ATS Resume Compatibility Auditor - System Design Documentation

Executive Summary

This document outlines the design of an ATS (Applicant Tracking System) compatibility auditor built in Genies to help users optimize their resumes for automated screening systems.

System Overview

Purpose

To analyze resumes and provide actionable feedback on ATS compatibility, helping job seekers improve their chances of passing automated screening filters.

Core Functionality

- Resume parsing simulation
- Keyword optimization analysis
- Format compatibility checking
- ATS-friendly suggestions
- Compatibility scoring

Technical Architecture

1. Resume Parser Module

Purpose: Simulate how ATS systems extract information from resumes

Components:

- **Text Extraction Engine**: Handles PDF, DOCX, and TXT formats
- **Section Identifier**: Recognizes standard resume sections (Contact, Experience, Education, Skills)
- **Data Structurer**: Organizes extracted information into standardized fields

Key Functions:

```
parseResume(file) -> {
  personalInfo: {},
  experience: [],
  education: [],
  skills: [],
  sections: []
}
```

2. Keyword Analysis Engine

Purpose: Analyze keyword density and relevance

Components:

- **Job Description Processor**: Extracts required keywords from job postings
- **Resume Keyword Extractor**: Identifies keywords in resumes
- **Relevance Scorer**: Calculates keyword match percentage
- **Industry Database**: Maintains industry-specific keyword libraries

Scoring Algorithm:

- Exact keyword matches: 100% weight
- Synonyms/variations: 80% weight
- Related terms: 60% weight
- Context relevance: Variable weight (20-40%)

3. Format Compatibility Checker

Purpose: Evaluate resume formatting for ATS readability

Compatibility Rules:

- **Preferred Formats**: PDF, DOCX, TXT
- **Font Requirements**: Standard fonts (Arial, Calibri, Times New Roman)
- **Layout Constraints**: Single-column preferred, minimal graphics
- **Section Headers**: Clear, standard naming conventions
- **Contact Information**: Proper placement and formatting

4. Content Analysis Module

Components:

- **Experience Validator**: Checks for proper date formats, job titles
- **Skills Matcher**: Compares skills against job requirements
- **Achievement Quantifier**: Identifies measurable accomplishments
- **Grammar Checker**: Basic language quality assessment

5. Scoring and Recommendation Engine

Scoring Categories (100-point scale):

- **Parseability** (25 points): How well ATS can extract information

- **Keyword Optimization** (30 points): Relevance to job requirements

- **Format Compliance** (20 points): ATS-friendly formatting

- **Content Quality** (25 points): Professional presentation and completeness

User Interface Design

Dashboard Components

1. **Upload Interface**: Drag-and-drop resume upload with format validation

2. **Job Description Input**: Optional field to paste job posting for targeted analysis

3. **Analysis Results Panel**: Visual breakdown of compatibility scores

4. **Recommendations Section**: Prioritized list of improvements

5. **Before/After Comparison**: Side-by-side view of original vs. optimized content

Analysis Report Structure

```
ATS Compatibility Report
├──── Overall Score: X/100
├──── Parseability Analysis
│    ├──── Text Extraction: Pass/Fail
│    ├──── Section Recognition: X/Y sections found
│    └──── Data Structure: Complete/Incomplete fields
├──── Keyword Analysis
│    ├──── Match Rate: X% of required keywords found
│    ├──── Missing Keywords: [list]
│    └──── Keyword Density: Optimal/Too Low/Too High
├──── Format Assessment
│    ├──── File Format: Compatible/Needs Change
│    ├──── Layout: ATS-Friendly/Needs Improvement
│    └──── Fonts & Styling: Compliant/Issues Found
└──── Recommendations
     ├──── High Priority: [critical issues]
     ├──── Medium Priority: [optimization opportunities]
     └──── Low Priority: [minor improvements]
```

Implementation Specifications

Data Models

```
javascript
```

```
// Resume Data Model
{
  id: String,
  fileName: String,
  uploadDate: Date,
  parsedContent: {
    personalInfo: {
      name: String,
      email: String,
      phone: String,
      location: String
    },
    experience: [{
      title: String,
      company: String,
      dates: String,
      description: String,
      achievements: [String]
    }],
    education: [{
      degree: String,
      institution: String,
      year: String,
      gpa: String
    }],
    skills: [String],
    certifications: [String]
  },
  analysisResults: {
    overallScore: Number,
    parseabilityScore: Number,
    keywordScore: Number,
    formatScore: Number,
    contentScore: Number,
    recommendations: [RecommendationObject]
  }
}
```

API Endpoints

POST /api/resume/upload - Upload resume file

POST /api/resume/analyze - Analyze resume for ATS compatibility

GET /api/resume/:id/report - Retrieve analysis report

POST /api/job-description/match - Match resume against specific job posting

GET /api/keywords/industry/:industry - Get industry-specific keywords

Processing Pipeline

1. **File Upload & Validation**
   - Format verification
   - File size limits
   - Security scanning

2. **Content Extraction**
   - Text parsing
   - Structure identification
   - Data normalization

3. **Analysis Execution**
   - Keyword analysis
   - Format checking
   - Content quality assessment

4. **Report Generation**
   - Score calculation
   - Recommendation prioritization
   - Visual report creation

ATS Compatibility Rules Database

Format Requirements

- **Acceptable Formats**: PDF (text-based), DOCX, TXT
- **Avoid**: Images, tables, text boxes, headers/footers with critical info
- **Font Guidelines**: 10-12pt size, standard fonts only
- **Layout**: Single column, clear section breaks

Section Standards

```
Required Sections:
- Contact Information (top of resume)
- Professional Experience/Work History
- Education
- Skills

Optional but Recommended:
- Professional Summary/Objective
- Certifications
- Projects (for technical roles)

Section Naming Conventions:
- Use standard headers (Experience, Education, Skills)
- Avoid creative names ("My Journey" instead of "Experience")
```

Content Guidelines

- **Date Formats**: MM/YYYY or Month YYYY

- **Job Titles**: Standard industry terminology

- **Company Names**: Full legal names preferred

- **Achievements**: Quantified with numbers/percentages

- **Skills**: Exact keyword matches for required technologies

Quality Assurance & Testing

Test Scenarios

1. **Format Testing**: Various file formats and layouts

2. **Keyword Testing**: Different keyword densities and variations

3. **Edge Cases**: Unusual formatting, international resumes

4. **Performance Testing**: Large file handling, concurrent users

Success Metrics

- **Accuracy**: 95% correct identification of ATS issues

- **Performance**: Analysis completion under 30 seconds

- **User Satisfaction**: 90% user approval rating

- **Success Rate**: 80% of users report improved ATS pass-through rates

Security & Privacy Considerations

- **Data Encryption**: All resume data encrypted at rest and in transit

- **Data Retention**: User-controlled deletion options
- **Privacy Policy**: Clear disclosure of data usage
- **Access Controls**: Secure user authentication and authorization

Future Enhancements

1. **Industry-Specific Analysis**: Tailored rules for different sectors
2. **Real-time ATS Testing**: Integration with actual ATS platforms
3. **Resume Builder Integration**: Direct optimization during resume creation
4. **Mobile Application**: On-the-go resume checking
5. **Batch Processing**: Analyze multiple resume versions simultaneously

Conclusion

This ATS compatibility auditor will provide comprehensive analysis and actionable recommendations to help users optimize their resumes for automated screening systems, significantly improving their chances of reaching human recruiters.