



**SOFTWARE ENGINEERING PROGRAM**  
**SWENG 894 – CAPSTONE EXPERIENCE**

**Team 2**

**Software Testing Report**

**10 August 2024**

**Jack Taylor**  
**Javier A. Rodriguez**  
**Julio Aira IV**

## Table of Contents

1. Introduction .....	3
1.1 Purpose of the Document.....	3
2. Overview of Testing Approach and Tools.....	3
2.1 Testing Approach .....	3
2.2 Tools Used .....	5
3. Test Case Specifications .....	5
3.1 Test Cases Overview .....	5
4. Test Implementation .....	6
4.1 Automated Unit Testing .....	6
4.2 System Testing.....	6
4.3 User Acceptance Testing (UAT) .....	8
5. Test Results .....	61
5.1 Test Results Overview .....	61
5.2 Coverage Analysis.....	61
5.3 Bug/Fault/Failure Report .....	61
6. Traceability Matrix.....	62
6.1 Traceability Matrix Overview .....	62
6.2 Traceability Matrixes .....	62
7. Conclusion .....	73
7.1 Summary of Findings.....	73
7.2 Overall Quality Assessment .....	73
7.3 Recommendations .....	73

# 1. Introduction

## 1.1 Purpose of the Document

This Software Testing Report is intended to provide a comprehensive overview of the testing practices adopted for the Automated Resume Screening Tool project. The report documents the test processes, methodologies, and outcomes, ensuring that the testing aligns with the project's requirements and goals.

The report will detail the testing practices implemented throughout the project's lifecycle. It will cover various testing levels, including unit testing, system testing, and user acceptance testing. The report will showcase how these practices were applied to ensure that the software meets its functional and non-functional requirements. Additionally, it will explain how these testing practices were integrated into the development process to maintain the quality of the codebase and ensure the reliability of the system.

The testing practices described in this report are closely associated with the software requirements outlined in the Software Requirement Specification (SRS). Each test case is mapped to specific functional and non-functional requirements, ensuring that all aspects of the system are thoroughly tested. The report will illustrate this association through a traceability matrix, demonstrating how the test cases cover all specified requirements.

The overall goal of this Software Testing Report is to indicate the quality of the codebase through detailed test results. This includes documenting any bugs, faults, or failures observed during the testing process. The report aims to provide stakeholders with a clear understanding of the system's robustness, reliability, and readiness for deployment by analyzing the outcomes of the tests executed.

## 2. Overview of Testing Approach and Tools

### 2.1 Testing Approach

The testing strategy for the Automated Resume Screening Tool was designed to ensure comprehensive coverage of all functional and non-functional requirements. The strategy adopted a multi-tiered approach that included unit testing for individual components, system testing for integrated components, and user acceptance testing (UAT) to validate the system against user expectations. This layered approach allowed for early detection of issues, ensuring that they were addressed promptly to maintain the quality and reliability of the system.

## **Discussion of the Types of Testing Used:**

- **Unit Testing:**
  - Unit testing was employed to validate the functionality of individual components, particularly the Python scripts for resume parsing and the React components in the front-end. Each unit test focused on a specific piece of functionality, ensuring that it performed as expected in isolation. The goal was to achieve high code coverage and detect any defects at an early stage in the development process.
- **System Testing:**
  - System testing was conducted to validate the interaction between different components of the system. This included testing the integration between the front-end and back-end, the database interactions, and the notification services. System testing aimed to ensure that the components worked together seamlessly and that the system met its overall requirements.
- **User Acceptance Testing (UAT):**
  - UAT was performed with a focus on the end-users' perspective, primarily HR professionals who would be using the Automated Resume Screening Tool. The UAT process involved validating the system against real-world scenarios, ensuring that the tool was user-friendly, met business goals, and provided accurate results in resume screening. This phase included feedback from potential users to refine the system further.

