



Bulk Data Import

with RChilli

June 2024 | Version 1.0

Copyright © 2024, RChilli Inc.



Purpose Statement

This document details integrating **Bulk Data Import** use case leveraging Resume and JD Parsing.

Disclaimer:

Licensed Materials - Property of RChilli

For information about RChilli trademarks, copyrights, and patents, refer to the RChilli Intellectual Property page

(<https://www.rchilli.com/>) on the RChilli website. All other trademarks and registered trademarks are the property of their respective holders.

No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of RChilli Inc. Under the law, reproducing includes translating into another language or format.

Every effort has been made to ensure that the information in this manual is accurate. RChilli Inc. is not responsible for printing or clerical errors. Information in this document is subject to change without notice.

If you have any comments or feedback on our documentation, please send them to us at support@rchilli.com

Contents

Introduction	4
Key Components and Flow.....	5
Benefits	5
Example Scenario	6
User Acceptance Testing (UAT)	7
UAT Scenario 1: Resume/Job Upload Test.....	7
UAT Scenario 2: FTP Data Transfer Test.....	7
UAT Scenario 3: Webhook Response Test	7
UAT Scenario 4: Data Parsing and Validation Test.....	8
UAT Scenario 5: Database Ingestion Test	8
Test Cases	9
Test Case 1: Resume Upload Test.....	9
Test Case 2: Job Description Upload Test	9
Test Case 3: FTP Transfer Test.....	9
Test Case 4: Webhook Response Test	10
Test Case 5: Data Parsing and Validation Test.....	10
Test Case 6: Database Ingestion Test	11
Test Case 7: Resume Format Validation Test.....	11
Test Case 8: Job Description Format Validation Test.....	11
Test Case 9: Large File Upload Test	12
Test Case 10: Multiple File Upload Test.....	12
Test Case 11: FTP Connection Failure Test	12
Test Case 12: Webhook Failure Test.....	13
Test Case 13: Data Integrity Test.....	13
Test Case 14: Database Performance Test.....	13
Test Case 15: Security Test for Data Transfers	14
Miscellaneous.....	15
Integration Estimation.....	15
Appendix	16



Introduction

The **Bulk Import Data** feature streamlines the process of adding large volumes of resumes and job postings into your ATS.

Utilizing the RChilli Bulk Upload, the system automates the extraction of relevant information from uploaded resumes and job descriptions, significantly reducing the time and effort required for data entry.

Bulk Import simplifies the data management process by:

1. **Uploading Data:** Users upload candidate resumes and job postings.
2. **Bulk Upload:** The RChilli Bulk Upload service processes the uploaded files from FTP.
3. **Data Parsing:** The system instantly processes the uploaded data using the RChilli Bulk Upload.
4. **Database Ingestion:** Extracted data is used to populate the ATS database, expediting the creation of candidate profiles and job postings.

Impact

This feature saves 85% of recruiters' time spent on creating candidate profiles and job postings, significantly enhancing overall recruitment productivity by automating data entry processes.



Key Components and Flow

1. Resume/Job Upload:

- a. **Action:** The client uploads resume, job descriptions, or ZIP files on the web page.
- b. **Response:** The uploaded files are sent to the FTP server for processing.

2. FTP Transfer:

- a. **Action:** Establish a secure FTP connection for data transfer.
- b. **Response:** Securely transfer uploaded files to the FTP server.

3. Bulk Upload Processing:

- a. **Action:** The RChilli Bulk Upload service processes the uploaded files from the FTP server.
- b. **Response:** Files are parsed and structured by the RChilli Bulk Upload.

4. Webhook Integration:

- a. **Action:** Configure a webhook to receive parsed data responses.
- b. **Response:** The processed data is posted to the integrated webhook.

5. Data Storage:

- a. **Action:** Process the response JSON received via the webhook.
- b. **Response:** Store the parsed data in the database for ATS.

Benefits

- 1. **Time Efficiency:** Reduces manual data entry effort.
- 2. **Automated Processing:** Ensures accurate and consistent data extraction and import.
- 3. **Scalability:** Handles large volumes of data effectively.
- 4. **Improved Productivity:** Allows recruiters to focus on high-value tasks.



