INTERNATIONAL ISLAMIC UNIVERSITY CHITTAGONG

TOOLS & TECHNOLOGIES OF INTERNET

PROJECT REPORT

SPRING'25 (5BF)

SUBMITTED TO

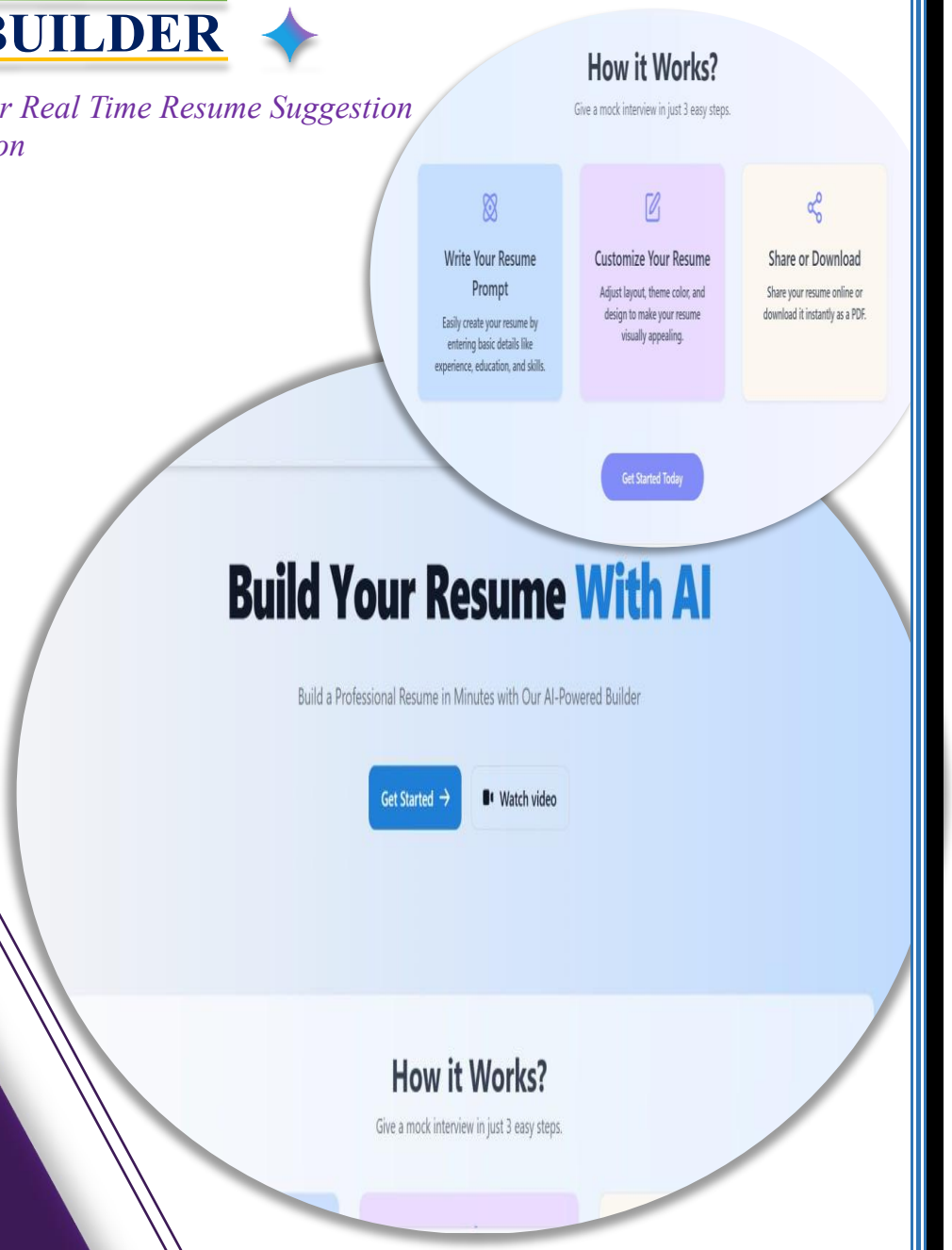
Sara Karim (ma'am) ;
Adjunct Lecturer ; Dept of CSE, IIUC

SUBMITTED BY

Kazi Namira Meyheg Sanam (C231450) &
Umme Benin Yeasmin Meem (C231452

AI RESUME BUILDER

Powered By Gemini API for Real Time Resume Suggestion & Smart Content Generation



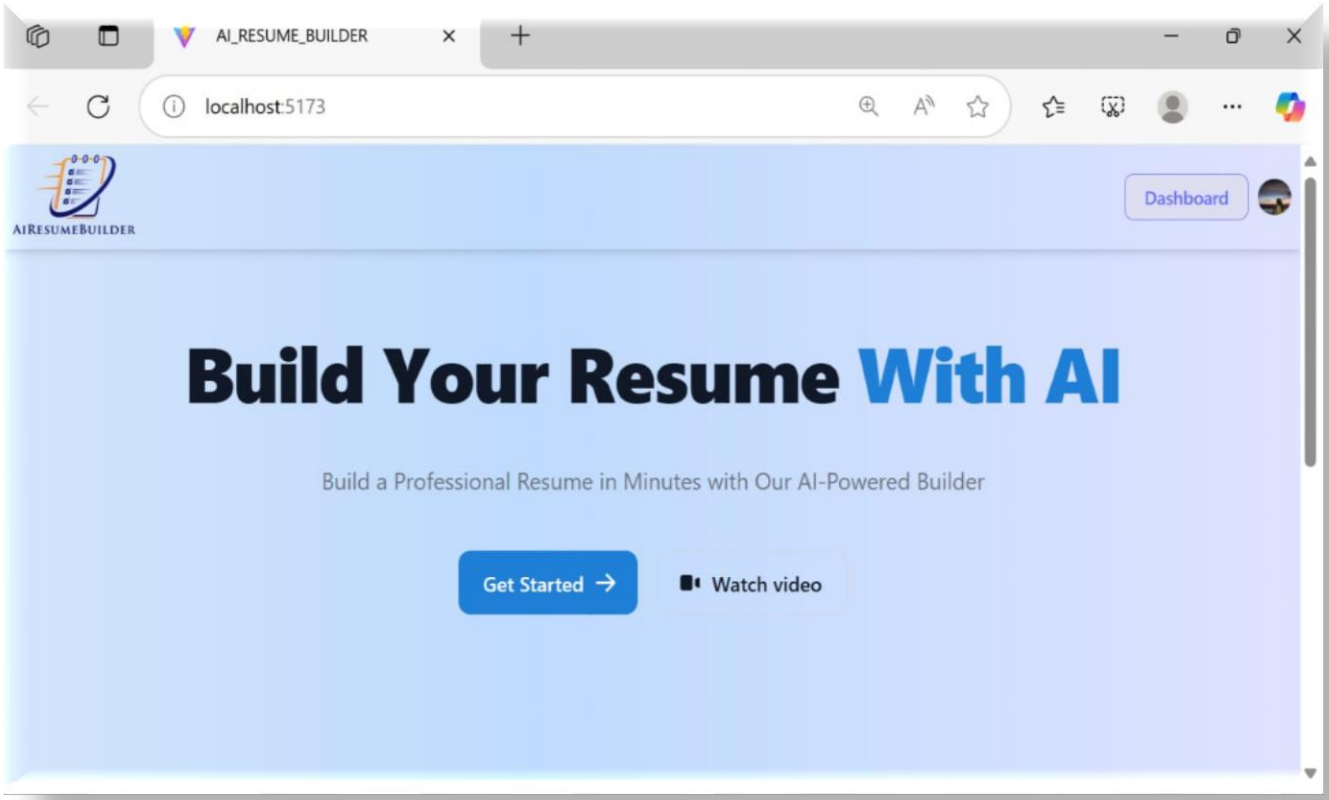
AN INTERACTIVE & ACCESSIBLE AI - POWERD RESUME BUILDER

Designed for Students, Developers To Practice, Debug & Learn Coding Efficiency

Developed Using Modern Web Technologies & Advanced AI Execution APIs

ABSTRACT

The **AI Resume Builder** is a full-stack web application developed to simplify the process of creating professional resumes using artificial intelligence. It integrates Google Gemini API to offer intelligent suggestions, provides dynamic form filling, and allows users to export resumes as downloadable PDFs. Built using modern technologies such as **React**, **Strapi**, **PostgreSQL**, and **Clerk**, the platform enhances both usability and functionality. This project demonstrates the potential of AI in solving real-world career-related challenges.



