# Resume Builder

A PROJECT REPORT
For

Mini Project(KCA353)
Session(2023-24)

Submitted By

Himanshee Singh

2200290140070

Submitted in partial fulfilment of the
Requirements for the Degree of

## MASTER OF COMPUTER APPLICATIONS

Under the Supervision of

Dr. Vipin Kumar
Associate Professor

Submitted to

DEPARTMENT OF COMPUTER APPLICATIONS
KIET Group of Institutions, Ghaziabad
Uttar Pradesh-201206
(Feb 2024)

# CERTIFICATE

Certified that **Himanshee Singh 2200290140070** have carried out the project work having “**Resume Builder**” for Master of Computer Applications from Dr. A.P.J. Abdul Kalam Technical University (AKTU) (formerly UPTU), Technical University, Lucknow under my supervision. The project report embodies original work, and studies are carried out by the student himself / herself and the contents of the project report do not form the basis for the award of any other degree to the candidate or to anybody else from this or any other University/Institution.

**Date:**

**Himanshee Singh (2200290140070)**

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Date:

**Dr Vipin Kumar**  
**Associate Professor**  
**Department of Computer Applications**  
**KIET Group of Institutions, Ghaziabad**

**Dr. Arun Kumar Tripathi**  
**Professor & Head**  
**Department of Computer Applications**  
**KIET Group of Institutions, Ghaziabad**

# ABSTRACT

In today's competitive job market, crafting the perfect resume is essential for landing your dream job. Resume builder platform is your go-to web application designed to simplify the resume creation process and give you the tools needed to stand out.

With Resume builder platform, you can effortlessly create, edit, and fine-tune your professional resume. Our user-friendly interface provides a robust WYSIWYG editor and a rich selection of customizable templates, ensuring that your resume reflects your skills and experiences flawlessly.

Worried about content? Our content suggestion engine offers guidance for each section, and our real-time preview feature lets you see your resume taking shape as you type. Once you're satisfied with your resume, Resume builder platform allows you to download it in various formats for maximum compatibility with employer systems.

Managing your job applications is simplified with our intuitive dashboard, where you can easily organize multiple resumes and track your application history. Your privacy and security are our top priorities, and you have full control with our privacy settings to determine who has access to your resume.

Should you ever need assistance, our user support and comprehensive help center are readily available. Start your journey with Resume builder platform today and empower yourself to create a winning resume that opens doors to your career aspirations.

# ACKNOWLEDGEMENT

Success in life is never attained single handedly. My deepest gratitude goes to my thesis supervisor, **Dr. Vipin Kumar** for his guidance, help and encouragement throughout my research work. Their enlightening ideas, comments, and suggestions. Words are not enough to express my gratitude to **Dr. Arun Kumar Tripathi**, Professor and Head, Department of Computer Applications, for his insightful comments and administrative help at various occasions. Fortunately, I have many understanding friends, who have helped me a lot on many critical conditions. Finally, my sincere thanks go to my family members and all those who have directly and indirectly provided me moral support and other kind of help. Without their support, completion of this work would not have been possible in time. They keep my life filled with enjoyment and happiness.

**Himanshee Singh**

# LIST OF CONTENT

# LIST OF TABLES

# LIST OF FIGURES

# CHAPTER 1

## INTRODUCTION

Resume is the first meeting between you and a prospective employer more often now than ever. So, how do you want to be remembered? Wrinkled and unorganized. Neat and structured. Long and boring. Precise and interesting. Companies do not have the time to interview every applicant that is interested in the job. If they did, there would not be a company to work for. They use an eliminating process. That's right - resumes. When a job seeker wants to apply for a job online then generally, he/she needs to attach his/her resume with the email. Online Resume Building System provides the users the popular resume formats & a better way to show their resumes to the employers. A job seeker does not need to attach a resume with every email, he/she just must include the URL of his/her resume and the employer can view the resume online by clicking on the link and can download as well.

### 1.1 Purpose

Purpose of Online Resume Builder is to provide a way to the customers to design their resumes according to their requirements.

a) Creating resumes online.

b) Customizing the look and details.

### 1.2 Scope

Online Resume Builder can be used in accordance with the requirements of the customers. Customers can customize their resumes with their choice of themes & details. The services are hard to be defeated by the competitors as the system is providing the customers exactly what they want.

## 1.3 Overview

Project is related to Online Resume Building.

- This project maintains 1 types of users.
- Users (Customers)

Facilities provided by this project are as follows

- Details of customers are recorded.
- Update of data is easy.
- Flow of information is fast and easy.
- Customers can login to their accounts and view & update their data.

## 1.4 Goals of Proposed System

**Planned approach towards working: -** The working of the system will be well planned and organized. The data will be stored properly in data stores, which will help in retrieval of information as well as its storage.

**Accuracy: -** The level of accuracy in the proposed system will be higher. All operation would be done correctly, and it ensures that whatever information is coming from the system is accurate.

**Reliability: -** The reliability of the proposed system will be high due to the above stated reasons. The reason for the increased reliability of the system is that now there would be proper storage of information.

**No redundancy: -** In the proposed system utmost care would be that no information is repeated anywhere, in storage or otherwise. This would assure economic use of storage space and consistency in the data stored.

**Immediate Retrieval of Information: -** The main objective of proposed system is to provide for a quick and efficient retrieval of information. Any type of information would be available whenever the user requires.

**Immediate storage of information: -** In manual system there are many problems to store & update the large amount of information

**Easy to operate: -** The system should be easy to operate and should be such that it can be easily understood by a new user.

## 1.5 Background

Online Resume Builder is a system which allows the customers to design their resumes in accordance with their requirements. System provides facilities like…

- Customizing the resumes according to the user requirements.
- Editing the design.
- Choosing from latest professional designers.
- Viewed resume notification for the customers.
- Various operations done in the system are as follows…
- Registering customers.
- Access to viewers/employers is allowed for public/shared data only. Writing resumes.
- Editing in design.
- Keeping track of latest formats of resumes. Viewed resume notification

## 1.6 Project Requirements

### 1.6.1 Software Requirements

**Operating System:**

The web application is designed to seamlessly operate on diverse environments, offering compatibility with Windows 7, 8, 10, Linux, or any newer versions of these operating systems.

**Web Browsers:**

Users can access the application through popular web browsers like Google Chrome, Internet Explorer, or any other modern browser, ensuring a user-friendly and accessible experience.

**PDF Viewing:**

Adobe Acrobat Reader is required for users to efficiently view and download any generated PDF documents directly from the application. This enhances the document-sharing capabilities of the platform**.**

**Integrated Development Environment (IDE):**

Developers can utilize Visual Studio Code or any preferred code editor for a smooth development experience. This facilitates efficient coding and collaboration among the development team.

**Backend:**

Firebase Backend-as-a-Service (BaaS): Leveraging Firebase as a Backend-as-a-Service to manage data storage and authentication. Firebase offers a serverless architecture, reducing the need for traditional backend infrastructure management.

**Database:**

Firebase Realtime Database or Firestore : Opting for either Firebase Realtime Database or Firestore to efficiently store and retrieve user data, encompassing profiles and resumes. These NoSQL databases offer real-time synchronization and scalability.

Table 1.1 Software requirements

| Operating system | Software requirement |
|---|---|
| Windows 7,8,10,<br>Linux, or any other higher version | Google chrome,<br>internet explorer, or any web browser |

### 1.6.2 Hardware Requirements:

**Processor:**

The application recommends a processor with a minimum specification of Core i3, i5, or i7 to ensure optimal performance during execution. This enables a smooth user experience while interacting with the application.

**RAM:**

A minimum of 2GB RAM is advised, with higher capacities recommended for efficient handling of various processes within the application. This helps in enhancing the responsiveness and overall performance of the application.

**Graphics Card:**

A dedicated graphics card, such as GTX 1650 (4 GB), is suggested to enhance graphical performance. This is particularly beneficial for applications with graphical elements, ensuring a visually appealing and smooth user interface.

**Internet Connection:**

A stable internet connection is essential for users to experience smooth data retrieval and real-time updates during the resume creation process. This ensures an uninterrupted and responsive user interaction with the application.

**Storage:**

Adequate storage space is required for storing project files, databases, and system backups. This ensures that the application has sufficient resources for data management and recovery.

Table 1.2 Hardware requirements

| **Processor** | **RAM** |
|---|---|
| Core i3,i5 or i7 | 2GB or more |

## 1.7 Technologies used:

This project will be an Internet application to be developed in following tools and technologies.

### Front End

- Web technology: - Django
- Languages used: - JavaScript, HTML5, CSS

### Back End

- Django, MySql

### DJANGO FRAMEWORK

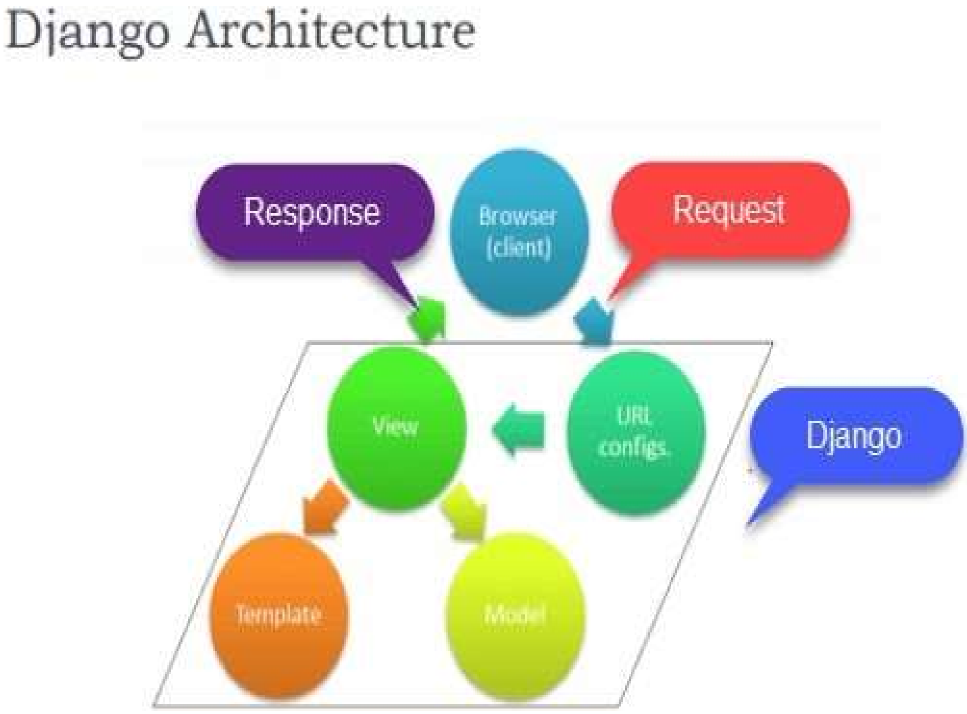
Django may be a high-level Python web framework that encourages rapid development and clean, pragmatic design. Django makes it easier to create better web apps quickly and with less code. Django has built-in support for Ajax, RSS, Caching and various other frameworks.