User Stories

1. Upload Resume/Job Files

- As a recruiter, I want to upload multiple resumes or job descriptions so that I can efficiently import large volumes of candidate and job data.

2. FTP Transfer

- As a system, I need to securely transfer uploaded files to the FTP server to ensure data integrity and security.

3. Parse Uploaded Files

- As a system, I need to parse the uploaded resumes or job descriptions to extract relevant data and present it in a structured format.

4. Store Parsed Data

- As a system, I need to store the parsed data in the database to maintain accurate records and facilitate the recruitment process.

Example Scenario

1. **Scenario:** A recruitment agency receives a large batch of resumes from a job fair.
2. **Action:** The agency uploads the resumes to the web page and configures the webhook in their RChilli account.
3. **Response:** The RChilli Bulk Upload system processes the resumes, parses the data, and sends the response to the webhook.
4. **Outcome:** The parsed candidate profiles are automatically stored in the database, significantly reducing manual data entry time for recruiters.



User Acceptance Testing (UAT)

Conduct comprehensive testing to ensure the feature performs as expected and integrates seamlessly with existing systems.

UAT Scenarios

UAT Scenario 1: Resume/Job Upload Test

- **Objective:** Verify successful upload and transfer of resume and job files to the FTP server.
- **Preconditions:** Resume or job files ready for upload.
- **Steps:**
 - i. Navigate to the web page.
 - ii. Upload resume or job files.
- **Expected Result:** Files are successfully uploaded and accessible on the FTP server.
- **Acceptance Criteria:** No errors during upload and files are correctly transferred.

UAT Scenario 2: FTP Data Transfer Test

- **Objective:** Ensure files are successfully transferred to the FTP server.
- **Preconditions:** Files are uploaded to the web page.
- **Steps:**
 - i. Upload files to the web page.
 - ii. Verify the presence of files on the FTP server.
- **Expected Result:** Files are transferred securely and are available on the FTP server.
- **Acceptance Criteria:** No errors during transfer and files are correctly accessible on the FTP server.

UAT Scenario 3: Webhook Response Test

- **Objective:** Ensure webhook receives and processes data correctly.

- **Preconditions:** FTP server with uploaded files and configured webhook.
- **Steps:**
 - i. Upload files to the FTP server.
 - ii. Verify webhook receives the processed data.
- **Expected Result:** Webhook receives and processes data correctly.
- **Acceptance Criteria:** Webhook triggers without delay and processes the data accurately.

UAT Scenario 4: Data Parsing and Validation Test

- **Objective:** Validate accurate parsing of resume and job data.
- **Preconditions:** Uploaded files and active webhook integration.
- **Steps:**
 - i. Trigger data parsing.
 - ii. Verify parsed data accuracy.
- **Expected Result:** Data is parsed accurately and validated.
- **Acceptance Criteria:** Correct data extraction and validation of data fields.

UAT Scenario 5: Database Ingestion Test

- **Objective:** Confirm successful data import into the database for ATS.
- **Preconditions:** Parsed and validated data files.
- **Steps:**
 - i. Complete the data import process.
 - ii. Verify data presence in the database.
- **Expected Result:** Parsed data is correctly imported and visible in the database.
- **Acceptance Criteria:** Data is imported without errors and data integrity is maintained.



Test Cases

Test Case 1: Resume Upload Test

- **Objective:** Verify that resumes can be uploaded successfully.
- **Preconditions:** Candidate has resume files ready.
- **Steps:**
 - a. Navigate to the resume upload page.
 - b. Click the upload button.
 - c. Select and upload a resume file.
- **Expected Result:** Resume upload completes without errors.
- **Test Data:** Various resume formats (PDF, DOCX, ZIP).
- **Pass Criteria:** Resume file is uploaded and available for processing.

Test Case 2: Job Description Upload Test

- **Objective:** Verify that job descriptions can be uploaded successfully.
- **Preconditions:** Job description files are ready.
- **Steps:**
 - a. Navigate to the job description upload page.
 - b. Click the upload button.
 - c. Select and upload a job description file.
- **Expected Result:** Job description upload completes without errors.
- **Test Data:** Various job description formats (PDF, DOCX, ZIP).
- **Pass Criteria:** Job description file is uploaded and available for processing.