Additionally, the **AI Resume Builder** emphasizes accessibility and user experience, enabling individuals with minimal technical knowledge to craft impactful resumes effortlessly. With features like real-time content generation, skill recommendations, and secure authentication, the platform serves as a powerful tool for **students**, **job seekers**, and **professionals** alike. By automating traditionally time-consuming tasks, it not only saves effort but also ensures consistency and quality in resume content — aligning with modern hiring standards and employer expectations.

TABLE OF CONTENTS

- **ABSTRACT** — 3
- **TABLE OF CONTENTS** — 4
- **INTRODUCTION** — 5
- **WHY WE MADE THIS PROJECT (OBJECTIVE)** — 6
- **MOTIVATION BEHIND THE PROJECT** — 6
- **TOOLS & TECHNOLOGIES USED** — 6
  - Frontend
  - Backend
- **KEY FEATURES** — 7
- **FLOW-CHART** — 8
  - Frontend
  - Backend
- **SYSTEM DESIGN** — 9
- **STEP-BY-STEP DEVELOPMENT** — 10-11
  - Create React Vite App and Install Libraries
  - Implement Clerk Authentication
  - Learn & Apply TailwindCSS
  - Strapi CMS setup from Scratch
  - PostgrateSQL
  - Generate AI Form with Gemini API
  - Set up Gemini API (AI Model)
  - Generate Resume Content From AI
  - Build Dynamic Forms
  - Convert React HTML to PDF
  - Deployment (Planned)
- **PROJECT USEFULNESS** — 11
- **REAL LIFE VALUE** — 12
- **SCREENSHOT** — 12-20
- **CHALLENGES & LEARNINGS** — 20
- **FUTURE SCOPE & DEPLOYMENT PLANS** — 21
- **CNCLUSION** — 21

🕮 INTRODUCTION

In today's fast-paced job market, a well-structured, professionally written resume is crucial. It serves as a candidate's first impression and often determines whether they get shortlisted for interviews. Yet, many—especially students and fresh graduates—struggle to craft effective resumes due to lack of experience, formatting knowledge, or writing skills. The process is often time-consuming and confusing, involving content decisions, tone adjustments, and layout design.

The **AI Resume Builder** project addresses these challenges by offering a smart, AI-powered, and user-friendly platform for generating professional resumes. It leverages Google's Gemini API to provide intelligent, real-time content suggestions and formatting improvements. Users can either input their own data or allow the AI to assist in creating sections like career objectives, summaries, and skills.

Technically, this is a full-stack web application built with **React.js** and **Tailwind CSS** for the frontend, offering a modern and responsive UI. The backend is powered by **Strapi CMS**, connected to a **PostgreSQL** database hosted on **Neon.tech**, and features secure authentication via **Clerk**, supporting both social and email/password login options.

The system includes dynamic, editable form fields that let users customize their resumes section by section. With just a few clicks, users can generate AI content, preview the resume live, and export it as a downloadable **PDF**. This smart workflow not only simplifies the process but also produces high-quality, structured resumes optimized for **ATS (Applicant Tracking Systems)**.

What sets this platform apart is its combination of **artificial intelligence**, modern UI/UX, and robust backend architecture. It's not just a resume tool — it's a full-featured, intelligent assistant for job seekers. By reducing the effort involved in resume writing and ensuring professionalism, the project offers real-world value to students, professionals, and career coaches alike.

✨ WHY WE MADE THIS PROJECT ?

- To help students and job seekers build resumes more efficiently.
- To explore integration of AI into a real-world application.
- To gain hands-on experience with full-stack web development.

💡 MOTIVATION BEHIND THE PROJECT

- Many users lack guidance when creating resumes.
- Fresh graduates and students often struggle to build strong, impactful resumes.
- Traditional/manual resume writing is time-consuming and inefficient.
- Leveraging AI can provide faster, smarter, and personalized resume creation.
- This project provided an opportunity to combine my interest in **backend development** and **AI integration**.
- Inspired by the power of AI to assist in real-world tasks like writing, formatting, and content generation.
- Wanted to challenge ourselves by integrating **authentication**, **AI**, **PDF export**, and **full-stack development** into a single project.

🛠 TOOLS & TECHNOLOGIES USED

FRONTEND

- **React.js (via Vite)**: Used for building a fast, interactive, and modular user interface.
- **Tailwind CSS**: Provides utility-first styling for a responsive and modern UI.

- **Clerk (Authentication)**: Handles secure user authentication with email/password and social login options.

BACKEND

- **Strapi CMS**: A **headless CMS** used to manage resume content and **API endpoints**.
- **PostgreSQL**: A robust relational database for storing user data and resume entries.
- **Google Gemini API**: Powers **AI-driven** content generation for resume suggestions.

These modern tools and frameworks were chosen to ensure scalability, performance, and ease of development. Together, they provide a cohesive development environment capable of handling both the technical and user experience demands of an **AI-powered resume builder**.

🛠 KEY FEATURES

Dynamic Form Builder

Empowers users to create personalized resumes by dynamically adding, editing, or removing sections such as education, experience, skills, and projects — all in real-time with a smooth interface.

AI-Powered Content Generation

Integrates Google Gemini API to provide smart, context-aware suggestions for resume content like career objectives, summaries, and skill descriptions based on minimal user input.

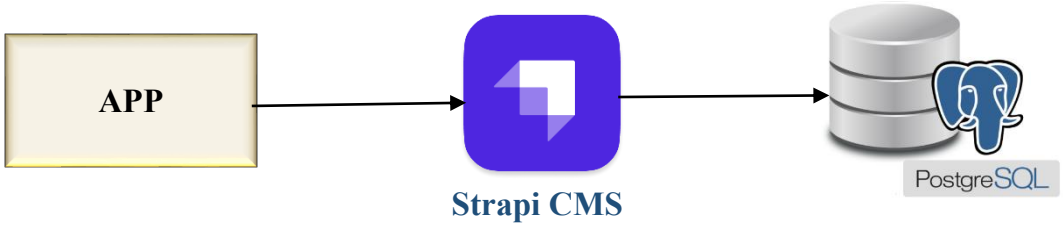
HTML-to-PDF Conversion

Transforms the completed resume (rendered in the browser) into a well-formatted, downloadable PDF file, allowing users to save or share their resumes instantly.

