**Week 4 Group Project: AI Resume Builder**

**Group Members:**

1. **Mthunzi Malebadi (Group Leader)**
2. **Sinesipho Sibulo**
3. **Ondela Citywayo**
4. **Vhuvhwavho Mawela Maselesa**
5. **Segai Bryton Mampshika**

**AI Resume Builder Presentation Documentation**

### **Introduction & Problem Statement**

**Title: AI Resume Builder – Smart, Simple & ATS-Friendly**

**1.1 Background & Motivation**

In today’s digital job market, job seekers face two main challenges:

* Creating a well-structured, professional resume
* Ensuring that resume passes through Applicant Tracking Systems (ATS)

Traditional resume builders are often generic, while writing a resume from scratch is time-consuming and stressful, especially for individuals unfamiliar with formatting or keyword optimization.

**1.2 Objective**

This AI Resume Builder was developed to:

* Generate customized resumes using simple user inputs
* Ensure ATS-compatibility with automated keyword checks
* Offer downloadable PDF, HTML & .DOC output for job applications

**1.3 Target Users**

* Students and graduates
* Career switchers
* Freelancers and early-career professionals

**Project Link:** [View the Resume Builder on CodePen](https://codepen.io/Ondela-Citywayo/pen/wBaQRzM)

### **How It Works – Features & Technologies**

**2.1 Key Features**

* **User Input Section:** Users enter basic information such as name, skills, education, and experience.
* **AI Resume Generation:** Auto-generate structured resume content based on user input using template logic.
* **ATS Compliance Checker:** Validates if the resume contains key industry-related terms.
* **PDF Download:** Converts the generated resume to a downloadable format using jsPDF.

**2.2 Tools & Technologies**

|  |  |
| --- | --- |
| **Tool** | **Purpose** |
| HTML/CSS | Layout, styling, and user interface |
| JavaScript | Logic for resume generation |
| jsPDF | Convert resume to downloadable PDF |
| AI Logic | Simulated intelligence for keyword check |

**2.3 User Flow**

1. Landing Page
2. User Information Input (User enters personal, educational, and professional information)
3. AI-suggested Information (Browser Templates)
4. ATS Checker (ATS keywords are checked and highlighted)
5. PDF Template Options
6. Download/Export (User downloads the resume as a PDF file/HTML file/.Doc file)

### **Step-by-Step Build Process**

**3.1 Project Setup (HTML Structure)**

* Built a simple HTML layout with a form to collect user input.
* Added a container to preview the resume and buttons for generation and downloading.

**3.2 Styling the UI (CSS)**

* Applied clean, responsive design using basic CSS.
* Focused on usability and readability across devices.

**3.3 Resume Generation Logic (JavaScript)**

* Created generate Resume() function to dynamically generate content.
* Captured form values and displayed a formatted resume preview.

**3.4 ATS Keyword Checking**

* Defined a keyword list and checked for presence in resume text.
* Alerted users if key skills were missing.

**3.5 PDF Generation using jsPDF**

* Used the jsPDF library to convert resume content to PDF.
* Implemented a downloadPDF() function.

**3.6 Testing and Refinement**

* Validated user inputs, tested responsiveness, improved formatting.

### **Technical Requirements & Specifications**

**4.1 Functional Requirements**

|  |  |  |
| --- | --- | --- |
| **ID** | **Requirement** | **Description** |
| FR1 | User Input Form | Collects career info (skills, experience, education). |
| FR2 | AI Content Generation | Auto-fills resume sections based on user inputs. |
| FR3 | Keyword Optimization | Integrates industry-specific keywords for ATS. |
| FR4 | Template Customization | 3+ visual templates with editable layouts. |
| FR5 | Export Options | Saves resumes as PDF, DOCX, or HTML. |
| FR6 | ATS Compatibility Check | Analyzes resume against ATS standards. |
| FR7 | Job Description Matching | Compares resume with job postings for fit. |
| FR8 | Feedback Loop | Improves suggestions based on user edits. |

**4.2 Non-Functional Requirements**

|  |  |  |
| --- | --- | --- |
| **ID** | **Requirement** | **Description** |
| NFR1 | Performance | Loads previews in <2 seconds. |
| NFR2 | Privacy | Uses local storage (no external databases). |
| NFR3 | Cross-Browser Support | Works on Chrome, Firefox, Edge. |
| NFR4 | Responsive Design | Mobile and desktop compatible. |
| NFR5 | Accessibility | Follows WCAG 2.1 guidelines. |

**4.3 Technical Stack**

* **Frontend:** HTML5, CSS3, JavaScript
* **Libraries:** jsPDF/html2pdf for PDF export
* **Storage:** Browser localStorage
* **Hosting:** CodePen or static sites like GitHub Pages/Netlify
* **Testing:** Manual + cross-browser testing (BrowserStack)

### **Impact, Future Scope & Evaluation**

**5.1 Value Delivered**

* **Efficiency:** Generates resumes in under 2 minutes
* **Accessibility:** Free and easy-to-use for anyone online
* **ATS Readiness:** Gives users higher confidence when applying to modern employers

**5.2 SWOT Analysis**

|  |  |
| --- | --- |
| **Category** | **Analysis** |
| Strengths | Fast online resume generation, ATS optimization |
| Weaknesses | Limited to browser storage, simpler AI compared to paid tools |
| Opportunities | NLP-powered job matching, LinkedIn integration |
| Threats | Competition from Zety, Novoresume, Resume.io |

**5.3 Evaluation Criteria**

|  |  |
| --- | --- |
| **Criteria** | **Weight (%)** |
| User Experience (Intuitiveness) | 25% |
| Output Quality (ATS/Design) | 25% |
| Technical Implementation | 20% |
| Documentation Quality | 15% |
| Innovation (AI Features) | 15% |

**5.4 Conclusion**

This project bridges the gap between job seekers and digital hiring systems. By combining AI logic with clean design and PDF tools, it empowers users to take control of their career documents — simply and smartly.

**Link to Project:** <https://codepen.io/Ondela-Citywayo/pen/wBaQRzM>