Testing was integrated into the development process using a continuous integration/continuous deployment (CI/CD) pipeline. Automated tests were executed as part of the CI/CD workflow, ensuring that every code change was tested before being merged into the main codebase. This approach allowed for immediate feedback on the quality of the code and ensured that the system remained stable throughout the development process. Additionally, test-driven development (TDD) practices were encouraged, where tests were written before the implementation, ensuring that the code met the specified requirements from the outset.

## 2.2 Tools Used

Tool	Why Chosen	How Used
VS Code	Robust features, including IntelliSense, debugging, and extensive plugin support.	Primary IDE for writing, debugging, and managing code in TypeScript and Python.
Jira	Excellent for agile project management with integration support for CI/CD tools.	Tracked user stories, tasks, and sprints to manage workflows and ensure project alignment.
GitHub	Industry-standard for version control with strong collaboration features.	Hosted the source code, facilitated branching, pull requests, and code reviews.
Babel	Compiles TypeScript/JavaScript ensuring compatibility with older browsers.	Compiled modern JavaScript and TypeScript into widely supported formats.
Jest	Simple, integrated testing framework with great support for React and TypeScript.	Used for unit testing front-end components and generating coverage reports.
Cypress	Easy setup for end-to-end testing with powerful debugging and real-time reloads.	Automated functional and integration testing simulating user interactions with the application.
GitHub Actions	Automates workflows directly from GitHub, supporting CI/CD tasks.	Managed automated testing, deployments, and releases in response to GitHub events.
IndexDB	Provides fast, flexible NoSQL data storage with cloud-based scalability.	Stored structured resume data for efficient access and analysis by the application.

## 3. Test Case Specifications

### 3.1 Test Cases Overview

The test cases for the Automated Resume Screening Tool were designed to ensure comprehensive coverage of both functional and non-functional requirements. These test cases were developed to validate individual components, system interactions, and user workflows, ensuring that the tool operates as intended across different scenarios. The test cases include unit tests for isolated functions, integration tests for component interactions, and user acceptance tests (UAT) that simulate real-world usage by HR professionals. Each test case is directly linked to specific software requirements as outlined in the Software Requirement Specification (SRS). By mapping test cases to requirements, we ensure that every feature and function of the system is thoroughly tested. This linkage is critical for verifying that the tool meets its intended purpose and adheres to the expected performance and quality standards. A traceability matrix is used to map test cases to their corresponding requirements, providing a clear overview of coverage and helping to identify any gaps in testing.

## 4. Test Implementation

### 4.1 Automated Unit Testing

The source code for the Automated Resume Screening Tool, including all unit tests, can be found in the following GitHub repository: [SWENG894 Team 2 - Sprint Four Branch](#).

The unit test scripts are primarily written using Jest for testing JavaScript and React components, as well as Pytest for testing Python scripts involved in resume parsing and data extraction. These scripts are designed to validate individual components in isolation, ensuring that each function and module behaves as expected under various conditions. The tests cover a range of scenarios, including edge cases, to ensure robustness.

### 4.2 System Testing

System testing for the Automated Resume Screening Tool involved evaluating the complete and integrated application to ensure that all components work together seamlessly. This phase of testing was conducted after unit tests were completed and focused on validating the overall functionality of the system, including interactions between the front-end and back-end, data flow, and user interface performance. The goal was to simulate real-world usage scenarios and verify that the system meets its functional requirements in a production-like environment.

Several key scenarios were tested during the system testing phase:

#### 1. Resume Upload and Processing:

- Tested the end-to-end process of uploading resumes in various formats (PDF, DOCX, TXT), verifying that the system correctly extracts data, categorizes resumes, and stores the information in the database.
- Configurations tested included handling large file sizes, different encoding formats, and concurrent uploads to ensure the system's robustness and performance under various conditions.

#### 2. User Authentication and Account Management:

- Verified the system's ability to handle user account creation, login, and session management.
- Specific scenarios included testing password recovery, multiple failed login attempts, and session continuity across multiple tabs or devices.