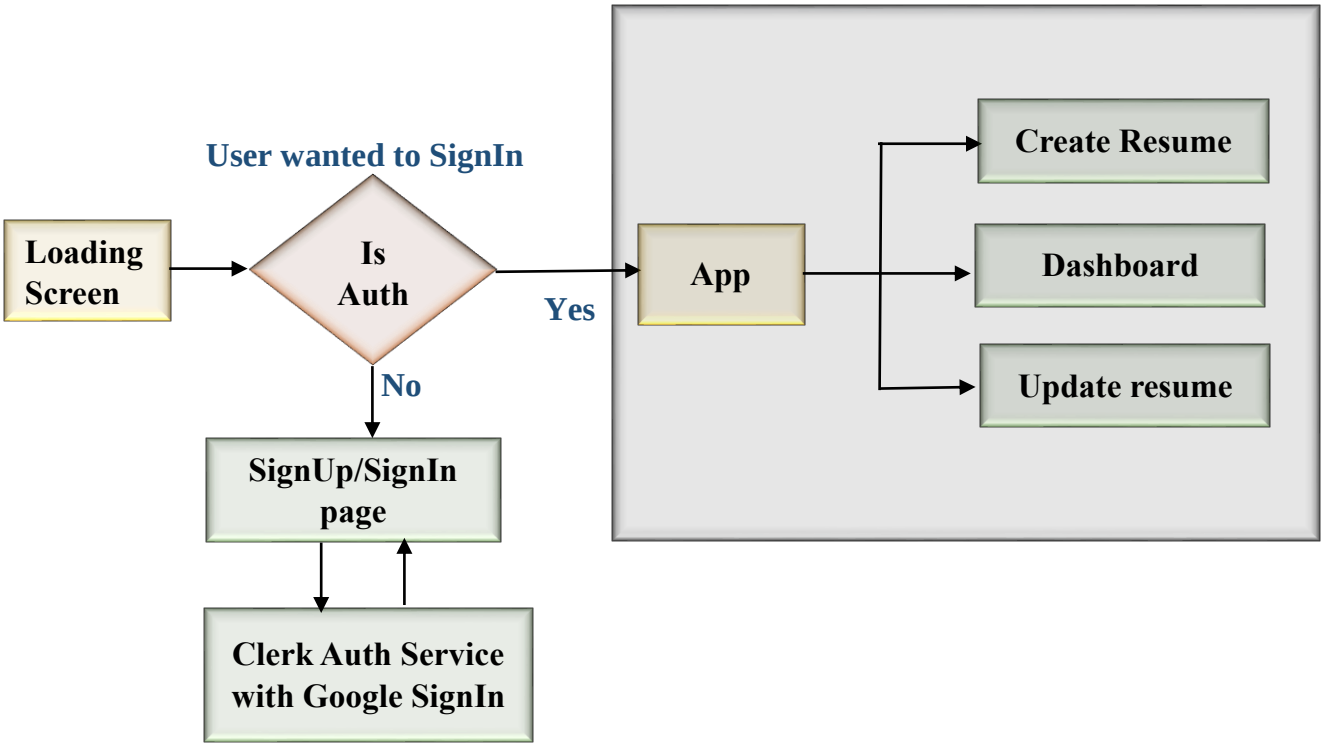
These features together offer a seamless, intelligent, and professional resume-building experience, reducing time and effort while improving output quality.

FLOW CHART

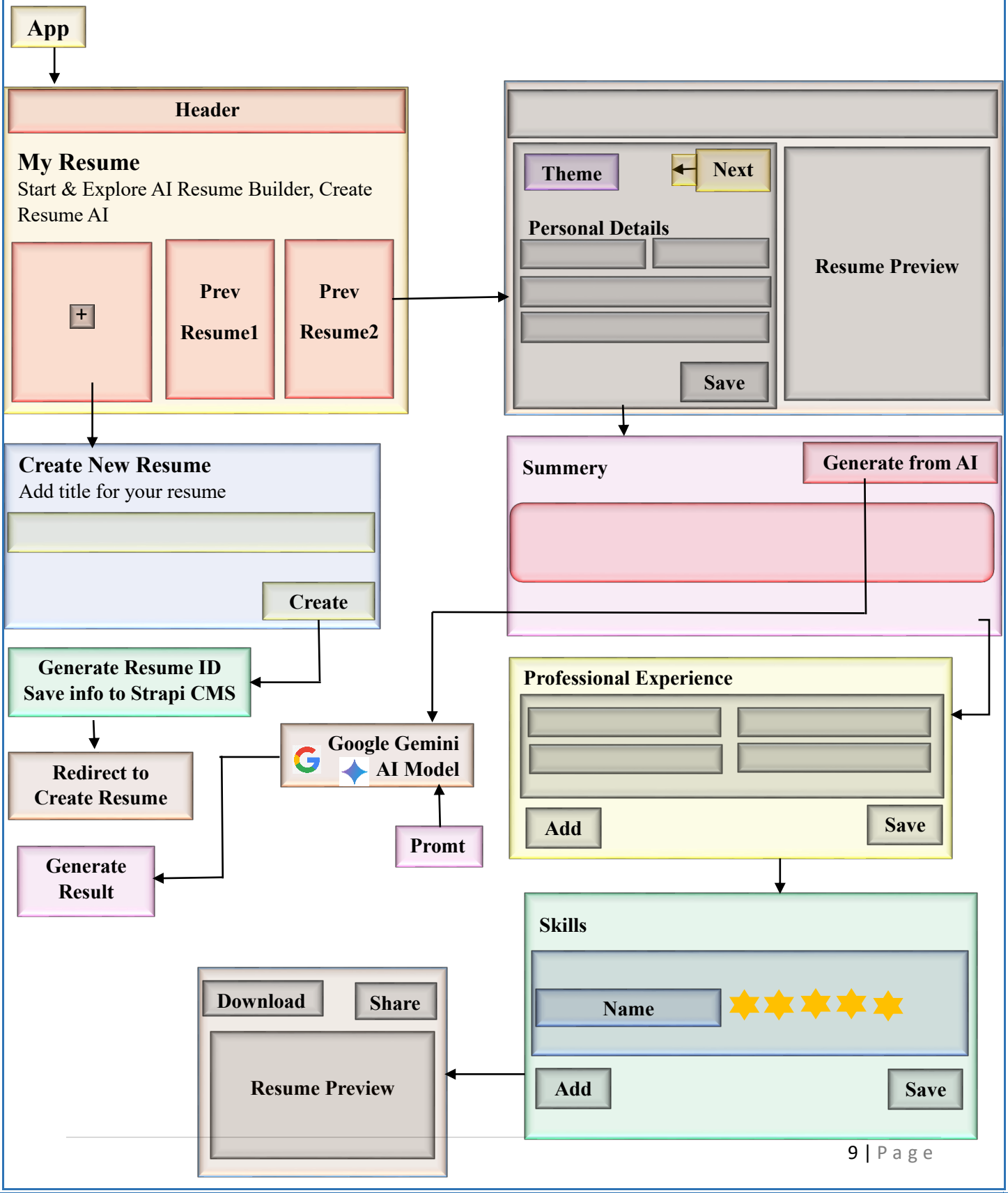
BACKEND (OVERVIEW)



FRONTEND (OVERVIEW)



SYSTEM DESIGN



STEP-BY-STEP DEVELOPMENT

1. Create React Vite App and Install Libraries

- npm create vite@latest ai-resume-builder
- **Installed:** @clerk/clerk-react, tailwindcss, axios, etc.

2. Implement Clerk Authentication

- **Configured Clerk for:**
  - Social login (Google, GitHub)
  - Email/password login
  - Protected routes & session management

3. Learn and Apply Tailwind CSS

- Built a clean and responsive UI using Tailwind.
- Custom utility classes for layout, spacing, and typography.

4. Strapi CMS Setup from Scratch

- Installed and configured **Strapi**.
- Created custom content types for resumes and user profiles.

5. PostgreSQL Setup

- Linked PostgreSQL with Strapi.
- Used environment variables to configure database connection.

6. Generate AI Form with Gemini API

- Created prompt-based AI form.
- Sent user input to Gemini API and displayed generated content.

7. Setup Gemini API (AI Modal)

- Fetched content using Gemini's generateContent method.
- Used modals to display AI suggestions.

8. Generate Resume Content from AI

- **AI fills:**
  - Career Objective
  - Experience Summary
  - Skills Description
  - Project Descriptions

9. Build Dynamic Forms

- Allowed users to add/remove sections dynamically.
- Real-time preview as users type.

10. Convert React to PDF

- Used REACT to convert user's resume into downloadable PDF format.

11. Deployment (planned)

- Frontend Will be deployed on **Vercel** for fast, global access.
- Backend **Strapi CMS** with **PostgreSQL** to be hosted on **Render** or run locally during development.

✓ PROJECT USEFULNESS

- **Time-Saving**: Users no longer need to think hard about resume content.
- **Professional Output**: Automatically generates polished resume content using Gemini AI.
- **Beginner Friendly**: Simple UI and guided steps make it usable by anyone.
- **Real-Time Editing**: Dynamic forms allow users to update and view changes instantly.
- **Exportable Resume**: Users can download their resume as a PDF.

REAL LIFE VALUE

- **For Students/Job Seekers**: Create job-ready resumes with AI help.
- **For Career Coaches**: Assist multiple clients efficiently.
- **For Freelancers**: Quickly prepare portfolios and resumes on demand.
- **Time Efficiency**: Greatly reduces the time needed to make a professional resume.