*Django Logo*

## DJANGO ARCHITECTURE

Django follows a Model-View-Controller (MVC) architecture, which is split upinto three different parts:

- **MODEL:** Model is the logical data structure behind the entire application and is represented by a database (generally relationaldatabases such as MySql).
- **VIEW:** View is the user interface — what you see in your browser when you visit a website. These are represented by HTML/CSS files.
- **CONTROLLER:** Controller is the middleman that connects the view and model together, meaning that it is the one passing data from themodel to the view.

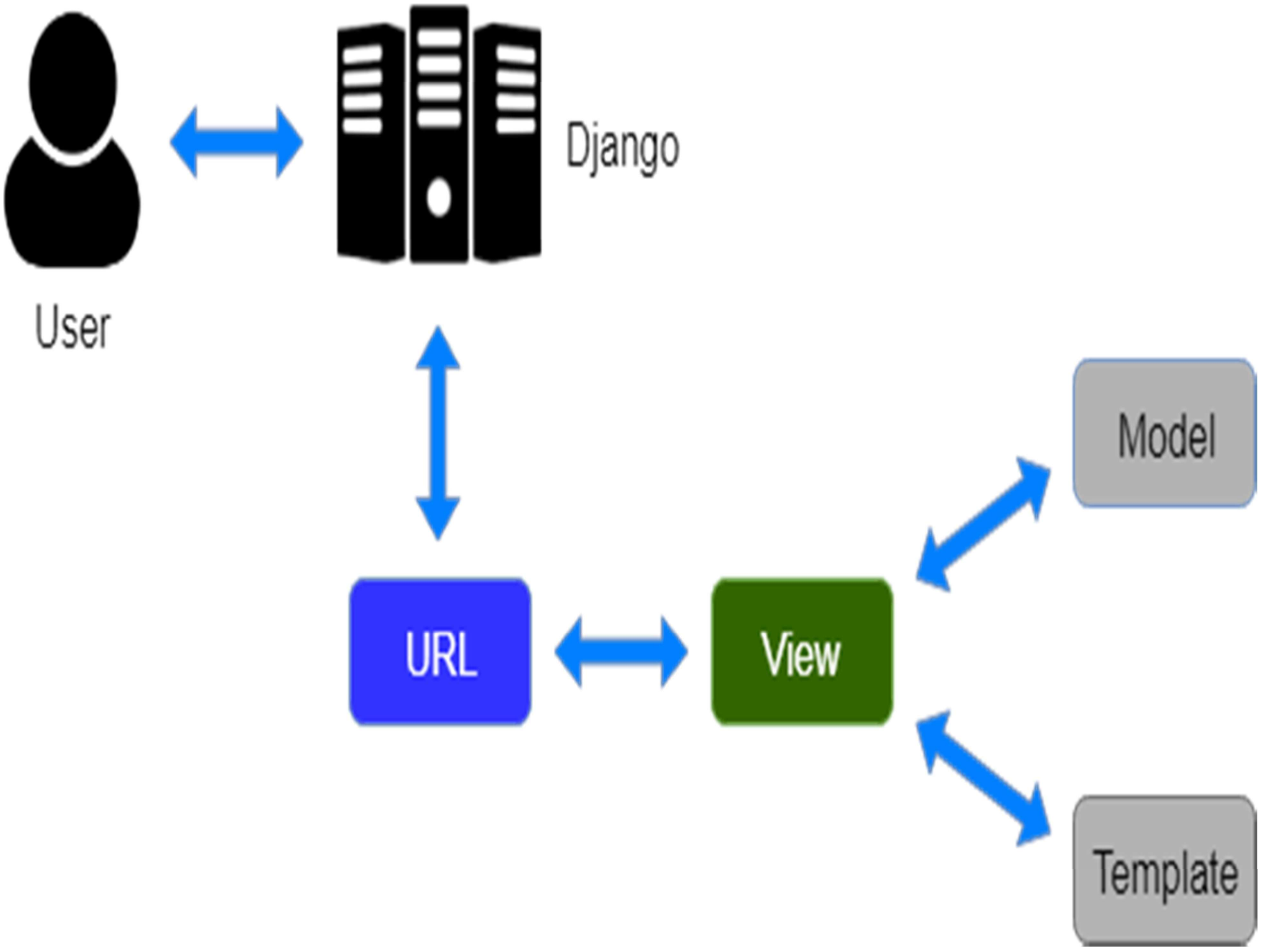


***Django Architecture Diagram***

## DJANGO ADVANTAGES

- Django supports multilingual websites through its built-in internationalization system.
- It reduces many development time and increases productivity.
- Django provides a nice ready-to-use user interface for administrative activities.
- Django is much simpler and the codes are easy when doing it inPython.
- It had been built to slip the applications from imagination to reality during a blink.
- Django has been in the picture for years and in that tenure, ithas emerged as the choice of a lot of businesses for their web applications.
- Django is cross-platform also compatible with the majority of databases making it highly versatile.
- Django because it is powerful enough to create a full-fledged APIin only two or three lines of code.
- Django offers the supply of working with its older versions andmakes use of its older formats and features.

**MVT Pattern**



*MVT Flow Diagram*

The MVT (Model View Template) is a software design pattern. It is a collection of three important components Model View and Template.

- **MODEL:** Model is going to act as the interface of your data. It is responsible for maintaining data. It is the logical data structure behind the entire application and is represented by a database (generally relational databases such as MySql, Postgres).

- **VIEW:** The View is the user interface — what you see in your browser when you render a website. It is represented by HTML/CSS/Javascript and Jinja files.

- **TEMPLATE:** A template consists of static parts of the desired HTML output as well as some special syntax describing how dynamic content will be inserted

## MYSQL

- MySQL has stand-alone clients that allow users to interact directly with a MySQL database using SQL, but more often MySQL is used with other programs to implement applications that need relational database capability. MySQL is a component of the LAMP web application software stack, which is an acronym for Linux, Apache, MySQL, Perl/PHP/Python. MySQL is used by many database-driven web applications, including Drupal, Joomla, phpBB, and WordPress

*MySQL Logo*

## ENVIRONMENT

Most of the framework will run through Visual Studio Code so we use VS Code to run the Front-end Angular framework and for Back-end Sublime Text is used as a run-time environment.

### VISUAL STUDIO CODE

Visual Studio Code is a free source-code editor made  by Microsoft for Windows,  Linux and mac OS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality.

***Visual Studio Code Logo***

# CHAPTER 2

## FEASIBILITY STUDY

Depending on the results of the initial investigation the survey is now expanded to a more detailed

feasibility study. “FEASIBILITY STUDY” is a test of system proposal according to its workability, impact of the organization, ability to meet needs and effective use of the resources. It focuses on these major questions: What are the user’s demonstrable needs and how does a system meet them? What resources are available for given system? What are the likely impacts of the system on the organization? Whether it is worth to solve the problem? During feasibility analysis for this project, following primary areas of interest are to be considered. Investigation and generating ideas about a new system does this. Steps in feasibility analysis Eight steps involved in the feasibility analysis are: Form a project team and appoint a project leader Prepare system flowcharts. Enumerate potential proposed system. Define and identify characteristics of proposed system. Determine and evaluate performance and cost effectiveness of each proposed system. Weight system performance and cost data. Select the best-proposed system. Prepare and report final project directive to management.

### Technical Feasibility

Technical feasibility is the study of resource availability that may affect the ability to achieve an acceptable system. This evaluation determines whether the technology needed for the proposed system is available or not. Can the work for the project be done with current equipment existing software technology & available personal?

Can the system be upgraded if developed?

If new technology is needed, then what can be developed? This is concerned with specifying equipment and software that will successfully satisfy the user requirement. The technical needs of the system may include Front-end and back- end selection An important issue for the development of a project is the selection of suitable front-end and back-end. When we decided to develop the project, we went through an extensive study to determine the most suitable platform that suits the needs of the academy as well as helps in development of the project. The aspects of our study included the following factors. Front-end selection:

It must have a graphical user interface that assists users that are not an advanced user of computer.

Scalability and extensibility.

Flexibility.

Robustness.

According to the organization requirement and the culture

Must provide excellent reporting features with good printing support.

Platform independent.

Easy to debug and maintain.

Event driven programming facility.

Front end must support some popular back end like Ms Access

### Economic Feasibility

Economic justification is generally the “Bottom Line” consideration for most systems. Economic justification includes a broad range of concerns that includes cost benefit analysis. In this we weight the cost and the benefits associated with the candidate system and if it suits the basic purpose of the organization i.e., profit making, the project is making to, the analysis and design phase. The financial and the economic questions during the preliminary investigation are verified to estimate the following:

The cost to conduct a full system investigation. The cost of hardware and software for the class of application being considered. The benefits in the form of reduced cost. The proposed system will give the minute information, as a result the performance is improved which in turn may be expected to provide increased profits. This feasibility checks whether the system can be developed with the available funds. Online Resume Builder does not require enormous amount of money to be developed. This can be done economically if planned judicially, so it is economically feasible. The cost of project depends upon the number of man-hours required.