#### 3. Dashboard and Analytics Display:

- Tested the functionality of the analytics dashboard, ensuring that it accurately displays recruitment data and provides interactive elements for dynamic analysis.

- Configurations included real-time updates as new resumes were processed and the ability to toggle between summary and detailed views.

#### **4. Real-Time Notifications:**

- Evaluated the system's ability to send real-time notifications for high-priority resume matches.
- Scenarios tested included triggering notifications based on specific criteria, verifying the accuracy and timeliness of these alerts, and ensuring that they integrate correctly with the user interface.

These tests were conducted in a controlled environment that closely mimicked production settings to ensure that the system would perform reliably under actual usage conditions.

### 4.3 User Acceptance Testing (UAT)

Test Type: Unit	Context: Resume Upload	Test ID: UT-01 – Upload Resume in Multiple Formats
Description	This test verifies that the system can successfully upload resumes in PDF, DOCX, and TXT formats and confirm that the upload was successful.	
Test URL	<a href="https://github.com/jja5458/sweng894team2capstone">https://github.com/jja5458/sweng894team2capstone</a>	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-23">https://jack7aylor.atlassian.net/browse/KAN-23</a>	
Preconditions	<ul style="list-style-type: none"><li>• The HR staff member is logged into the system.</li><li>• The system is operational and can accept file uploads.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Log in as an HR staff member.	The system should authenticate the user and display the dashboard.
2	Navigate to the resume upload section.	The system should display the resume upload interface.
3	Upload a resume file in PDF format.	The system should accept the file and display a success message.
4	Upload a resume file in DOCX format.	The system should accept the file and display a success message.
5	Upload a resume file in TXT format.	The system should accept the file and display a success message.
Post-Conditions	The system should have the uploaded resume files stored and a confirmation of successful uploads should be available.	
Associated Requirements	RUT-1: The system shall provide a user interface for HR users to upload resumes in multiple formats including PDF, DOCX, and TXT.	



Test Type: System	Context: Resume Upload	Test ID: UT-02 – File Size Limit
Description	Verify that the system correctly handles files that exceed the size limit.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-23">https://jack7aylor.atlassian.net/browse/KAN-23</a>	
Preconditions	HR staff is logged in.	
Test Execution Steps		
Step	Action	Expected Response
1	Attempt to upload a resume file that exceeds the system’s maximum file size limit.	System responds with dialog informing the user that file size is too large.
Post-Conditions	The system should not store files that exceed the size limit.	
Associated Requirements	RUT-1 (interface must handle file uploads appropriately), RUT-3 (storing data correctly in a structured format).	

Test Type: System	Context: Resume Upload	Test ID: UT-03 – Resume Content Parsing
Description	Verify that the system can parse and extract information from resumes correctly, regardless of the file format.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-23">https://jack7aylor.atlassian.net/browse/KAN-23</a>	
Preconditions	HR staff is logged in.	
Test Execution Steps		
Step	Action	Expected Response
1	Upload a well-formatted resume in PDF, DOCX, and TXT formats.	The system should parse the resumes and display extracted information for verification.
Post-Conditions	The system should store extracted data from the resume correctly in DynamoDB.	
Associated Requirements	RUT-2 (utilize AWS Comprehend for data extraction), RUT-3 (store data in structured format).	

Test Type: System	Context: Resume Upload	Test ID: UT-04 – Simultaneous Uploads
Description	Check how the system handles multiple files being uploaded at the same time.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-23">https://jack7aylor.atlassian.net/browse/KAN-23</a>	
Preconditions	HR staff is logged in.	
Test Execution Steps		
Step	Action	Expected Response
1	Select multiple files in the supported formats and upload them simultaneously.	The system should successfully upload all files and confirm each upload was successful.
Post-Conditions	The system should store all successfully uploaded files.	
Associated Requirements	RUT-1 (handle multiple file uploads through the user interface).	