SCREENSHOT

WEBSITE (PREVIEW)

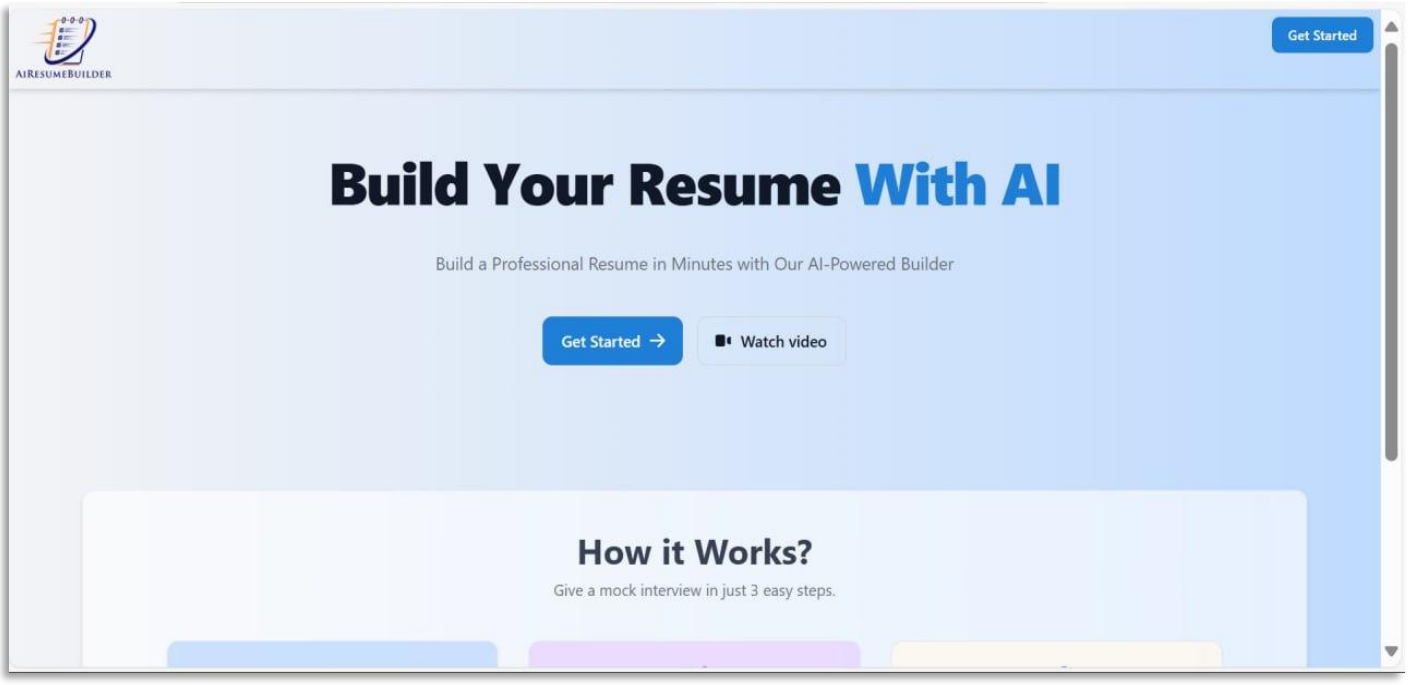


Fig : This is our Home Page before user LogIn

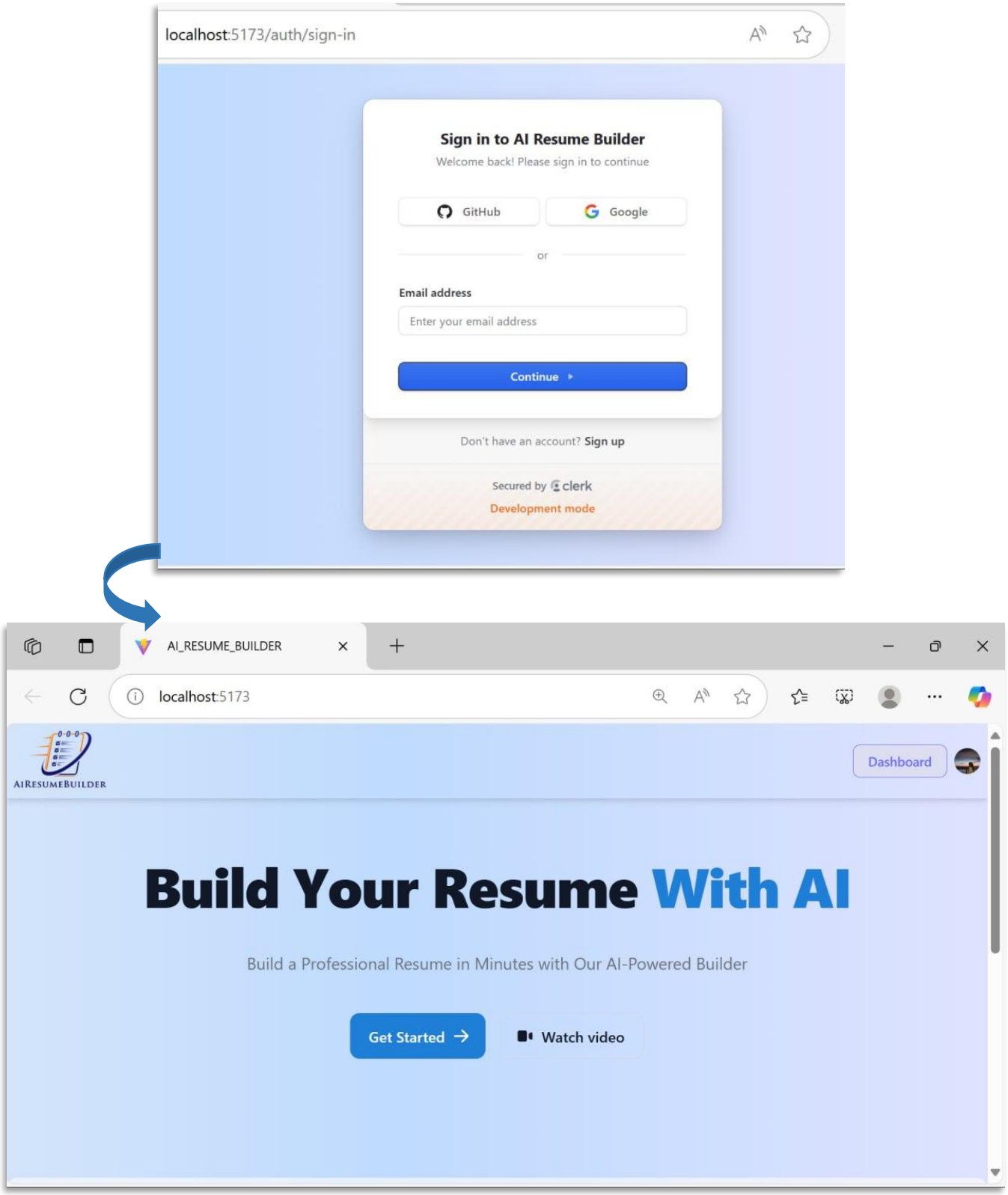


Fig : user will log in through Clerk authentication and then go to the dashboard

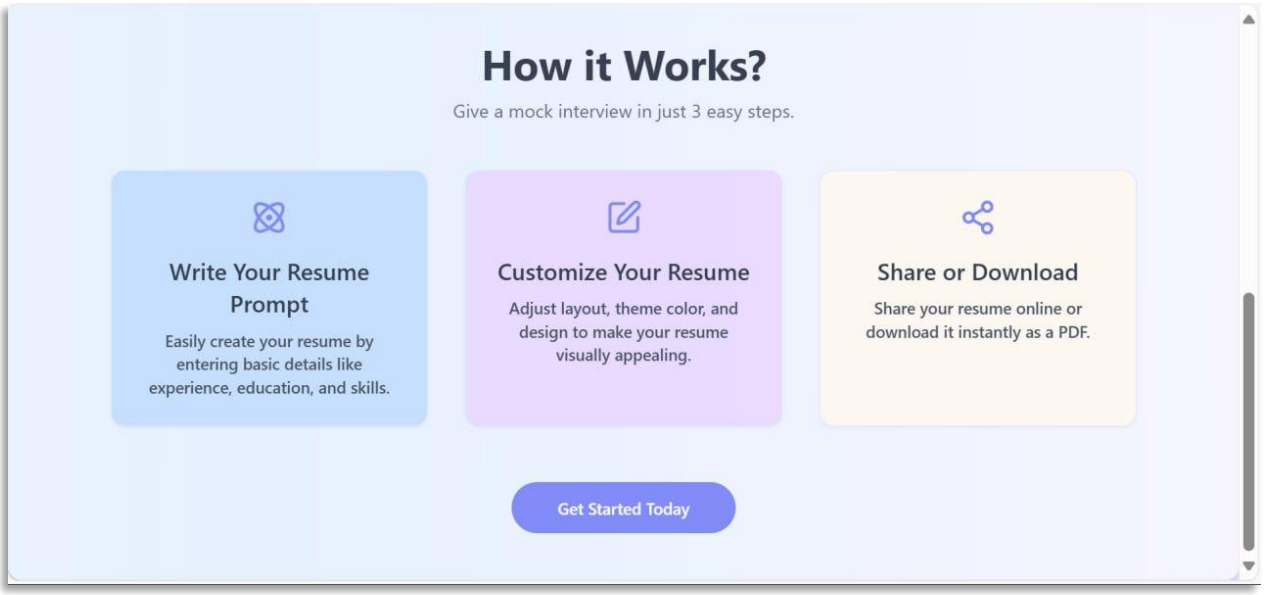