### Operational Feasibility

It is mainly related to human organizations and political aspects. The points to be considered are:

What changes will be brought with the system? What organization structures are disturbed?

What new skills will be required? Do the existing staff members have these skills? If not, can they be trained in due course of time? The system is operationally feasible as it very easy for the End users to operate it

### Schedule Feasibility

Time evaluation is the most important consideration in the development of project. The time schedule required for the development of this project is very important since more development time effect machine time, cost and cause delay in the development of other systems. Online Resume Builder can be developed in the considerable amount of time.

# CHAPTER 3

## ARCHITECTURAL DESIGN

Architectural design represents the structure of data and program components that are required to build a computer-based system. It considers the architectural style that the system will take, the structure and properties of the components that constitute the system, and the interrelationships that occur among all architectural components of a system.

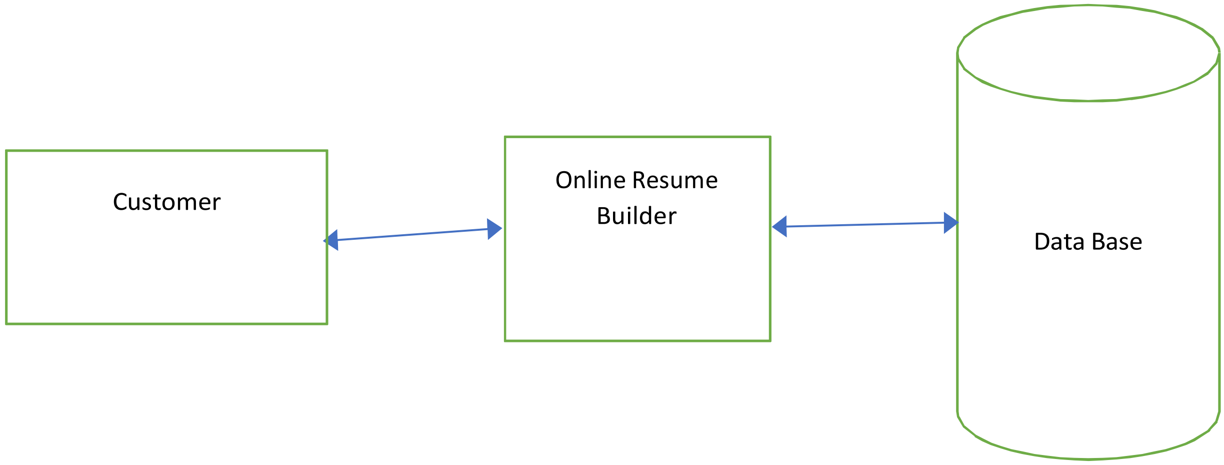


Figure 3.1 System diagram

## ER Diagram

The object/relationship pair is the cornerstone of the data model. These pairs are represented graphically using E-R diagrams. A set of primary components are identified for the ERD: data objects, attributes, relationships, and various type indicators. The primary purpose of ERD is to represent data objects and their relationships

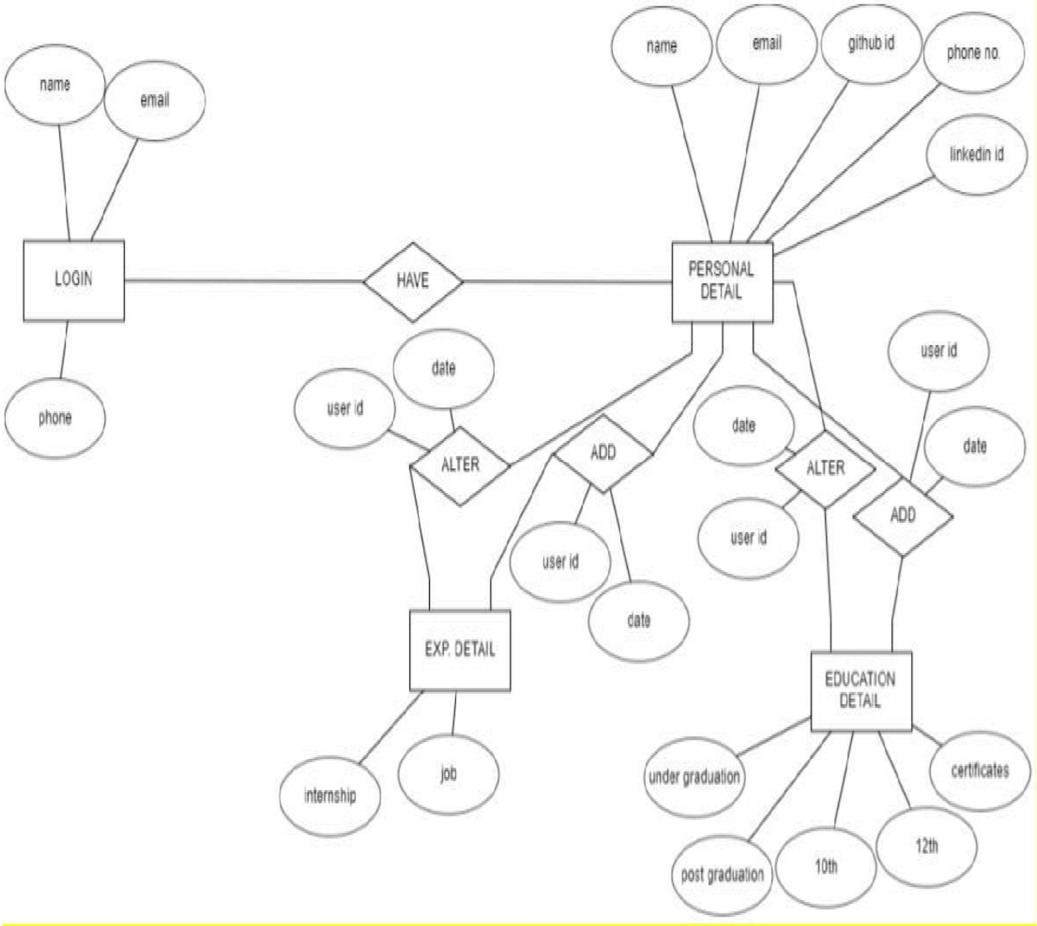


Figure 3.2 E-R Diagram

**DFD (Data Flow Diagram)**

The data flow diagram enables the software engineer to develop models of the information domain and functional domain at the same time. As the DFD is refined into greater level of detail, the analyst performs an implicit functional decomposition of the system. At the same time, the DFD refinement results in corresponding refinement of data as it moves through the processes that embody the application

0 level DFD:

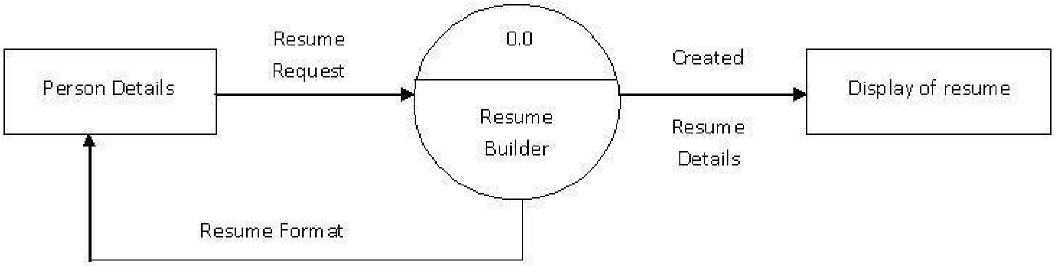


Figure 3.3DFD

1 level DFD:

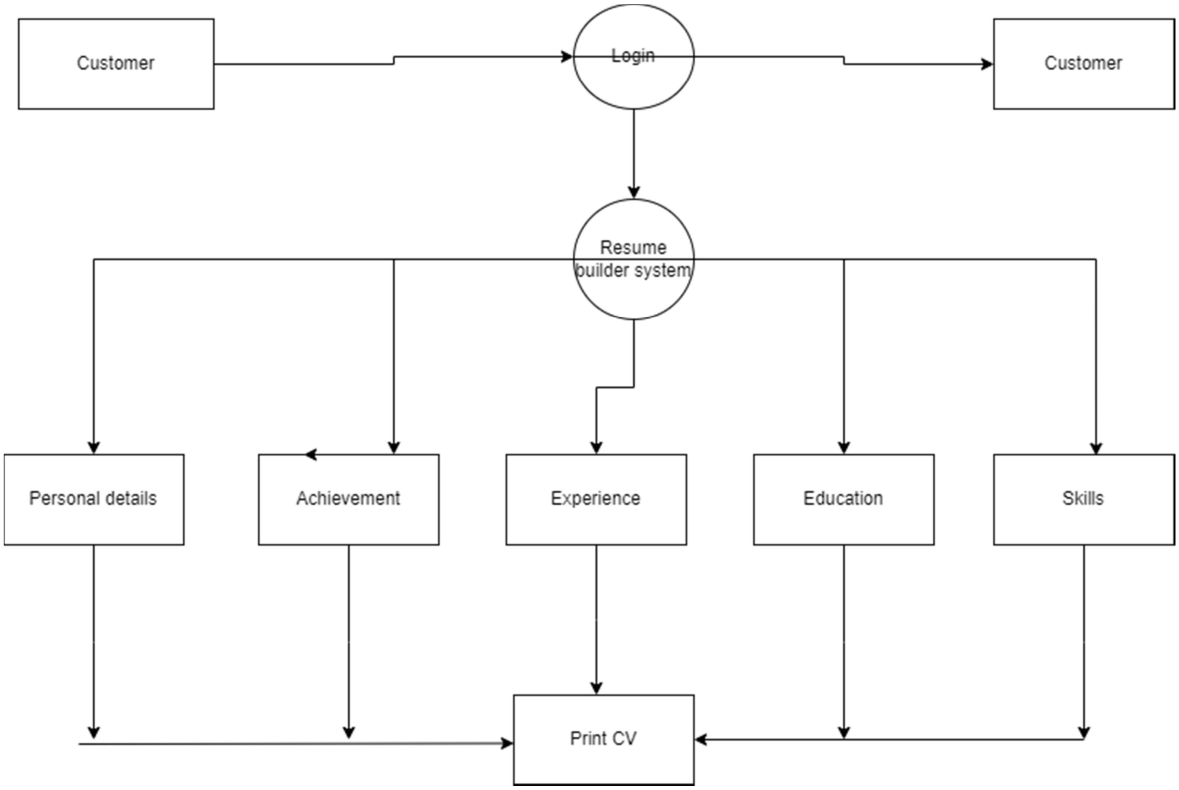


Figure 3.4 1 Level DFD

## Use case Diagram

Use Case Model is an approach that is a combination of text and pictures to improve the understanding of requirements. A use case model' is describing the complete functionality of a system by identifying how everything that is outside the system interacts with it.