Test Case 3: FTP Transfer Test

- **Objective:** Ensure files are transferred to the FTP server correctly.
- **Preconditions:** Files are uploaded to the web page.
- **Steps:**

- a. Upload files to the web page.
- b. Verify the presence of files on the FTP server.
- **Expected Result:** Files are transferred securely and are available on the FTP server.
- **Test Data:** Uploaded files.
- **Pass Criteria:** No errors during transfer, and files are correctly accessible on the FTP server.

Test Case 4: Webhook Response Test

- **Objective:** Verify the webhook receives and processes parsed data correctly.
- **Preconditions:** FTP server with uploaded files and configured webhook.
- **Steps:**
 - a. Upload files to the FTP server.
 - b. Verify the webhook receives the processed data.
- **Expected Result:** Webhook receives and processes data correctly.
- **Test Data:** Processed data files.
- **Pass Criteria:** Webhook triggers without delay and processes the data accurately.

Test Case 5: Data Parsing and Validation Test

- **Objective:** Validate accurate parsing of resume and job data.
- **Preconditions:** Uploaded files and active webhook integration.
- **Steps:**
 - a. Trigger data parsing.
 - b. Verify the accuracy of the parsed data.
- **Expected Result:** Data is parsed accurately and validated.
- **Test Data:** Various resume and job description formats.
- **Pass Criteria:** Correct data extraction and validation of data fields.

Test Case 6: Database Ingestion Test

- **Objective:** Confirm successful data import into the database for ATS.
- **Preconditions:** Parsed and validated data files.
- **Steps:**
 - a. Complete the data import process.
 - b. Verify the presence of data in the database.
- **Expected Result:** Parsed data is correctly imported and visible in the database.
- **Test Data:** Parsed resume and job description data.
- **Pass Criteria:** Data is imported without errors and data integrity is maintained.

Test Case 7: Resume Format Validation Test

- **Objective:** Ensure the system correctly identifies and processes various resume formats.
- **Preconditions:** Resume files in different formats (PDF, DOCX, ZIP) are ready.
- **Steps:**
 - a. Navigate to the resume upload page.
 - b. Upload a resume file in each format.
- **Expected Result:** The system correctly processes each resume format without errors.
- **Test Data:** PDF, DOCX, ZIP resume files.
- **Pass Criteria:** Each resume format is successfully processed and parsed.

Test Case 8: Job Description Format Validation Test

- **Objective:** Ensure the system correctly identifies and processes various job description formats.
- **Preconditions:** Job description files in different formats (PDF, DOCX, ZIP) are ready.
- **Steps:**
 - a. Navigate to the job description upload page.
 - b. Upload a job description file in each format.
- **Expected Result:** The system correctly processes each job description format without errors.



- **Test Data:** PDF, DOCX, ZIP job description files.
- **Pass Criteria:** Each job description format is successfully processed and parsed.

Test Case 9: Large File Upload Test

- **Objective:** Ensure the system can handle large file uploads without performance issues.
- **Preconditions:** Large resume and job description files are ready.
- **Steps:**
 - a. Navigate to the upload page.
 - b. Upload large resume and job description files.
- **Expected Result:** The system successfully uploads and processes the large files.
- **Test Data:** Large resume and job description files (e.g., >100MB).
- **Pass Criteria:** Files are uploaded and processed without errors or significant delays.

Test Case 10: Multiple File Upload Test

- **Objective:** Verify that multiple files can be uploaded and processed simultaneously.
- **Preconditions:** Multiple resume and job description files are ready.
- **Steps:**
 - a. Navigate to the upload page.
 - b. Upload multiple resume and job description files at once.
- **Expected Result:** All files are successfully uploaded and processed.
- **Test Data:** Multiple resume and job description files.
- **Pass Criteria:** All files are uploaded and processed without errors.

Test Case 11: FTP Connection Failure Test

- **Objective:** Verify the system's behavior when the FTP server is unavailable.
- **Preconditions:** FTP server is intentionally set to be unavailable.
- **Steps:**
 - a. Attempt to upload files to the FTP server.



- **Expected Result:** The system should handle the error gracefully and provide an appropriate error message.
- **Test Data:** Resume and job description files.
- **Pass Criteria:** The system shows an appropriate error message and does not crash.

