to visualize patterns from data using matplotlib and Built several DL models

- Microsoft Python engineer May-2024 to Jul-2024
- Learn to use python in NLP tasks to take advantages of LLM

Projects

- Transformer based translation model
- Tensorflow, Trasformer architecture, NumPy, WMT Translation dataset

- Exam Proctoring system
- React, Flask, OpenCV, NumPy, Mediapipe, Pillow, WebSocket, ThreadPool

- Persion bounding box detection
- OpenCV, Json, Tensorflow, Pandas, Matplotlib, NumPy

Certifications

Azure AI Engineer association

C/C++

DataScientist Internship

Intro To responsive AI

Bits & byts of computer

[More Certificates](#)

Microsoft

Cad Planet

Onlei Tech

SimpliLearn

Coursera

```
<h3 class="Heading">Technical Skills</h3>
<div class="skills SpaceBetween">
  <ul>
    <li>TensorFlow</li> <br> <li> PyTorch</li> <br> <li> Scikit-learn</li>
  </ul>
  <ul>
    <li> Keras</li> <br> <li> Hugging Face</li> <br> <li> C/C++</li>
  </ul>
  <ul>
    <li> Java</li> <br> <li> JavaScript</li> <br> <li> React</li>
  </ul>
</div>
</div>
</body>
</html>
```

CONCLUSION

The "Online Resume Builder Web Application" effectively addresses the challenges associated with traditional resume-building methods. By integrating AI-based tools like GPT-Neo, NLP libraries, and professional PDF templates, the platform simplifies the resume creation process for users. The system ensures secure, efficient, and user-friendly operations, making it a valuable resource for job seekers across various industries.

This project not only showcases the seamless integration of front-end and back-end technologies but also highlights the potential of AI to enhance productivity in practical applications. With positive feedback from initial users, the platform stands as a promising tool for the future.

FUTURE ENHANCEMENTS

The system is designed with scalability in mind, allowing for continuous development and improvement. Possible future enhancements include:

1. Multilingual Support

To cater to a global audience, the system will incorporate translation features, enabling users to create resumes in multiple languages.

2. Expanded Template Library

Introduce industry-specific templates for fields like healthcare, IT, education, and creative industries.

3. LinkedIn Integration

Allow users to import their LinkedIn profile data directly into the system for faster and easier resume creation.

4. Mobile Application Development

Develop a mobile app for enhanced accessibility, allowing users to build resumes on the go.

5. Advanced AI Features

Implement a recommendation system that suggests keywords and skills based on job descriptions provided by users.

6. Real-time Collaboration

Enable users to share their resumes with career counselors or peers for feedback and real-time editing.

7. Cloud Storage Integration

Offer options to save resumes to popular cloud platforms like Google Drive and Dropbox for easier storage and access.

BIBLIOGRAPHY

References to frameworks, libraries, and resources used in the project development:

- 1. Flask Framework Documentation:** <https://flask.palletsprojects.com>
- 2. GPT-Neo GitHub Repository:** <https://github.com/EleutherAI/gpt-neo>
- 3. React.js Official Documentation:** <https://reactjs.org>
- 4. WeasyPrint Library Documentation:** <https://weasyprint.org>
- 5. ReportLab Library Documentation:** <https://www.reportlab.com>
- 6. PostgreSQL Official Documentation:** <https://www.postgresql.org>
- 7. HTTPS and Secure Communication:**
<https://developer.mozilla.org/en-US/docs/Web/HTTP/Overview>
- 8. spaCy NLP Documentation:** <https://spacy.i>