A Use Case Diagram is given below that relates to this application.

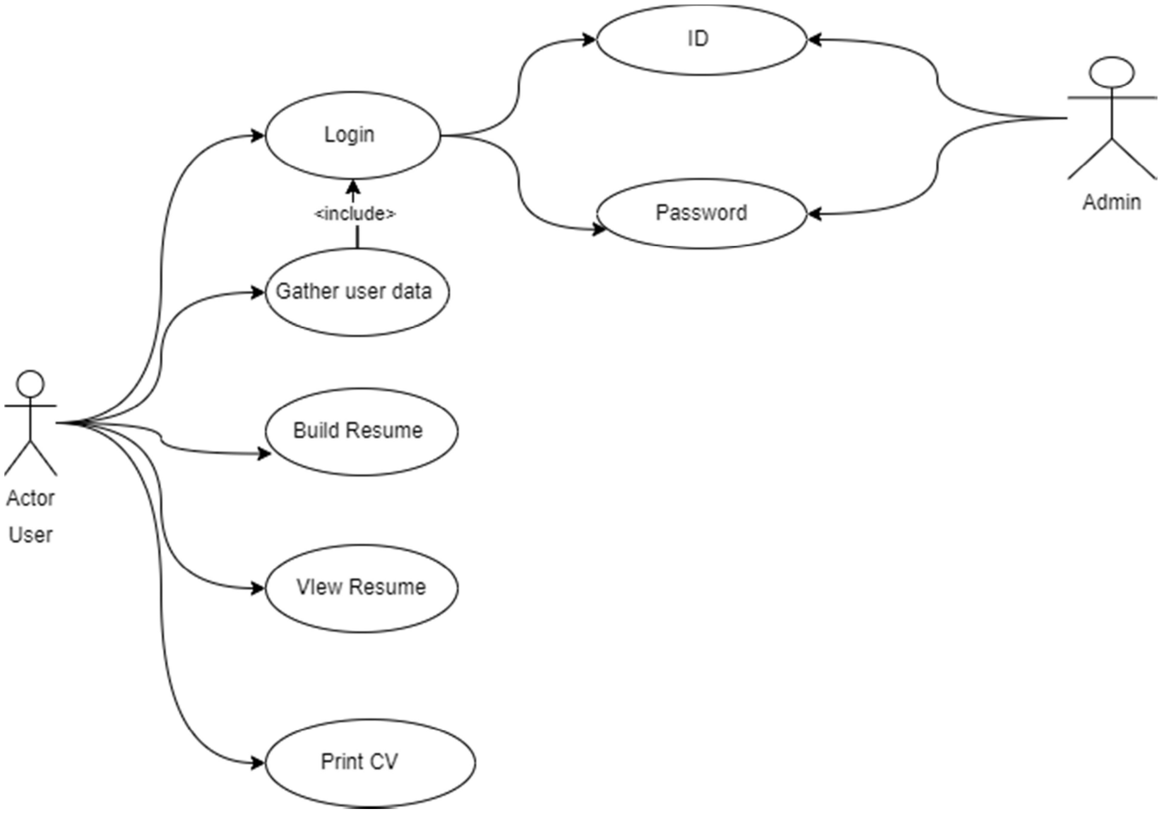


Figure 3.5 Use Case Diagram

# CHAPTER 4

## REQUIREMENT ANALYSIS

### Functional Requirements

### User Registration and Login

Validate user input for registration fields, such as ensuring a valid email format and enforcing password strength requirements.

Implement secure authentication mechanisms to protect user accounts.

### Resume Creation

The system should provide a user-friendly interface for creating a resume.

Users should be able to enter information in multiple fields, including:

Personal Information (e.g., name, contact details, address)

Education (e.g., degrees, institutions, graduation years)

Work Experience (e.g., job titles, companies, employment periods)

Skills (e.g., technical skills, soft skills)

Additional Sections (e.g., certifications, projects)

### Real-time Preview

As users input information, the system should dynamically generate a preview of the resume.

Users should be able to view and review the resume in real-time to ensure accuracy and formatting.

### Printing Functionality

Users should be able to generate a printable version of their resume.

The printable version should include all entered information in a well-formatted layout.

### Responsive Design

Ensure the website is responsive, providing a consistent and user-friendly experience across various devices and screen sizes.

## Non-Functional Requirements

### Performance

The website should exhibit responsive performance, ensuring quick loading times and smooth user interactions.

Response times for critical operations (e.g., saving, previewing) should be within acceptable limits.

### Compatibility

The platform should be compatible with various web browsers, operating systems, and devices to ensure a broad user reach.

### Usability

The user interface should be intuitive and user-friendly, catering to users with varying levels of technological proficiency.

### Scalability

The system should be designed to scale horizontally to accommodate potential increases in user numbers and data volume.

Scaling efforts should be seamless and not disrupt ongoing user activities.

### Error Handling

The system should gracefully handle errors, providing clear and  user-friendly messages.

# Chapter 5

# Implementation

## Code: -

## 5.1 Sign up:

```
<title>Online Cv</title>

<link rel="stylesheet" type="text/css" href="/static/css/bootstrap.min.css">
<link rel="stylesheet" type="text/css" href="/static/css/sb-admin.css">
<link rel="stylesheet" type="text/css" href="/static/css/sb-admin.min.css">
<link rel="stylesheet" type="text/css" href="/static/css/index.css">

<nav class="navbar navbar-expand navbar-dark bg-dark static-top">

  <a class="navbar-brand mr-1" href="/" style="color: #FFFF00; font-weight: bold;">Online Cv Generator</a>

  <p class="d-none d-md-inline-block  ml-auto mr-0 mr-md-3 my-2 my-md-0"></p>
  <ul class="navbar-nav ml-auto ml-md-0">
    <a href="/" id="btn_link_space"><button class="btn_link">Home</button></a>
    <a href="/login/" id="btn_link_space"><button class="btn_link">Sign In</button></a>
    <a href="/register/" id="btn_link_space"><button class="btn_link">Sign Up</button></a>
  </ul>
</nav>
```

```
<div id="content-wrapper" class="mt-4" style="padding-top: 1%;">
     <div class="container-fluid">
       <div class="col-md-12">



        <div class="row">
        <div class="col-md-4">
           </div>


        <div class="card col-md-3">
            <h1></h1>
            <form method="POST" action="">
                <input type="hidden" name="csrfmiddlewaretoken"
value="WmLgIoH5KoCKYmW4PUJXDIaQAAuN5H3vJbMWNEuyv63FU7x7hMBZNaSf9HWA
Dvf3">
<div class="form-group">
                    <label>Username</label>
                    <input class="form-control" type="text" name="username" placeholder="Enter
Username" required/>
              </div>
            <div class="form-group">
               <label>Email</label>
               <input class="form-control" type="email" name="email" placeholder="Enter Email"
required/>
            </div>

            <div class="form-group">
               <label>Password</label>
               <input class="form-control" type="password" name="password" placeholder="Enter
Password" required/>
            </div>

            <div class="form-group">
               <button class="loginbtn">Sign Up</button>
            </div>

            <a href="/login/">Have account ? Sign In here</a>


         </form>
        </div>

        <div class="col-md-4">
        </div>

      </div>

     </div>

     <footer class="footer fixed-bottom" style="width: 100%; height: 8%;">

   <div class="">
      <p class="mt-3 text-center">&copy; Learn Python 2022</p>

   </div>
</footer>
    </div>

     </div>
```

## 5.2 Login

```
  {% include 'core/header.html' %}
{% load static %}

<div id="content-wrapper" class="mt-4"
style="padding-top: 1%;">
    <div class="container-fluid">
      <div class="col-md-12">


       <div class="row">
       <div class="col-md-4">
          </div>

       <div class="card col-md-3">
           <h1></h1>
           <form method="POST" action="">
               {% csrf_token %}

               <div class="form-group">
                   <label>Username</label>
                   <input class="form-control" type="text"
name="username" placeholder="Enter Username"
required/>
               </div>

               <div class="form-group">
                   <label>Password</label>
                   <input class="form-control"
type="password" name="password"
placeholder="Enter Password" required/>
               </div>

               <div class="form-group">
                   <button class="loginbtn">Sign
In</button>
               </div>
               <a href="{% url 'reg-form' %}">Don't Have
account ? Sign Up here</a>


               {% if messages %}
               {% for message in messages %}

               <p class="text-danger text-center"
style="font-weight: bold;">{{ message }}</p>
               {% endfor %}
               {% endif %}
           </form>
          </div>

          <div class="col-md-4">
          </div>

       </div>\

      </div>

      {% include 'core/footer.html' %}
    </div>
</div>




{% include 'core/header.html' %}
{% load static %}

<div id="content-wrapper" class="mt-4"
style="padding-top: 1%;">
    <div class="container-fluid">
      <div class="col-md-12">


       <div class="row">
       <div class="col-md-4">
          </div>

       <div class="card col-md-3">
           <h1></h1>
           <form method="POST" action="">
               {% csrf_token %}

               <div class="form-group">
                   <label>Username</label>
                   <input class="form-control" type="text"
name="username" placeholder="Enter Username"
required/>
               </div>

               <div class="form-group">
                   <label>Password</label>
                   <input class="form-control"
type="password" name="password"
placeholder="Enter Password" required/>
               </div>

               <div class="form-group">
                   <button class="loginbtn">Sign
```

```
In</button>
                </div>
                <a href="{% url 'reg-form' %}">Don't Have
account ? Sign Up here</a>


                {% if messages %}
                {% for message in messages %}

                <p class="text-danger text-center"
style="font-weight: bold;">{{ message }}</p>
                {% endfor %}
                {% endif %}

            </form>
           </div>

           <div class="col-md-4">
           </div>

        </div>

       </div>

       {% include 'core/footer.html' %}
     </div>
</div>
```