Test Case 12: Webhook Failure Test

- **Objective:** Ensure the system handles webhook failures correctly.
- **Preconditions:** Webhook is intentionally configured incorrectly or set to be unavailable.
- **Steps:**
 - a. Upload files to the FTP server.
 - b. Verify the system's response to the webhook failure.
- **Expected Result:** The system handles the error gracefully and provides an appropriate error message.
- **Test Data:** Resume and job description files.
- **Pass Criteria:** The system shows an appropriate error message and does not crash.

Test Case 13: Data Integrity Test

- **Objective:** Verify the integrity of the parsed data after upload and processing.
- **Preconditions:** Files are uploaded, and parsing is complete.
- **Steps:**
 - a. Compare the parsed data with the original uploaded data.
- **Expected Result:** The parsed data accurately reflects the original data.
- **Test Data:** Sample resume and job description files.
- **Pass Criteria:** No data is lost or incorrectly parsed.

Test Case 14: Database Performance Test

- **Objective:** Assess the performance of the database ingestion process with a high volume of data.
- **Preconditions:** Large number of parsed data entries ready for database ingestion.



- **Steps:**
 - a. Initiate the database ingestion process with a high volume of parsed data.
- **Expected Result:** The database handles the high volume of data without performance degradation.
- **Test Data:** Large dataset of parsed resumes and job descriptions.
- **Pass Criteria:** Data is ingested in a timely manner without significant delays or errors.

Test Case 15: Security Test for Data Transfers

- **Objective:** Ensure the security of data transfers between the web page, FTP server, and webhook.
 - **Preconditions:** Security protocols (e.g., SSL/TLS) are configured for data transfers.
 - **Steps:**
 - a. Monitor data transfer processes.
 - b. Check for encrypted data transfers.
 - **Expected Result:** All data transfers are secure and encrypted.
 - **Test Data:** Sample resume and job description files.
 - **Pass Criteria:** Data transfers are encrypted and secure without vulnerabilities.
-



Miscellaneous

The integration depends on your application workflow. There might be different steps involved in your integration which may vary from application to application.

This document indicates basic steps which are involved, you can review your application and add other UAT, and test cases based upon your needs.

Consider data privacy, error handling, and logging mechanisms to ensure a robust and reliable implementation. Ensure that data transfers are secure and comply with relevant data protection regulations.

Integration Estimation

Disclaimer: This estimation is based on a sample application with simple workflow. This may vary depending on the complexity of your application and the skill set/experience of the team involved in the integration.

1. **Development: 40-45 hours**
 - a. FTP Integration: 6 hours
 - b. Data Parsing: 10 hours
 - c. Database Ingestion: 10 hours
 - d. Webhook Setup and Automated Import: 14 hours
2. **Testing: 15 hours**
 - a. Unit Tests: 5 hours
 - b. Integration Tests: 5 hours
 - c. UAT: 5 hours



Appendix

Resume Parser API Details:

https://docs.rchilli.com/kc/c_RChilli_resume_parser_rest_API

Get API key and Endpoints:

https://docs.rchilli.com/kc/c_RChilli_resume_parser_API_authentication

Resume Parser Response Schema:

https://docs.rchilli.com/kc/c_RChilli_resume_parser_response_Schema

Resume Parser Error Code:

https://docs.rchilli.com/kc/c_RChilli_Resume_parser_error_code

JD Parser API Details:

https://docs.rchilli.com/kc/c_RChilli_JD_parser

Get API key and Endpoints:

https://docs.rchilli.com/kc/c_RChilli_JD_parser_Authentication

JD Parser Response Schema:

https://docs.rchilli.com/kc/c_RChilli_JD_parser_Response_Schema

JD Parser Error Code:

https://docs.rchilli.com/kc/c_RChilli_JD_parser_error_code

Resume Parser – Bulk Integration:

https://docs.rchilli.com/kc/c_Rchilli_resume_parser_bulk_upload_integration

JD Parser – Bulk Integration:

https://docs.rchilli.com/kc/c_JD_Parser_bulk_upload



Postman Integration and Sample Code:

<https://documenter.getpostman.com/view/6003669/Sznh1GQX?version=latest#0b486ceb-a8b6-496b-b53b-73c4adbc6987>

Language Supported:

<https://www.rchilli.com/languages>