

Introduction to Tech Stack

The Resume Builder platform aims to provide a seamless user experience while ensuring robust backend operations, scalability, and security. The chosen architecture is a MERN (MongoDB, Express.js, React.js, Node.js) stack, which offers full-stack JavaScript development capabilities and ensures efficient, scalable solutions.

1. Frontend Details :

- **Framework/Library:** React.js
- **Reasons for Choice:**
 - Highly responsive and dynamic user interface.
 - Component-based architecture for reusability and modular development.
 - Rich ecosystem with libraries for form validation and state management (e.g., Redux or Context API).
 - Excellent support for real-time features like live error-checking and tooltips.

2. Backend Details :

- **Programming Language:** JavaScript (Node.js)
- **Framework:** Express.js
- **Reasons for Choice:**
 - Non-blocking I/O for high-performance APIs.
 - Flexibility in handling real-time guidance and dynamic form submissions.
 - Easy integration with frontend frameworks like React.
 - Extensive library ecosystem for functionalities like authentication and data validation.

3. Database :

- **Type:** NoSQL
- **Database Solution:** MongoDB
- **Reasons for Choice:**
 - Schema flexibility to accommodate diverse resume fields.
 - High scalability for handling a large volume of student data.
 - Native JSON support for seamless integration with Node.js.

4. Hosting and Deployment :

- **Cloud Service:** AWS (Amazon Web Services)
- **CI/CD Tools:** GitHub Actions, Docker, Jenkins
- **Reasons for Choice:**
 - **AWS:**
 - Elastic Beanstalk for automated deployment and scaling.
 - S3 for static file storage (e.g., templates, evidence uploads).
 - RDS or DynamoDB for database management.

- **GitHub Actions:**
 - Streamlined workflow for continuous integration and delivery.
 - Direct integration with the project repository.
- **Docker:**
 - Containerization ensures consistency across development, testing, and production environments.
- **Jenkins:**
 - Robust automation capabilities for build and deployment pipelines.

Why This Tech Stack?

1. **Scalability:**
 - The MERN stack provides high scalability, essential for accommodating the growing user base of students.
2. **Performance:**
 - Non-blocking architecture and schema-less database design ensure fast response times and efficient data handling.
3. **Ease of Development:**
 - React's component-based model and Node.js's extensive ecosystem accelerate development time while maintaining high code quality.
4. **Cloud Benefits:**
 - AWS's comprehensive tools offer cost-efficient and reliable hosting, with global reach and integrated monitoring tools.
5. **Security and Verification:**
 - Integration with cloud-native tools ensures secure data handling, while OCR and API integrations facilitate robust verification mechanisms.