## 5.3 Personal

```
        from django.http import JsonResponse
from your_app.models import Profile
from django.core.exceptions import ObjectDoesNotExist

def fetchProfile(request):
    id = request.POST.get('id')
    print('Cv ID is', id)

    try:
        user_profile = Profile.objects.get(cv_id=id)
    except ObjectDoesNotExist:
        return JsonResponse({'error': 'Profile not found'}, status=404)

    user_profile_data = {
        'fname': user_profile.fname,
        'mname': user_profile.mname,
        'lname': user_profile.lname,
        'email': user_profile.email,
        'phone': user_profile.phone,
        'bio': user_profile.bio,
        'dob': user_profile.dob,
        'country': user_profile.country,
        'region': user_profile.region,
        'occupation':  user_profile.occupation
    }

    return  JsonResponse(user_profile_data)
```

## 5.4 Education

```
from django.db import models
from your_app.models import Profile

class Education(models.Model):
    profile = models.ForeignKey(Profile, on_delete=models.CASCADE)
    degree = models.CharField(max_length=100)
    institution = models.CharField(max_length=150)
    field_of_study = models.CharField(max_length=150)
    start_date = models.DateField()
    end_date = models.DateField(null=True, blank=True)
    # Add other fields as needed

    def __str__(self):
        return f"{self.degree} in {self.field_of_study} from {self.institution}"
```

## 5.5 Experience

```
# experience/views.py
from django.shortcuts import get_object
from django.http import JsonResponse
from .models import Experience

def fetch_experience(request, experience_id):
    experience = get_object(Experience, pk=experience_id)
    experience_data = {
        'title': experience.title,
        'description': experience.description,
        'start_date': str(experience.start_date),
        'end_date': str(experience.end_date) if experience.end_date else None,
        # Add more fields as needed
    }
    return JsonResponse(experience_data)
```

## 5.6 Project

```
<!-- PROJECT 1 -->

        <div class="form-group">
            <label for="experience1">Project 1 :</label>
        </div>
        <div class="com">
            <div class="form-group ">
                <label for="company1">Project Title :</label>
                <input type="text" id="company1" name="project1">
            </div>
            <div class="form-group">
                <label for="post1">Duration :</label>
                <input type="text" id="post1" name="durat1">
            </div>
        </div>
        <div class="form-group">
            <label for="description1">Description :</label>
            <input type="text" id="duration1" name="desc1">
        </div>

        <!-- PROJECT 2 -->

        <div class="form-group">
            <label for="experience1">Project 2 :</label>

        </div>
        <div class="com">
            <div class="form-group ">
                <label for="company1">Project Title :</label>
                <input type="text" id="company1" name="project2">
            </div>
            <div class="form-group">
                <label for="post1">Duration :</label>
                <input type="text" id="post1" name="durat2">
            </div>
        </div>
        <div class="form-group">
            <label for="description1">Description :</label>
            <input type="text" id="duration1" name="desc2">
        </div>



        <!-- ======================EXPERICNECE  SECTION==================== -
->
        <div class="form-group">
            <label for="experience1">Experience 1:</label>
        </div>
        <div class="com">
            <div class="form-group ">
                <label  for="company1">Company  Name  :</label>
                <input  type="text"  id="company1"  name="company1">
            </div>
            <div class="form-group">
                <label for="post1">Post :</label>
                <input type="text" id="post1" name="post1">
            </div>
            <div class="form-group">
                <label for="duration1">Duration :</label>
                <input type="text" id="duration1" name="duration1">
            </div>
        </div>
        <div class="form-group ">
            <label for="description1">Description :</label>
            <div class="op">
                <input type="text" id="duration1" name="lin11">
            </div>
        </div>
```

```
<div class="form-group">
    <label for="experience1">Experience 2:</label>
</div>
<div class="com">
    <div class="form-group ">
        <label  for="company1">Company  Name  :</label>
        <input  type="text"  id="company1"  name="company2">
    </div>
    <div class="form-group">
        <label for="post1">Post :</label>
        <input type="text" id="post1" name="post2">
    </div>
    <div class="form-group">
        <label for="duration1">Duration :</label>
        <input type="text" id="duration1" name="duration2">

    </div>
</div>
<div class="form-group">
    <label for="description1">Description :</label>
    <input type="text" id="duration1" name="lin21">
</div>


<!--  =========================ACHIEVEMENT================  -->


<div class="form-group">
    <label for="experience1">Achievement :</label>
</div>
<div class="form-group ">
    <label for="company1">First :</label>
    <input type="text" id="company1" name="ach1">
</div>
<div class="form-group">
    <label for="post1">Second :</label>
    <input type="text" id="post1" name="ach2">
```

```
                </div>
                <div class="form-group">
                    <label for="duration1">Third :</label>
                    <input type="text" id="duration1" name="ach3">
                </div>
                <button type="submit" class=" btn1 ">Submit</button>
            </div>
    </form>
</body>
</html>
```

### 5.7 url.py

```
from django.contrib import admin
from  django.urls  import  path
from home.views import *

urlpatterns = [
    path('', home, name = 'home'),
    path('resume/', gen_resume, name = 'resume'),
    path("admin/", admin.site.urls),
]
```

## 5.8 Setting.py

```
INSTALLED_APPS = [
    "django.contrib.admin",
    "django.contrib.auth",
    "django.contrib.contenttypes",
    "django.contrib.sessions",
    "django.contrib.messages",
    "django.contrib.staticfiles",
    "home",
]
```

## Deployement of the Project

### Run these commands to apply the migrations:

```
python3  manage.py  makemigrations
python3 manage.py migrate
```

### Run the server with the help of following command:

```
python3 manage.py runserver
```

# CHAPTER 6

# TESTING

## 6.1 Introduction

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionalities of components, sub-assemblies, and/or a finished product it is the process of exercising software with the intent of ensuring that the software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

## 6.2 Types of Testing

### 6.2.1 Unit Testing

Unit testing focuses verification effort on the smallest unit of software design, the module. The unit testing, we have is white box oriented and some modules the steps are conducted in parallel.

### 6.2.2 Integration Testing

Testing is done for each module. After testing all the modules, the modules are integrated and testing of the final system is done with the test data, specially designed to show that the system will operate successfully in all its aspects conditions. Thus, the system testing is a confirmation that all is correct and an opportunity to show the user that the system works.

The purpose of integration testing is to verify functional, performance and reliability requirements placed on major design items. These "design items", i.e. assemblages (or groups of units), are exercised through their interfaces using black box testing, success and error cases being simulated via appropriate parameter and data inputs. Simulated usage of shared data areas and inter-process communication is tested and individual subsystems are exercised through their input interface.

### System Testing

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration point.

# CHAPTER 7

# SNAPSHOTS

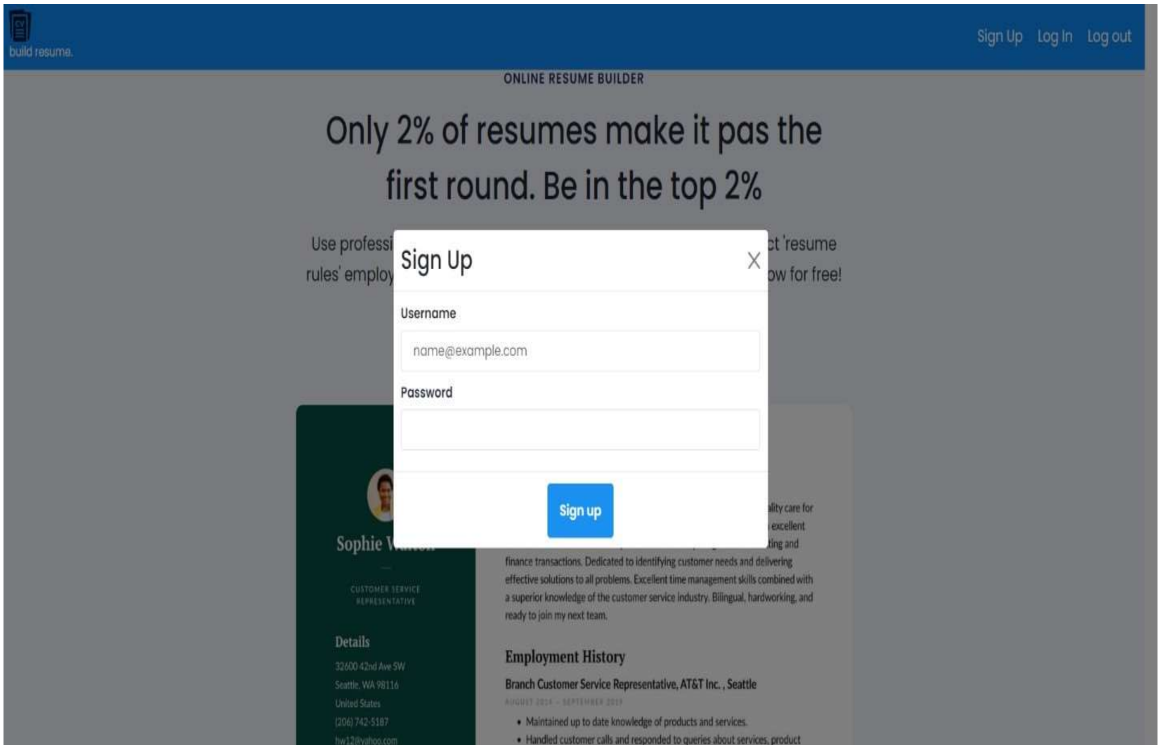
## 7.1 Snap Shot

### Signup

The Signup screen in our project is a crucial component that facilitates the user registration process, enabling individuals to create personalized accounts on our platform. The screen is presented as a modal, providing a focused and interactive environment for users to input their information.