Fig : This is the Footers of our website

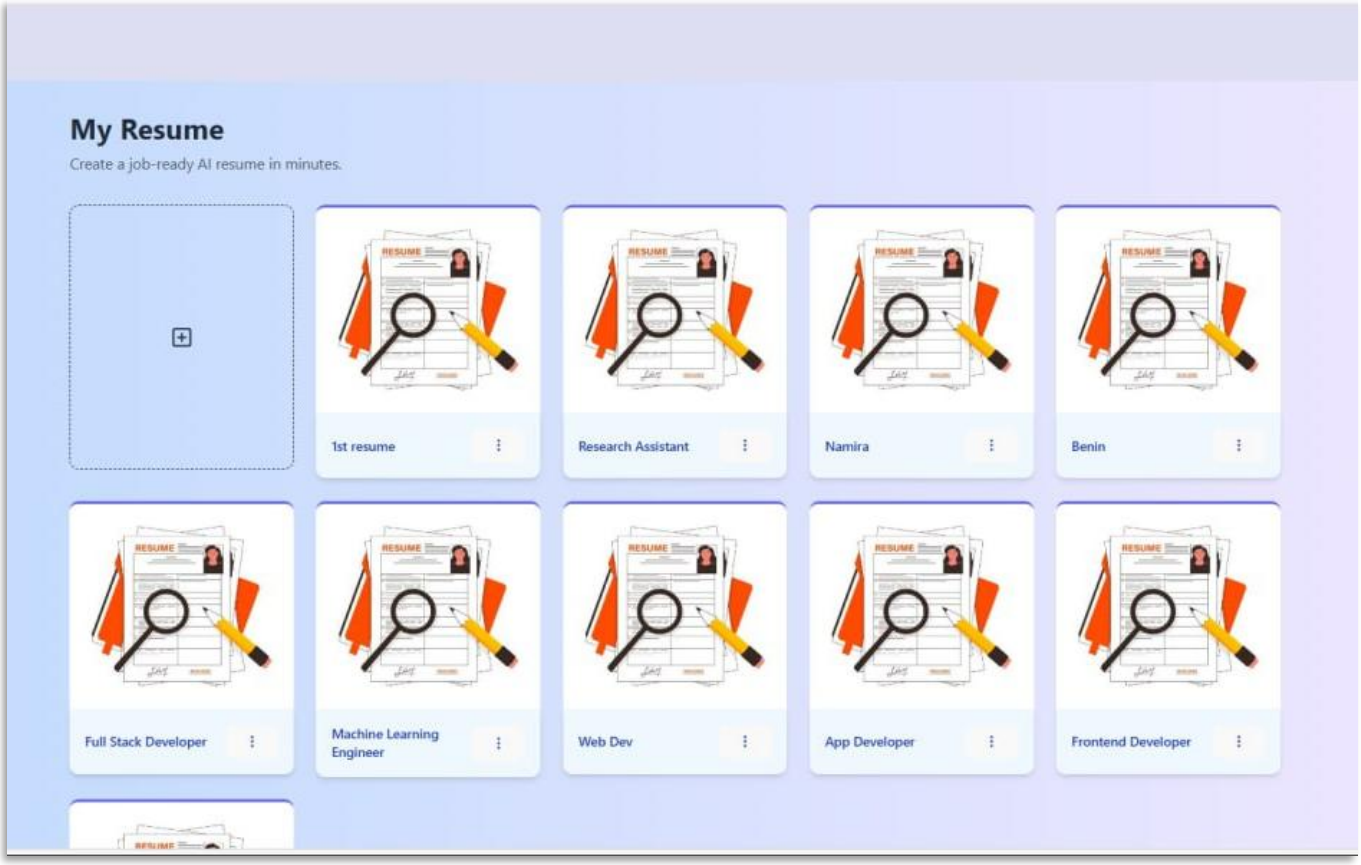


Fig : This is the resume section, where the user will create resumes

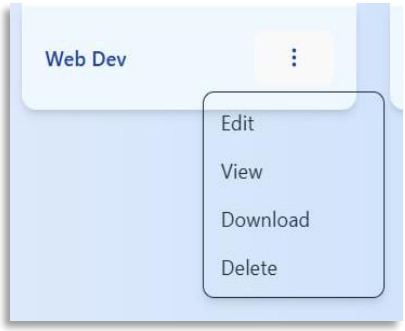
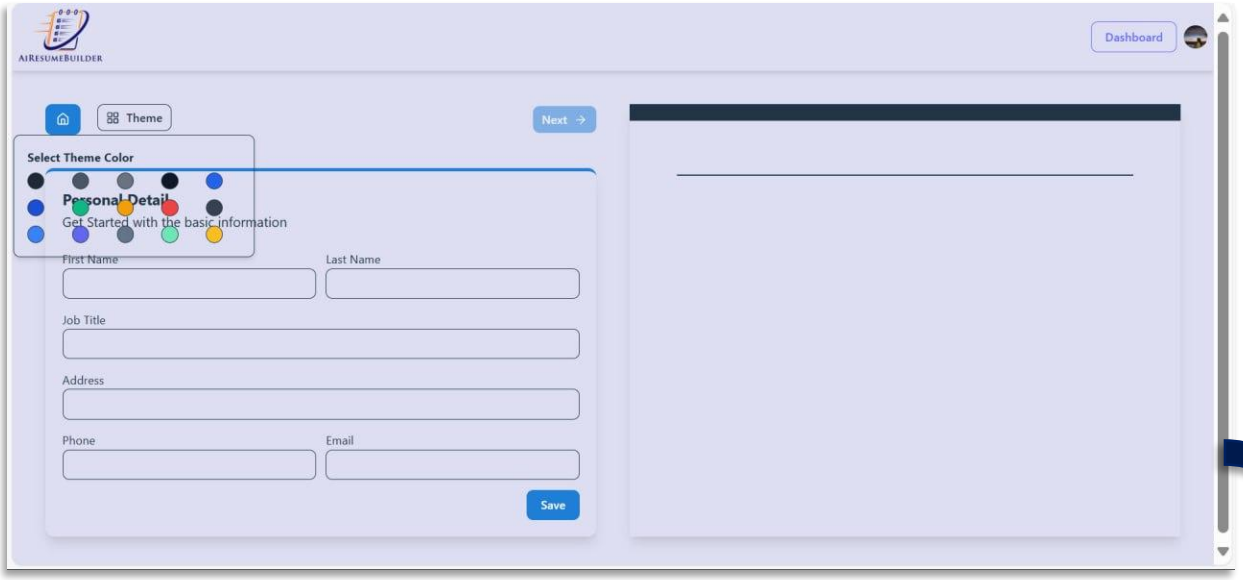


Fig : And use various features such as view, edit, delete, and download.