Test Type:	Context: Resume Upload	Test ID: UT-05 – Resume Upload Progress
Description	Ensure that the system provides feedback on the upload progress for larger files.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-23">https://jack7aylor.atlassian.net/browse/KAN-23</a>	
Preconditions	HR staff is logged in.	
Test Execution Steps		
Step	Action	Expected Response
1	Upload a large resume file.	The system should display a progress bar or similar indicator during the upload process.
Post-Conditions	The user should receive feedback about the upload process until completion.	
Associated Requirements	RUT-1 (interface provides feedback during uploads).	

Test Type: System	Context: User Accounts	Test ID: UT-06 – Account Creation
Description	Verify that an HR staff member can create a new user account.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-29">https://jack7aylor.atlassian.net/browse/KAN-29</a>	
Preconditions	HR staff has access to the account creation page.	
Test Execution Steps		
Step	Action	Expected Response
1	Navigate to the account creation page.	System displays the account creation form.
2	Input a valid username and password.	System accepts the username and password input.
3	Submit the account creation form.	System creates the account and displays a success message.
Post-Conditions	New user account exists in the system.	
Associated Requirements	RUT-6 (support user account creation).	

Test Type: System	Context: User Accounts	Test ID: UT-07 – Account Login
Description	Ensure that HR staff can log into their account using valid credentials.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-29">https://jack7aylor.atlassian.net/browse/KAN-29</a>	
Preconditions	HR staff has an existing user account.	
Test Execution Steps		
Step	Action	Expected Response
1	Navigate to the login page.	System displays the login form.
2	Enter valid username and password.	System accepts the credentials.
3	Click on the login button.	System grants access and redirects to the HR dashboard.
Post-Conditions	HR staff is logged into the system.	
Associated Requirements	RUT-7 (allow user authentication).	

Test Type: System	Context: User Accounts	Test ID: UT-08 – Password Reset
Description	Verify that HR staff can reset their password if forgotten.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-29">https://jack7aylor.atlassian.net/browse/KAN-29</a>	
Preconditions	HR staff has an existing account and has access to their registered email.	
Test Execution Steps		
Step	Action	Expected Response
1	Navigate to the login page.	System displays the login form with a "Forgot Password" option.
2	Click on the "Forgot Password" link.	System displays a form to enter the email.
3	Enter the registered email and submit the request.	System sends a reset link to the provided email.
4	Follow the email link to reset the password.	System allows setting a new password.
5	Enter a new password and confirm it.	System confirms the password has been reset and redirects to the login page.
Post-Conditions	HR staff's password is updated in the system.	
Associated Requirements	RUT-6 (support user account creation).	

Test Type: System	Context: User Accounts	Test ID: UT-09 – Unauthorized Login Attempt
Description	Check the system's response to repeated login attempts with incorrect credentials.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-29">https://jack7aylor.atlassian.net/browse/KAN-29</a>	
Preconditions	HR staff has an existing user account.	
Test Execution Steps		
Step	Action	Expected Response
1	Navigate to the login page.	System displays the login form.
2	Enter incorrect credentials and attempt to log in five times.	System rejects each login attempt and eventually locks the account or requires additional verification after multiple failed attempts.
Post-Conditions	Account is either locked temporarily or requires additional verification.	
Associated Requirements	RUT-7 (authentication), RUT-6 (account security features).	

Test Type: System	Context: User Accounts	Test ID: UT-10 – Account Deletion
Description	Verify that an admin can delete an HR staff member's account.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-29">https://jack7aylor.atlassian.net/browse/KAN-29</a>	
Preconditions	Admin is logged into the system with administrative privileges. HR staff account exists and is identified for deletion.	
Test Execution Steps		
Step	Action	Expected Response
1	Navigate to the admin dashboard.	System displays the admin dashboard.
2	Access the account management section.	System shows the account management interface with a list of HR staff accounts.
3	Locate the HR staff member's account to be deleted.	System identifies and displays the selected HR staff account.
4	Select the option to delete the account.	System provides an option to delete the account.
5	Confirm the deletion when prompted.	System deletes the account and provides a confirmation message that the account has been successfully removed.
Post-Conditions	The HR staff's account is completely removed from the system.	
Associated Requirements	RUT-7 (authentication), RUT-6 (account security features).	