User Registration

The primary purpose of the Signup screen is to facilitate user registration, allowing individuals to create accounts tailored to their preferences and needs.



Only 2% of resumes make it pas the first round. Be in the top 2%

Use professi … ct 'resume rules' employ … ow for free!

… ality care for … excellent … ting and finance transactions. Dedicated to identifying customer needs and delivering effective solutions to all problems. Excellent time management skills combined with a superior knowledge of the customer service industry. Bilingual, hardworking, and ready to join my next team.

- Maintained up to date knowledge of products and services.
- Handled customer calls and responded to queries about services, product

Figure .7.1 User registration

### 7.2 Login

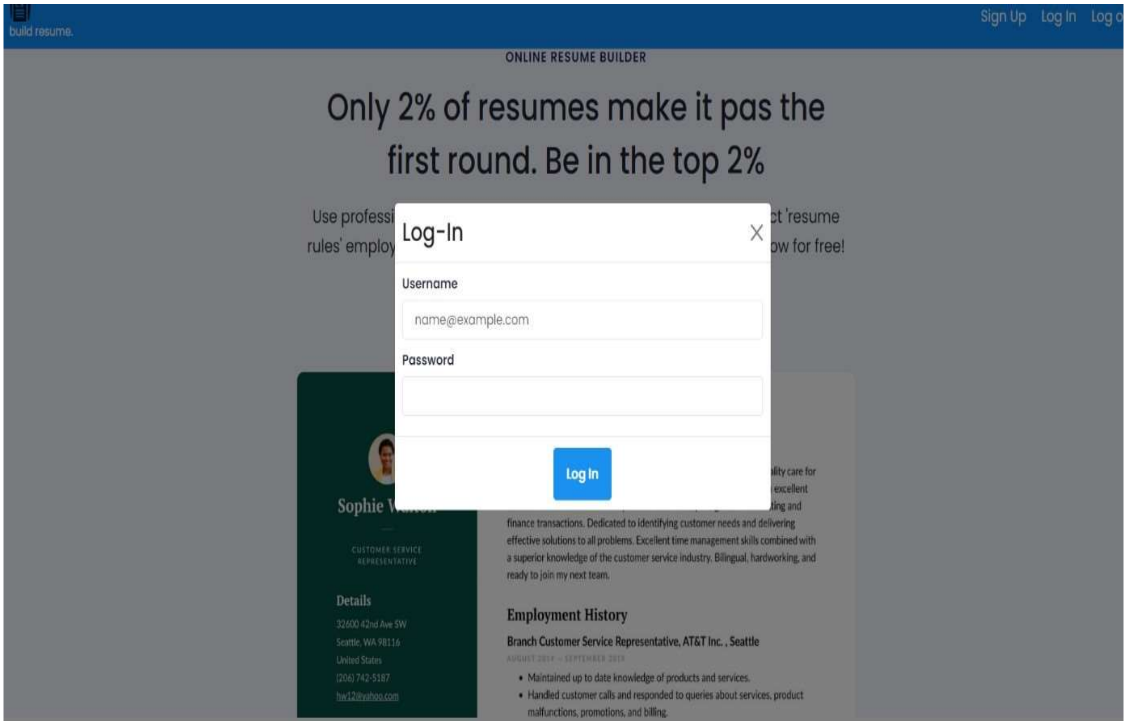
The Login screen is a pivotal component in our project, serving as the entry point for users to access their accounts on our platform.

#### User Authentication

The primary purpose of the Login screen is to authenticate users, allowing them secure access to their personalized accounts

#### Username/Email and Password Input:

Users are required to input their registered username/email and corresponding password for account authentication.



Only 2% of resumes make it pas the first round. Be in the top 2%

Use professi … ct 'resume rules' employ … ow for free!

32600 42nd Ave SW
Seattle, WA 98116
United States
(206) 742-5187
hw12@yahoo.com

… ality care for … excellent … ting and finance transactions. Dedicated to identifying customer needs and delivering effective solutions to all problems. Excellent time management skills combined with a superior knowledge of the customer service industry. Bilingual, hardworking, and ready to join my next team.

Branch Customer Service Representative, AT&T Inc. , Seattle

AUGUST 2014 — SEPTEMBER 2019

- Maintained up to date knowledge of products and services.
- Handled customer calls and responded to queries about services, product malfunctions, promotions, and billing.

Figure .7.2 Login

### 7.3 Personal details

The Personal Information screen is a crucial segment of our project, offering users the ability to input and manage their essential details, contributing to a comprehensive and personalized profile.

**Key Input Fields:**

**First Name, Last Name, and Middle Name:**

Users are prompted to provide their full name, including first, last, and middle names, fostering a complete and accurate representation of their identity.

**Designation:**

A designated field for users to specify their professional title or designation, enabling a detailed and professional profile presentation.

**Address:**

Users can input their current residential or professional address, contributing to a comprehensive understanding of their geographical location.

**Email:**

The email input field allows users to update or verify their email address, ensuring accurate communication and account management.

**Phone Number:**

A dedicated field for users to provide their phone number, facilitating easy communication and enhancing account security.

**Summary:**

Users are invited to craft a summary that highlights key aspects of their professional or personal identity. This field serves as a concise overview for other platform users.

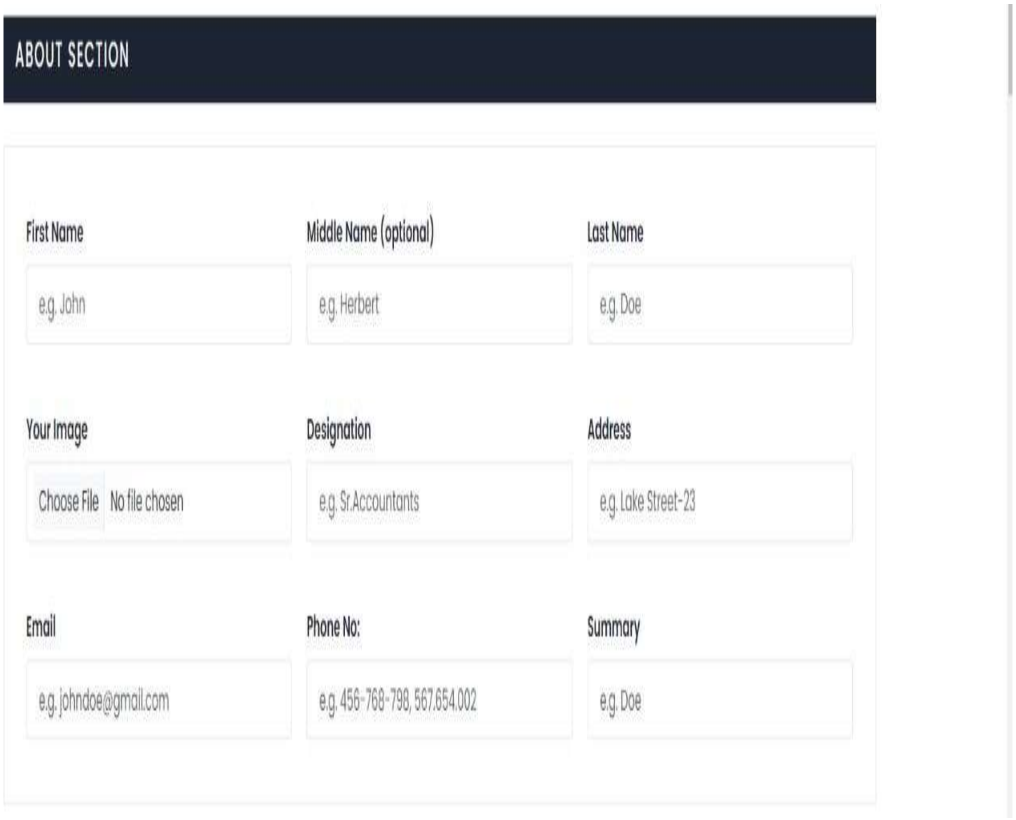


Figure .7.3 Personal details

### 7.4 Education

The Education Information screen is a vital component of our project, allowing users to input and manage their educational background, providing a comprehensive overview of their academic achievements.

**Key Input Fields:**

**School:**

Users can specify the name of the educational institution or school where they pursued their academic studies.

**Degree:**

A dedicated field allows users to input the specific degree or qualification they obtained during their time at the specified educational institution.

**Details:**

This field accommodates additional details related to the user's academic experience, such as major, specialization, or any notable achievements during their studies.

**Start Date and End Date:**

Users are prompted to provide the start and end dates of their enrolment at the educational institution, offering a chronological representation of their academic journey.

**City:**

A designated field for users to specify the city where the educational institution is located, providing context to their academic history.