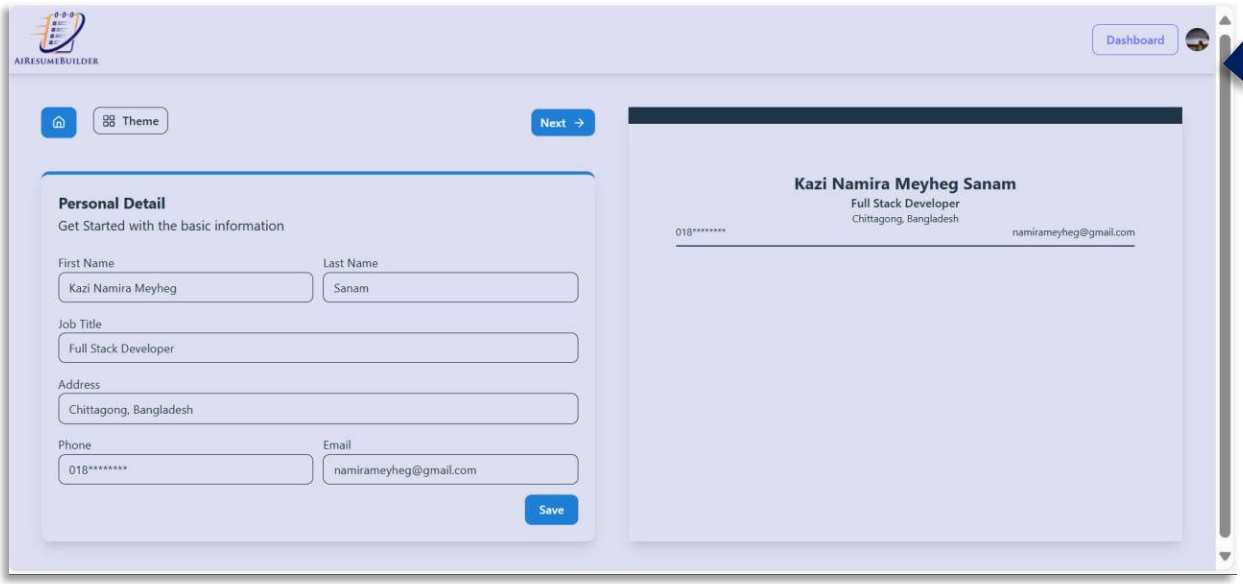


Fig : This is the personal details section where the user will enter their basic personal information

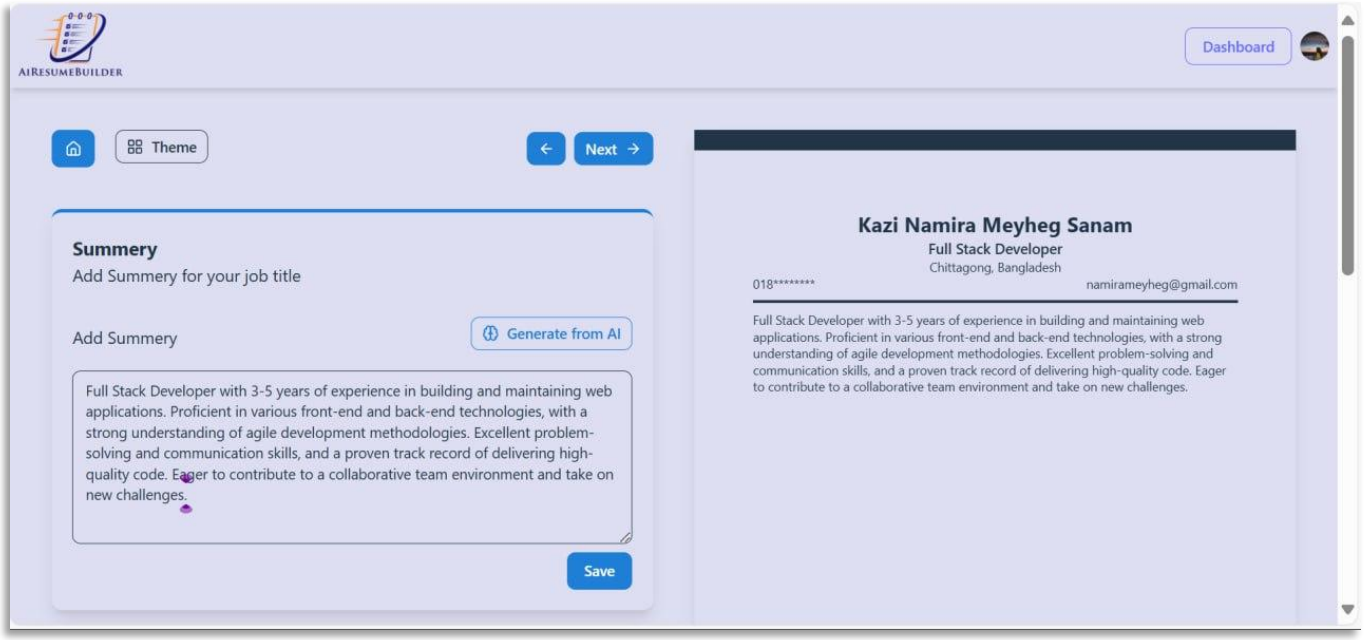


Fig : This is the summary section whose information will be used by the AI And then, one by one, I'll need to fill in each field.

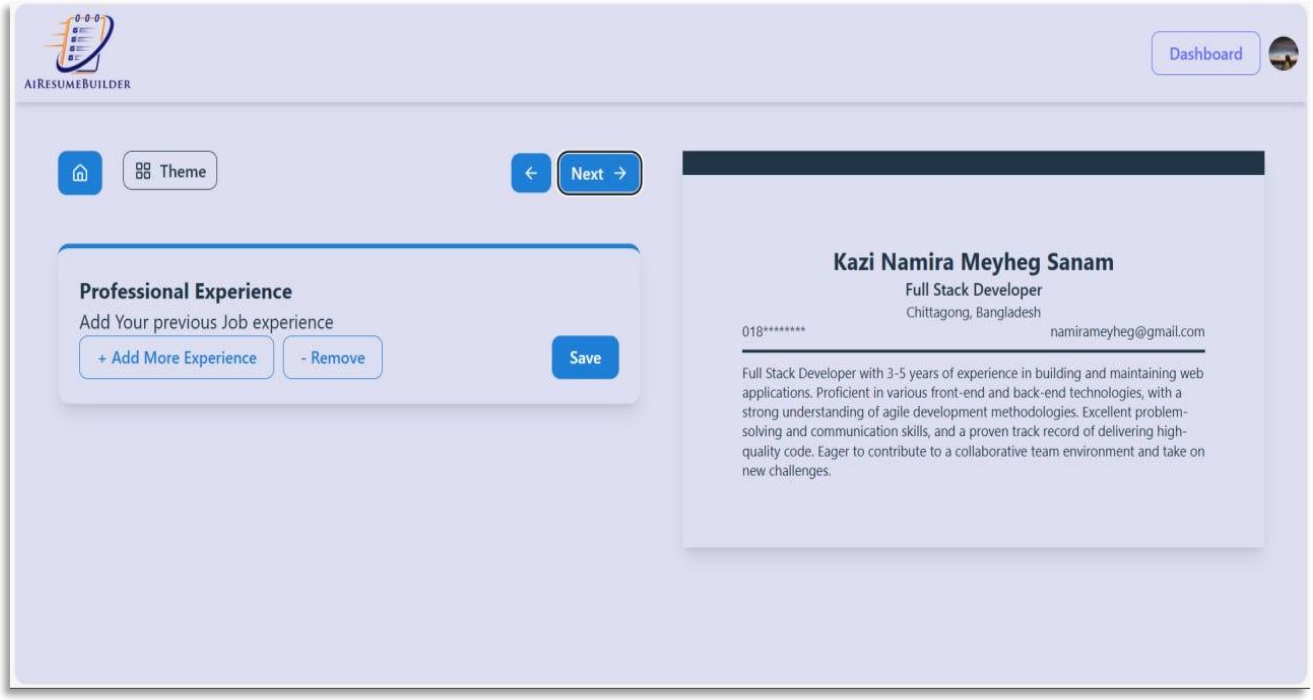


Fig : This is Professional Experience Section where user will add their experience