Test Type: System	Context: Application Login	Test ID: UT-11 – Relogin After Logout
Description	Confirm that the system requires re-login after the user has logged out.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-30">https://jack7aylor.atlassian.net/browse/KAN-30</a>	
Preconditions	HR staff is logged into the system and then logs out.	
Test Execution Steps		
Step	Action	Expected Response
1	After logout, attempt to navigate back using the browser's back button.	The system does not allow access to previously viewed pages and redirects to the login page.
Post-Conditions	Secure logout process.	
Associated Requirements	RUT-7 (user authentication).	

Test Type: System	Context: Application Login	Test ID: UT-12 – Session Continuity
Description	Ensure that a user's session is maintained across multiple tabs without needing to log in again.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-30">https://jack7aylor.atlassian.net/browse/KAN-30</a>	
Preconditions	HR staff is logged into the system in one tab.	
Test Execution Steps		
Step	Action	Expected Response
1	Open a new tab and navigate to the HR dashboard.	The system recognizes the existing session and does not require a new login.
Post-Conditions	Session continuity is maintained across tabs.	
Associated Requirements	RUT-7 (secure user authentication and session management).	

Test Type: System	Context: Application Login	Test ID: UT-13 – Case-Sensitive Username
Description	Verify that the username field during login is case-sensitive.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-30">https://jack7aylor.atlassian.net/browse/KAN-30</a>	
Preconditions	HR staff has a username in a specific case format.	
Test Execution Steps		
Step	Action	Expected Response
1	Attempt to log in with the correct username in a different case.	Login fails with username in incorrect case.
2	Attempt to log in with the correct username in the correct case.	Login succeeds with username in correct case.
Post-Conditions	Validates the case sensitivity of the username field.	
Associated Requirements	RUT-7 (user authentication).	

Test Type: System	Context: Application Login	Test ID: UT-14 – Login Redirection
Description	Verify that the system redirects a user not logged in to the login page when trying to access the admin dashboard.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-30">https://jack7aylor.atlassian.net/browse/KAN-30</a>	
Preconditions	HR admin is not logged in.	
Test Execution Steps		
Step	Action	Expected Response
1	Attempt to access a secure HR dashboard page directly via URL.	System redirects to the login page.
Post-Conditions	User remains not logged in until credentials are entered.	
Associated Requirements	RUT-7 (user authentication).	

Test Type:	Context: Application Login	Test ID: UT-15 – Multifactor Authentication
Description	Ensure that the system requires and correctly handles multi-factor authentication for HR staff members during the login process.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-30">https://jack7aylor.atlassian.net/browse/KAN-30</a>	
Preconditions	Multi-factor authentication is enabled for the HR staff account.	
Test Execution Steps		
Step	Action	Expected Response
1	Navigate to the login page.	System displays the login page.
2	Enter valid username and password.	System accepts the username and password, then prompts for the multi-factor authentication code.
3	Submit the login form.	Upon entering the correct MFA code, the system grants access and redirects to the HR dashboard.
4	Enter the code from the multi-factor authentication device or app when prompted.	If an incorrect MFA code is entered, the system denies access and prompts to retry the MFA code.
Post-Conditions	HR staff is either successfully logged in with correct MFA validation or denied access with incorrect MFA input.	
Associated Requirements	RUT-7 (user authentication should include a secure multi-factor process).	