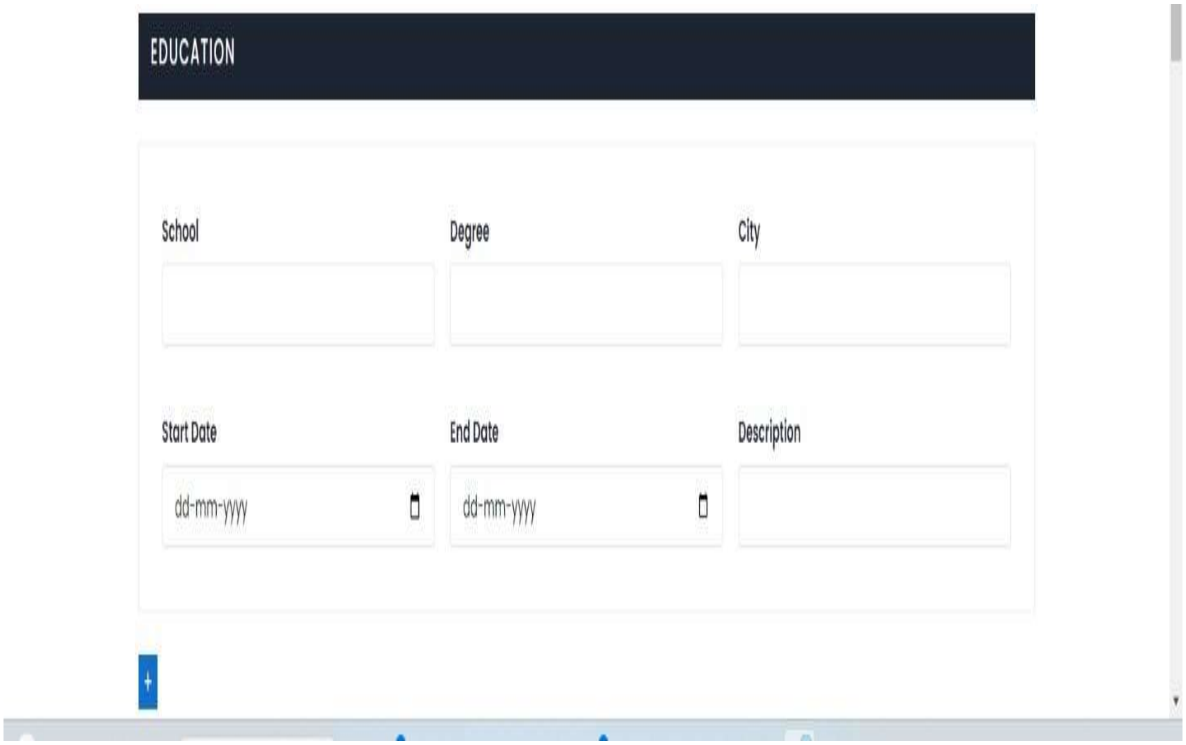


Figure .7.4 Education

### 7.5 Experience

The Experience Information screen is an integral feature of our project, empowering users to input and manage their professional work experience, providing a detailed overview of their career accomplishments.

**Key Input Fields:**

**Title:**
Users can specify their job title or role during their tenure at a particular organization, providing clarity on their professional responsibilities.

**Organization:**
A dedicated field enables users to input the name of the organization or company where they gained their professional experience.

**Start Date and End Date:**
Users are prompted to provide the start and end dates of their employment, presenting a chronological representation of their professional journey.

**Location:**
A designated field for users to specify the location or city where the organization is based, providing context to their professional experience.

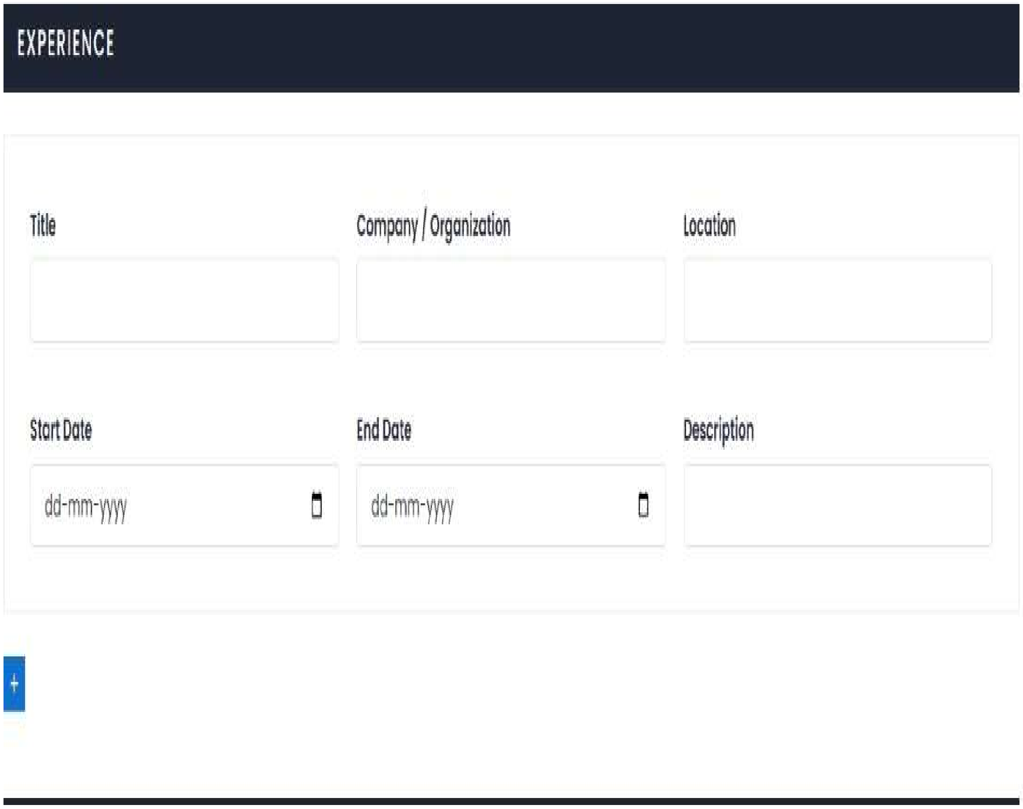


Figure .7.5 Experience

### 7.6 Project info

The Project Information screen is a key feature within our project, allowing users to input and manage details about their professional projects. This section offers users the opportunity to showcase their project experience and contributions.

**Key Input Fields:**

**Project Name:**

Users can provide a descriptive title for their project, offering clarity on the nature or purpose of the undertaken endeavour.

**Project Description:**

A dedicated field allows users to articulate a comprehensive project description, detailing the objectives, scope, and outcomes of the project.

**Project Link:**

Users can include a link to the project, providing a direct reference for interested parties to explore further details or view the project in action.

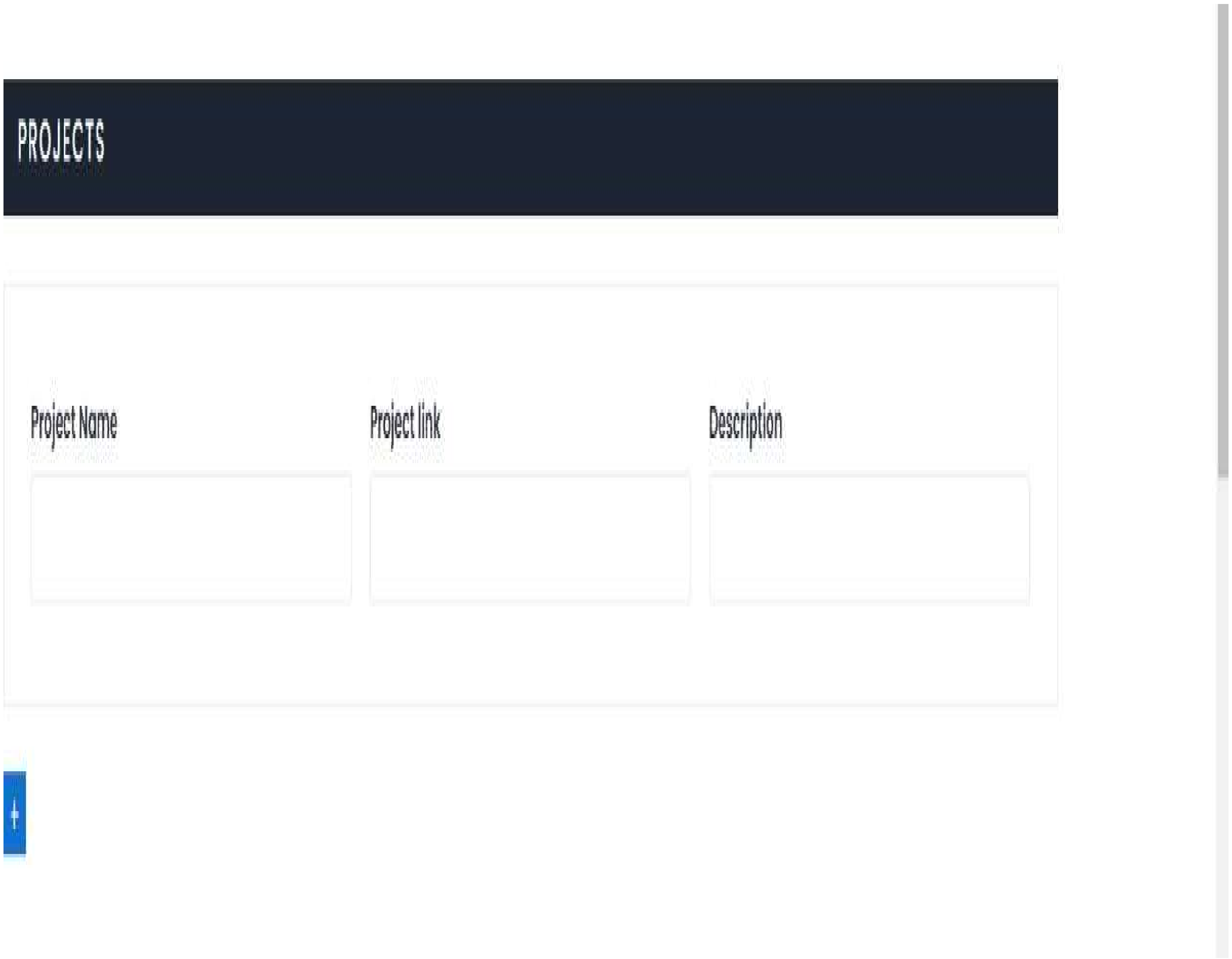
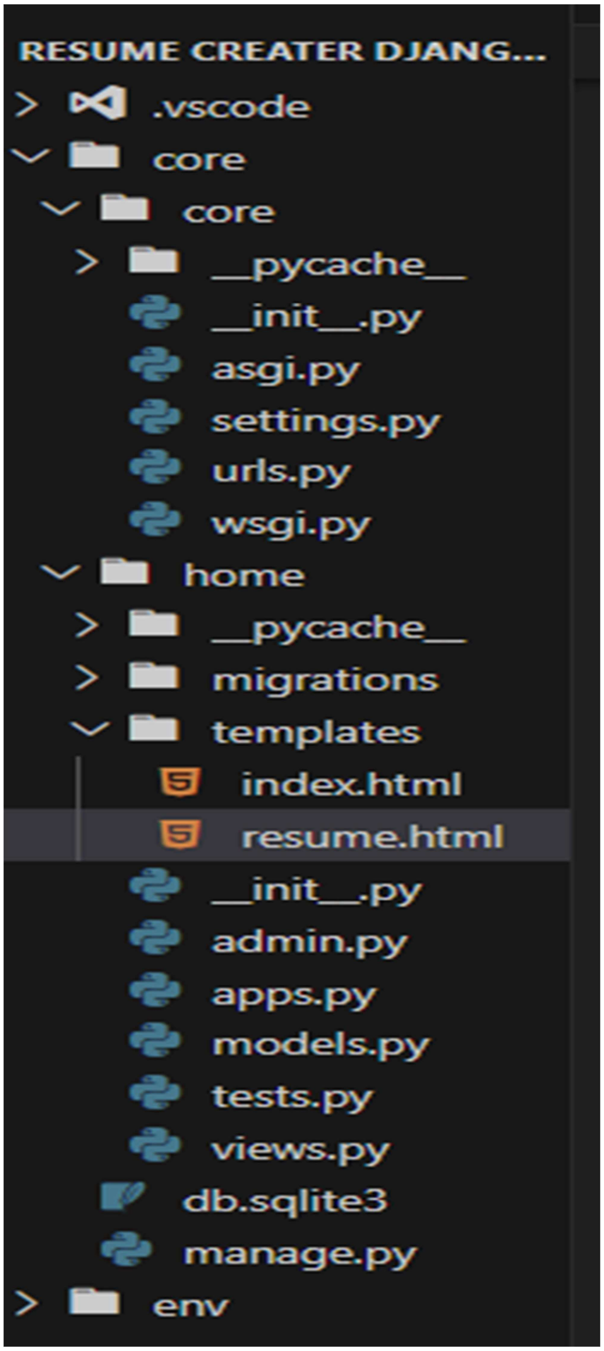