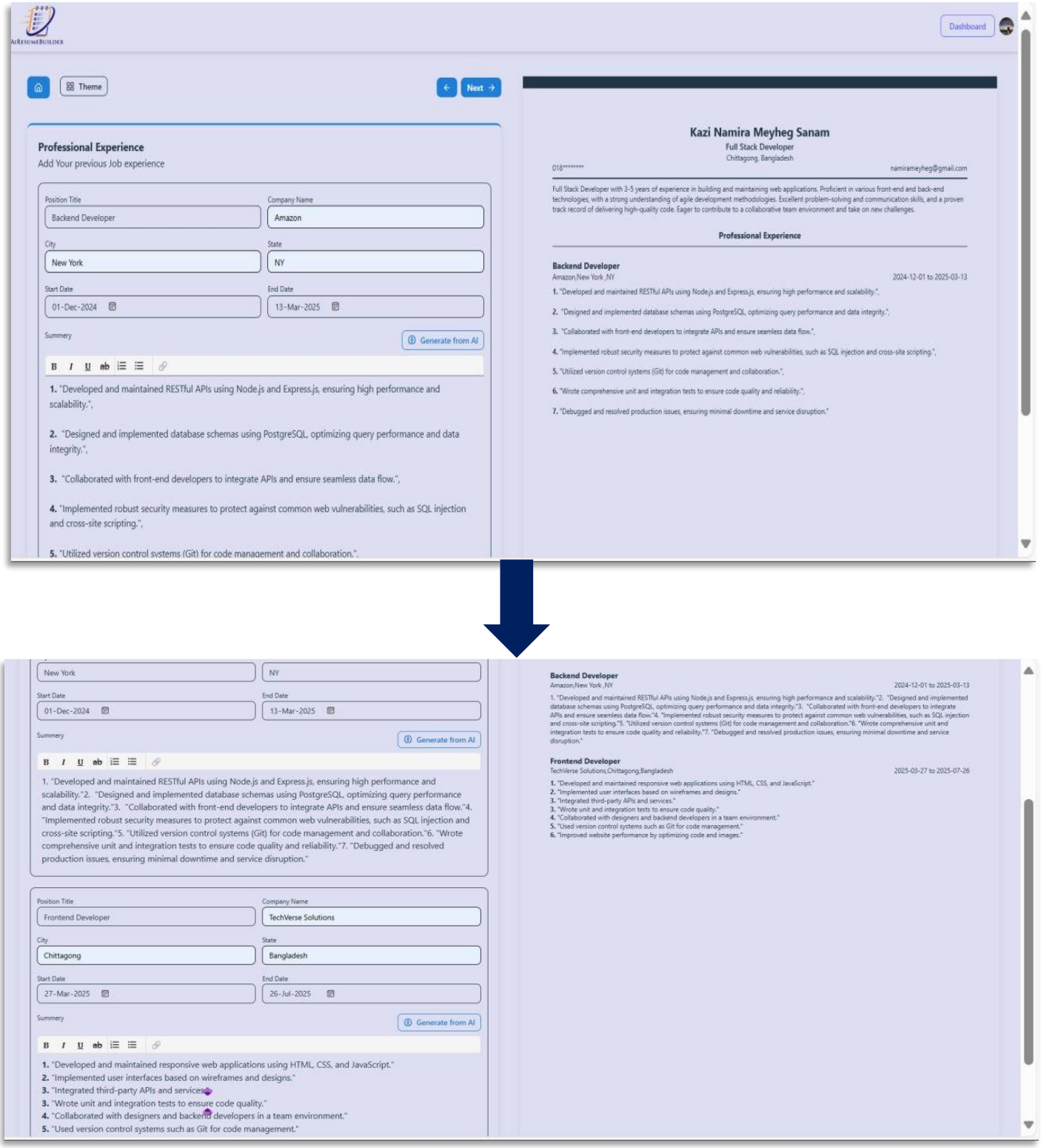


Fig : User will fill up their personal details

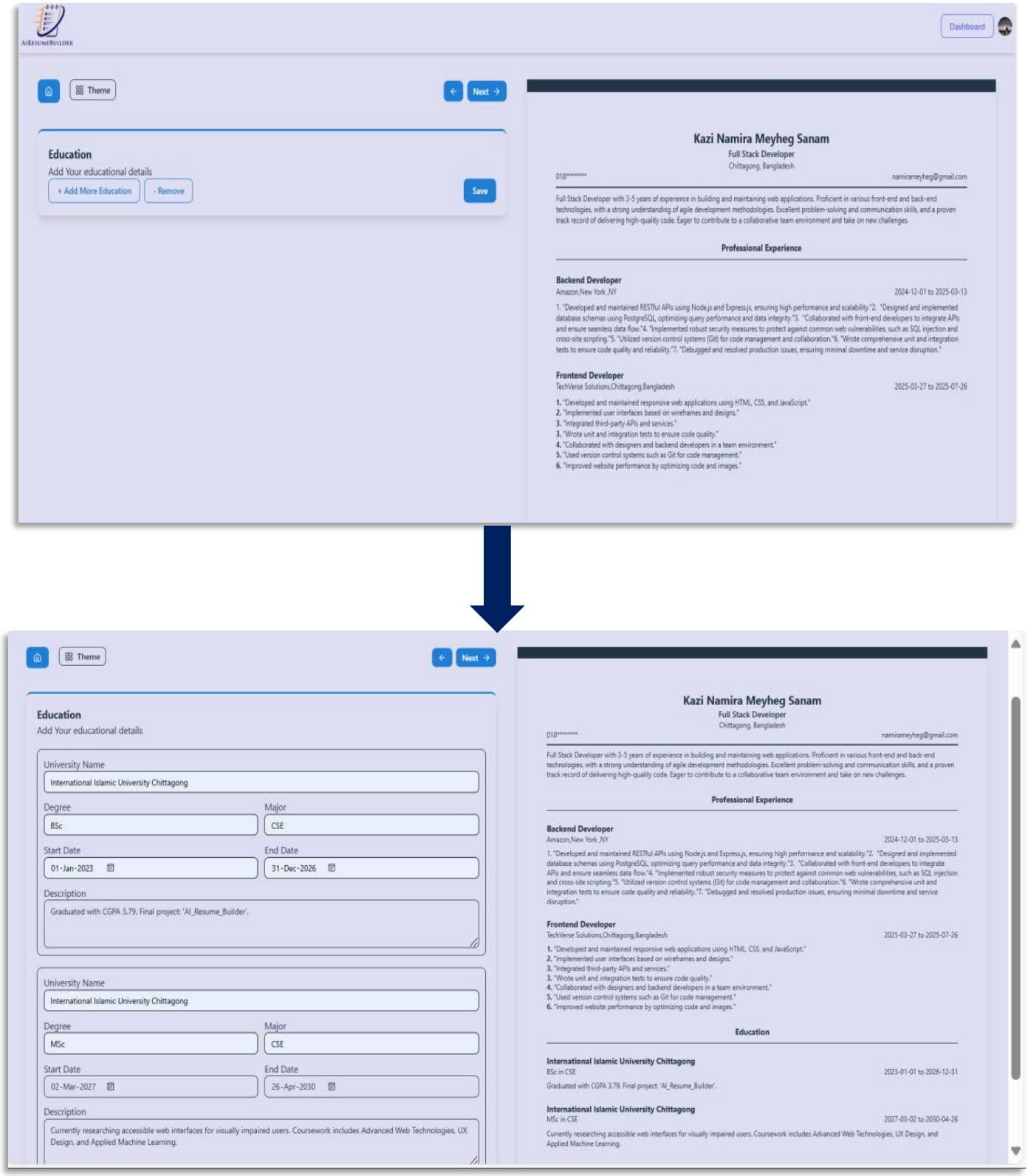


Fig : This is Education Section where user will add their Educational Status/Background