Test Type: Unit	Context: Upload Resume with Rich Formatting	Test ID: UT-16
Description	Verify that the system accurately processes resumes with complex formatting (e.g., multiple columns, text boxes).	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-25">https://jack7aylor.atlassian.net/browse/KAN-25</a>	
Preconditions	<ul style="list-style-type: none"><li>HR user is logged into the system.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Log in as an HR staff member.	HR staff member successfully logs into the system.
2	Navigate to the resume upload section.	HR staff member reaches the resume upload section.
3	Select a resume file with rich formatting in DOCX format.	The system recognizes the selected file.
4	Click on the upload button to upload the selected resume file.	The system uploads the file and begins processing.
5	Verify that the system correctly processes the file and extracts data despite complex formatting.	The system processes the file, correctly extracts data despite the complex formatting.
6	Confirm that the extracted data is populated in the candidate's profile accurately.	The candidate's profile is populated with correctly extracted data from the resume file.
Post-Conditions	Extracted data is correctly stored in DynamoDB.	
Associated Requirements	RUT-1, RUT-2, RUT-3	

Test Type: Unit	Context: Upload Empty Resume	Test ID: UT-17
Description	Test system's response to uploading an empty or nearly empty resume file.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-25">https://jack7aylor.atlassian.net/browse/KAN-25</a>	
Preconditions	<ul style="list-style-type: none"><li>HR user is logged into the system.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Log in as an HR staff member.	HR staff member successfully logs into the system.
2	Navigate to the resume upload section.	HR staff member reaches the resume upload section.
3	Select an empty PDF file as a resume.	The system recognizes the selected file.
4	Click on the upload button to upload the file.	The system uploads the file and begins processing.



5	Verify that the system processes the file but recognizes the lack of content.	The system recognizes the lack of content and alerts the HR user that no relevant data could be extracted.
Post-Conditions	No meaningful data stored in DynamoDB; alert or notification might be issued.	
Associated Requirements	RUT-2	

Test Type: Unit	Context: Upload of Incorrectly Formatted PDF	Test ID: UT-18
Description	Ensure that the system handles a PDF resume that is not text-based (e.g., scanned image PDF).	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-25">https://jack7aylor.atlassian.net/browse/KAN-25</a>	
Preconditions	<ul style="list-style-type: none"><li>HR user is logged into the system.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Log in as an HR staff member.	HR staff member successfully logs into the system.
2	Navigate to the resume upload section.	HR staff member reaches the resume upload section.
3	Select a scanned image PDF file as a resume.	The system recognizes the selected file.
4	Click on the upload button to upload the file.	The system uploads the file and begins processing.
5	Verify that the system attempts to extract data but fails due to lack of recognizable text.	The system attempts to extract data but fails, resulting in an error or notification about the issue.
6	Confirm that no text is extracted and the system logs an error or issues a notification.	No text is extracted; system logs the error or issues a notification to the HR user about the issue.
Post-Conditions	No text extracted; system logs error or issues notification.	
Associated Requirements	RUT-1, RUT-2	

Test Type: Unit	Context: Resume Update and Re-extraction	Test ID: UT-19
Description	Test system’s capability to handle the re-upload of a previously uploaded resume with updates.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-25">https://jack7aylor.atlassian.net/browse/KAN-25</a>	
Preconditions	<ul style="list-style-type: none"><li>HR user is logged in; an original resume is already uploaded and processed.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response

1	Log in as an HR staff member.	HR staff member successfully logs into the system.
2	Navigate to the resume upload section.	HR staff member reaches the resume upload section.
3	Select the updated version of the previously uploaded resume.	The system recognizes the selected file.
4	Click on the upload button to upload the updated resume file.	The system uploads the file and begins processing.
5	Verify that the system processes the new file and overwrites existing data.	The system processes the new file, overwrites existing data, and updates the candidate's profile.
6	Confirm that the candidate's profile is updated with the new information.	The candidate's profile is updated with the new information, replacing the old data in DynamoDB.
Post-Conditions	Updated profile reflects the changes, replacing old data in DynamoDB.	
Associated Requirements	RUT-1, RUT-2, RUT-3	

