**Technologies to Use**

**Frontend Development (User Interface)**

1. **React.js / Angular / Vue.js**: For creating a responsive and dynamic user interface.
2. **HTML, CSS, JavaScript**: Core web technologies for basic structure, styling, and interactivity.
3. **Bootstrap / Tailwind CSS**: For designing a user-friendly and modern UI quickly.
4. **React Query / Axios**: For managing API requests to the backend.

**Backend Development (Data Handling and Logic)**

1. **Node.js with Express**: Lightweight and efficient for building APIs.
2. **Python with Django / Flask**: Alternative frameworks for server-side development.
3. **Ruby on Rails**: For rapid application development.
4. **Spring Boot (Java)**: If you prefer Java-based solutions.

**Database (Data Storage)**

1. **MongoDB**: For storing user data in a NoSQL format, which is flexible and scalable.
2. **PostgreSQL / MySQL**: For structured and relational data storage.
3. **Firebase Realtime Database**: For quick prototyping with real-time data.

**Resume Generation**

1. **PDF Generation Libraries**:
   * **jsPDF**: For creating and downloading PDF files directly from the frontend.
   * **pdf-lib**: For manipulating PDF content dynamically.
   * **Puppeteer**: For converting HTML templates to PDFs.
2. **HTML-to-PDF Converters**:
   * **wkhtmltopdf** (backend-based solution).
   * **HTML Canvas Libraries**: For advanced custom resume formatting.

**Cloud Storage (Optional)**

1. **AWS S3 / Google Cloud Storage**: To store and retrieve user data or generated resumes.
2. **Firebase Storage**: For simple integration with a Firebase-based backend.

**User Authentication**

1. **OAuth2 / OpenID Connect**: For social login (e.g., Google, Facebook).
2. **Firebase Authentication**: Easy-to-use and integrates well with Firebase services.
3. **JWT (JSON Web Tokens)**: For secure user authentication in a custom backend.

**Hosting Platforms**

1. **Frontend**:
   * **Netlify / Vercel**: For deploying the frontend.
   * **GitHub Pages**: For hosting static sites.
2. **Backend**:
   * **Heroku / AWS EC2 / Render**: For deploying backend services.
   * **Firebase Functions**: For serverless backend development.

**Development Workflow**

1. **Frontend**: Build forms for user input (e.g., Education, Skills, Projects) and provide a preview of the resume in different templates.
2. **Backend**: Create APIs to save and retrieve user data. Optionally, implement user login and profile management.
3. **Resume Generation**: Use HTML templates styled with CSS for the resume. Convert the templates to PDF using a library like jsPDF.
4. **File Download**: Allow users to download the resume by triggering the jsPDF.save() method or backend-generated PDF response.
5. **Template Selector**: Implement a template gallery in the UI with previews for users to choose from.
6. **Testing**: Test across devices and browsers for compatibility and ensure file downloads work as expected.