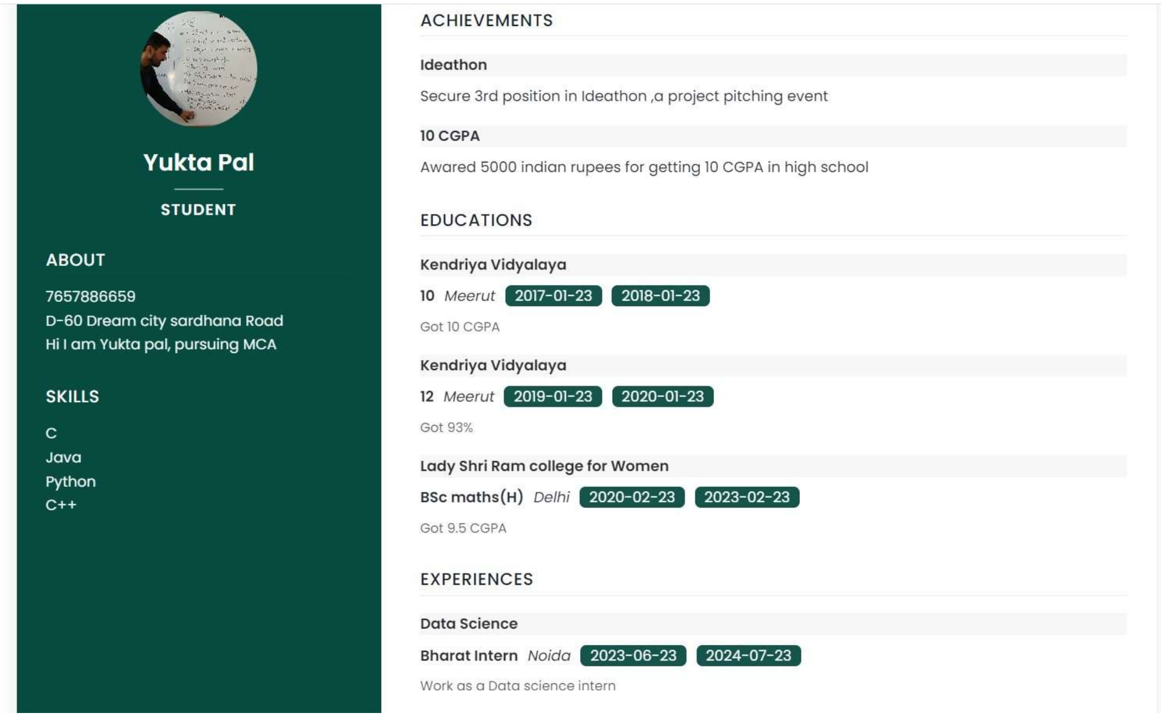
Figure .7.6 Project information

**File Structure**



```
RESUME CREATER DJANG...
> .vscode
∨ core
  ∨ core
    > __pycache__
      __init__.py
      asgi.py
      settings.py
      urls.py
      wsgi.py
  ∨ home
    > __pycache__
    > migrations
    ∨ templates
        index.html
        resume.html
      __init__.py
      admin.py
      apps.py
      models.py
      tests.py
      views.py
    db.sqlite3
    manage.py
> env
```

### 7.7 Final Outcome



Yukta Pal

STUDENT

ABOUT

7657886659

D-60 Dream city sardhana Road

Hi I am Yukta pal, pursuing MCA

SKILLS

C

Java

Python

C++

ACHIEVEMENTS

Ideathon

Secure 3rd position in Ideathon ,a project pitching event

10 CGPA

Awared 5000 indian rupees for getting 10 CGPA in high school

EDUCATIONS

Kendriya Vidyalaya

10 *Meerut* 2017-01-23 2018-01-23

Got 10 CGPA

Kendriya Vidyalaya

12 *Meerut* 2019-01-23 2020-01-23

Got 93%

Lady Shri Ram college for Women

BSc maths(H) *Delhi* 2020-02-23 2023-02-23

Got 9.5 CGPA

EXPERIENCES

Data Science

Bharat Intern *Noida* 2023-06-23 2024-07-23

Work as a Data science intern

Figure .7.7 Final Snapshot

# CHAPTER-8

# CONCLUSION AND FUTURE ENHANCEMENTS

## 8.1 Conclusion:

- In the journey of developing and enhancing our platform, we've strived to create a robust and user-centric environment for individuals to showcase their professional identity and accomplishments. The evolution of this project has been guided by the principles of user experience, innovation, and adaptability to the dynamic needs of our users.

- Throughout the development process, we have focused on providing a seamless and intuitive interface, ensuring that users can effortlessly navigate and interact with the platform. The incorporation of features such as instant profile interaction, diverse templates, and real-time collaboration reflects our commitment to empowering users in expressing their uniqueness and professional journey.

- Our platform not only serves as a repository for users' resumes and project portfolios but also as a dynamic space for collaboration, skill endorsement, and continuous improvement. The introduction of features like project analytics, customizable templates, and a responsive dashboard aims to enrich the user experience and foster a sense of community among our diverse user base.

- As we conclude this phase of development, we remain dedicated to ongoing improvements and innovations. The feedback and engagement of our users will continue to be invaluable in shaping the future direction of the platform. We envision a space that not only meets the current expectations of our users but also anticipates and adapts to the evolving landscape of professional networking and self-presentation.

- Our commitment extends beyond providing a platform; it lies in creating a supportive and empowering ecosystem where individuals can truly shine. We look forward to the exciting journey ahead, filled with further enhancements, collaborations, and the collective success of our user community.

## 8.2 Future Enhancements

**Instant Editing Mode:**

Introduce an instant editing mode that allows users to update their information directly from the dashboard. This seamless interaction reduces the need for navigating to specific edit screens, enhancing user convenience.

**Template Library:**

Create a library of diverse profile templates for users to choose from. Include templates tailored for different professional fields or personal preferences, allowing users to switch between them seamlessly.

**History Tracking:**

Implement a history tracking feature that records changes made to the profile over time. Users can review and revert to previous versions of their profiles if needed.

**Skills Endorsements:**

Implement a skills endorsement system where users can endorse each other's skills based on their project contributions. This would enhance the credibility of showcased skills.

# REFERENCES

5 Buildings web application
https://youtu.be/ojFkXf-iGXo?si=bJuTp8d2jtxGrSN4

6 OpenAI Chat Platform
https://chat.openai.com/

7 W3Schools - Online Web Tutorials
https://www.w3schools.com/

8 JavaTpoint - HTML and JavaScript Tutorial
https://www.javatpoint.com/html-javascript

9 Real Python Django Tutorial: : https://realpython.com/tutorials/django/

10 Django Documentation: https://docs.djangoproject.com/en/stable/

11 Django Testing Documentation: https://docs.djangoproject.com/en/stable/topics/testing/

12 "Two Scoops of Django 3.x: Best Practices for the Django Web Framework" by Daniel Roy Greenfeld and Audrey Roy Greenfeld.