AI-Powered Resume Screener Using AWS Textract and Serverless Architecture

**Project Summary**

The Resume Screener Project is a serverless, cloud-native solution designed to automatically extract, analyze, and evaluate resumes submitted to an Amazon S3 bucket. Using AWS services such as Textract, Lambda, DynamoDB, and SNS, the system screens resume for key job-related skills, stores evaluation data, and notifies relevant stakeholders, all without manual intervention or third-party platforms.

**Project Inspiration & Motivation**

This project was born from a real-world challenge faced by HR departments and recruiters globally:

***"How can we quickly and objectively assess thousands of resumes without bias or burnout and without costly ATS systems?"***

Manual resume review processes are:

* Time-consuming
* Inconsistent
* Biased
* Costly

Traditional Applicant Tracking Systems (ATS) are expensive and often out of reach for startups, NGOs, and small to medium-sized enterprises (SMEs).

Inspired by the scalability and affordability of cloud-native serverless technologies, this project seeks to automate and democratize resume screening—making recruitment faster, fairer, and financially accessible.

**Problem Statement**

Recruiters and hiring teams often spend hours manually reviewing hundreds of resumes for specific skills or certifications. This causes:

* Delayed response times for applicants
* Missed talent due to human error
* Burnout among HR personnel
* Inefficiencies in high-volume recruitment scenarios

Additionally, many small organizations cannot afford expensive subscription-based ATS platforms. There’s a need for a cost-effective, scalable, and intelligent screening tool.

Technical Architecture

Core AWS Services Used

|  |  |
| --- | --- |
| **AWS Service** | **Role** |
| S3 | Stores incoming resumes in PDF format |
| Textract | Extracts raw text and structured content from resumes |
| Lambda | Serverless function that orchestrates the process |
| DynamoDB | Stores resume metadata, scores, and keyword matches |
| SNS (Simple Notification Service) | Sends alerts with screening results |

**How It Works (Workflow)**

1. **Resume Upload:** A resume is uploaded to an S3 bucket (manually or via a form).
2. **Triggering Lambda:** The S3 upload triggers a Lambda function.
3. **Text Extraction (Textract):** AWS Textract is invoked to analyze and extract text from the resume.
4. **Keyword Matching:** Extracted text is compared with a configurable list of job-specific keywords (e.g., AWS, Python, Certified, Security).
5. **Scoring:** A match score is calculated and stored.
6. **Data Storage:** Resume ID, matched keywords, score, and metadata are saved in DynamoDB.
7. **Notification:** A formatted message is sent via SNS to notify recruiters or HR teams.

**Sample Output (Lambda Return JSON)**

*json*

*{*  
 *"processed\_file": "uploads/devops-engineer2.pdf",*  
 *"score": 6,*  
 *"keywords\_found": ["AWS", "Python", "Cloud", "Certified", "Architecture", "Security"]*  
*}*

**Achievements & Results**

* Reduced resume screening time from minutes to seconds
* Standardized and objective skill-based scoring
* Structured metadata storage for reporting and filtering
* Real-time notifications for quick review
* Saved cost for small businesses compared to commercial ATS platforms

**Social, Economic & Industry Impact**

This solution provides a low-cost and high-impact alternative to traditional applicant tracking systems, especially for:

* Startups looking to scale fast without expensive recruitment platforms
* Small and Medium Businesses (SMBs) aiming to reduce HR costs
* NGOs and nonprofits operating on limited budgets
* Developing regions where digital HR infrastructure is lacking

By democratizing access to intelligent recruitment tools, this project contributes to:

* More inclusive hiring practices
* Job accessibility for overlooked candidates
* Efficiency and fairness in global recruitment processes

**Opportunities for Improvement & Scaling**

**Room for Improvement**

* Add semantic matching (e.g., “serverless” matches “Lambda”)
* Integrate AI/ML models for contextual scoring beyond keywords
* Support custom keyword sets for different job roles
* Build a web interface for recruiters to search/filter results

**Scalability Possibilities**

* Enable multi-tenant support for HR agencies
* Add multi-language support with Amazon Comprehend
* Extend with OCR correction for poorly scanned PDFs
* Integrate with email ingestion to trigger resume processing

**Technologies & Tools**

|  |  |
| --- | --- |
| **Category** | **Tech/Tool** |
| Cloud Platform | AWS |
| Serverless | AWS Lambda |
| Document Parsing | AWS Textract |
| Storage | Amazon S3 |
| Database | DynamoDB |
| Messaging | Amazon SNS |
| Language | Python 3.x |

**Security & Compliance**

* Uses IAM roles to restrict access to AWS services
* Textract operates in-memory (no intermediate file storage)
* SNS messages can be secured via email subscriptions or encrypted topics
* DynamoDB access is limited to Lambda role only

**Author & Ownership**

This project was designed and implemented by [Your Name], a Cloud and Security-focused DevOps Engineer passionate about solving real-world problems using intelligent, scalable, and accessible technology.

**Appendix**

* Lambda function code
* Sample SNS message formats
* DynamoDB schema design
* Test resumes and scoring examples
* Deployment guide