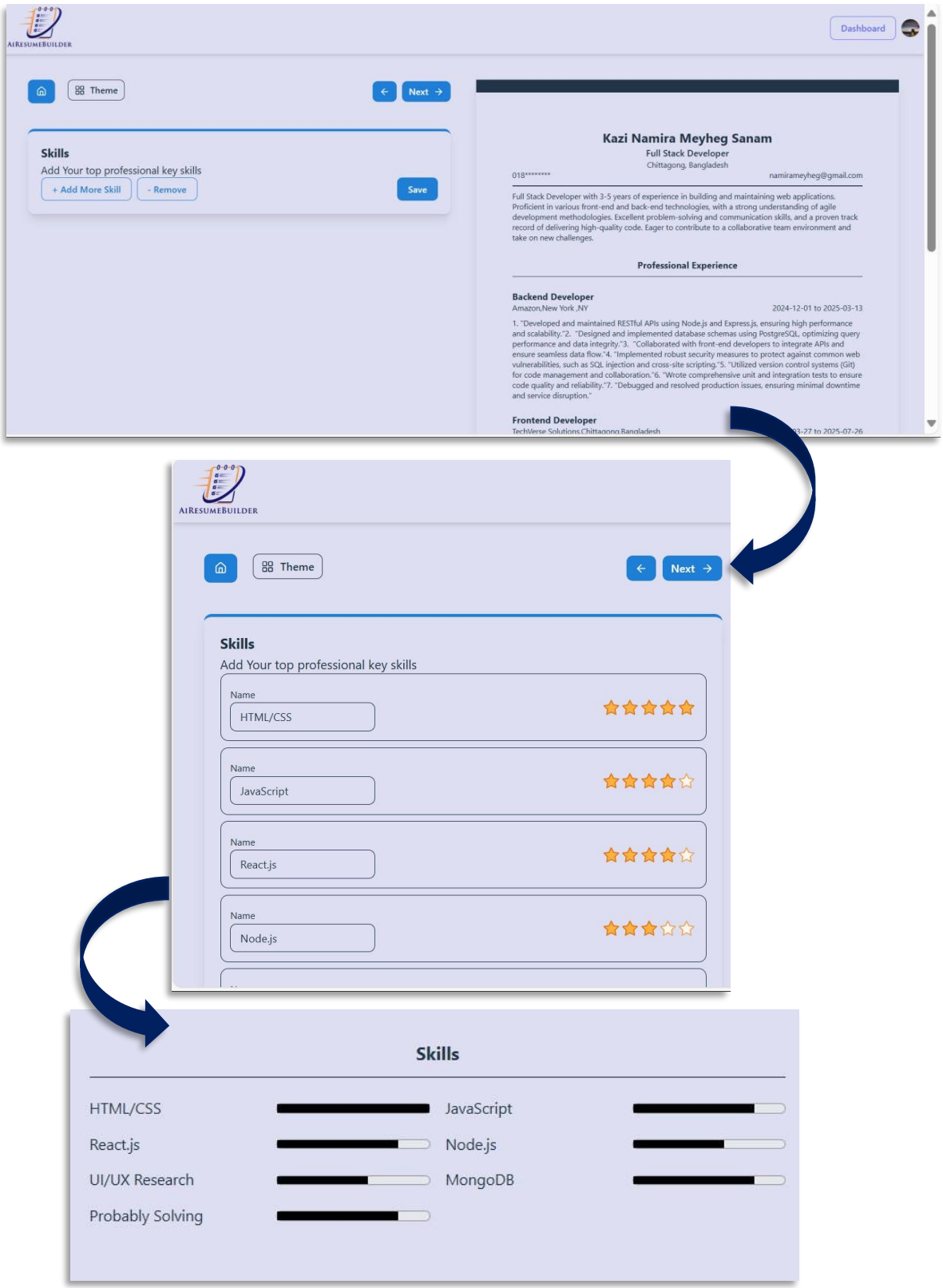


Fig : This is Skills Section where user will add their Skills

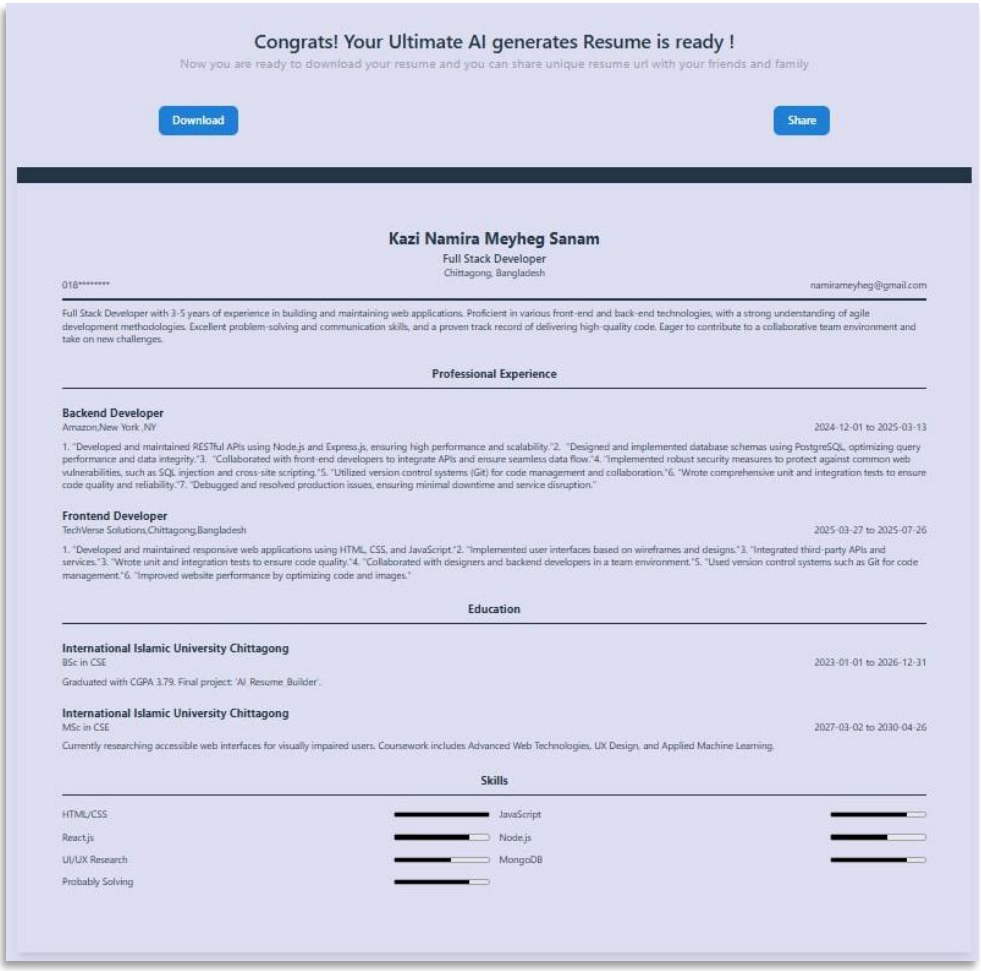


Fig : This is the final resume, which can later be downloaded

⚙ CHALLENGES & LEARNINGS

- Integrated **Google Gemini API** with **Strapi backend for AI** suggestions.
- Built dynamic, real-time editable forms in **React**.
- **Implemented PDF** export and handled layout issues.
- Set up secure social & email login with **Clerk**.
- Focused on creating a clean, **responsive UI** with **Tailwind CSS**.

🚀 FUTURE SCOPE & DEPLOYMENT PLANS

- Deploy frontend on **Vercel** and backend on **Render**.
- Add more AI-powered suggestions and keyword optimization.
- Offer multiple templates and customization.
- Build user dashboards and real-time collaboration features.
- Add usage analytics and insights.

🔚 CONCLUSION

The **AI Resume Builder** successfully combines **modern** **web technologies** and **artificial intelligence** to address a real-world challenge faced by job seekers and students: crafting a professional and compelling resume. By integrating features like **dynamic form building**, **AI-powered content suggestions**, and **instant PDF export**, it transforms a traditionally time-consuming process into an efficient, user-friendly experience.

Beyond showcasing technical skills in **full-stack development**, **API integration**, and **responsive design**, this project highlights how thoughtful use of AI and intuitive design can truly empower users — especially those with limited technical or writing expertise. The platform makes resume creation faster, smarter, and more personalized, ultimately helping users present themselves confidently in the job market.

In essence, the **AI Resume Builder** stands as a practical and meaningful example of how technology can solve everyday problems and add real value to people's professional lives.

THE END