Test Type: System	Context: Upload of Large Resume File	Test ID: UT-20
Description	Evaluate system performance when processing a resume with extensive content.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-25">https://jack7aylor.atlassian.net/browse/KAN-25</a>	
Preconditions	<ul style="list-style-type: none"><li>HR user is logged into the system; a large detailed resume file is ready for upload.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Log in as an HR staff member.	HR staff member successfully logs into the system.
2	Navigate to the resume upload section.	HR staff member reaches the resume upload section.
3	Select a large DOCX resume file with extensive sections and details.	The system recognizes the selected file.
4	Click on the upload button to upload the large resume file.	The system uploads the file and begins processing.
5	Verify that the system processes the file without significant delay.	The system processes the file without significant delay, correctly extracting all required information.
6	Confirm that the candidate's profile is updated with the extracted information.	The candidate's profile is updated with the extracted information, and the data is correctly stored in DynamoDB.
Post-Conditions	Extracted data is stored correctly in DynamoDB, system maintains performance.	

Associated Requirements	RUT-1, RUT-2, RUT-3
-------------------------	---------------------

Test Type: Unit	Context: Storage of Large Resume Data	Test ID: UT-21
Description	Ensure the system can handle and store large amounts of extracted data from a resume on the local server.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-26">https://jack7aylor.atlassian.net/browse/KAN-26</a>	
Preconditions	<ul style="list-style-type: none"><li>Resume data has been extracted from a complex, detailed resume file.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Log in as an HR staff member.	HR staff member successfully logs into the system.
2	Navigate to the data storage section.	HR staff member reaches the data storage section.
3	Trigger the storage process for the large set of extracted data.	The system initiates the storage process.
4	Verify that the system successfully stores all the extracted data.	The system successfully stores all the extracted data on the local server without truncation or data loss.
5	Confirm the stored data is verifiable on the local server.	The stored data is verifiable on the local server and matches the input in terms of completeness and integrity.
Post-Conditions	Data is verifiable on the local server and matches the input in terms of completeness and integrity.	
Associated Requirements	RUT-3	

Test Type: Unit	Context: Data Consistency on Rapid Re-upload	Test ID: UT-22
Description	Verify that rapid consecutive updates to a candidate's resume maintain data consistency in DynamoDB.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-26">https://jack7aylor.atlassian.net/browse/KAN-26</a>	
Preconditions	<ul style="list-style-type: none"><li>A resume file has been uploaded and processed, and then quickly re-uploaded with minor updates.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Log in as an HR staff member.	HR staff member successfully logs into the system.
2	Navigate to the resume upload section.	HR staff member reaches the resume upload section.
3	Upload an initial resume file and wait for it to be processed.	The system processes the initial resume file and updates the candidate's profile.

4	Quickly re-upload an updated resume file, triggering re-extraction and storage.	The system processes the updated resume file and consistently updates the existing records with the new data.
5	Verify that the system does not create duplicate entries or data corruption during rapid updates.	The system maintains data consistency without creating duplicate entries or data corruption.
6	Confirm that the local server reflects the latest data accurately.	The local server reflects the latest data accurately and consistently.
Post-Conditions	Local server reflects the latest data consistently	
Associated Requirements	RUT-3	

Test Type: Unit	Context: Handling of Special Characters and Encoding	Test ID: UT-23
Description	Test the system's ability to correctly handle and store extracted data containing special characters or various encoding formats.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-26">https://jack7aylor.atlassian.net/browse/KAN-26</a>	
Preconditions	<ul style="list-style-type: none"><li>Extracted data includes special characters (e.g., accents, symbols) or non-standard encoding.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Log in as an HR staff member.	HR staff member successfully logs into the system.
2	Navigate to the data storage section.	HR staff member reaches the data storage section.
3	Select the extracted data containing special characters or different encodings.	The system recognizes the selected data.
4	Trigger the storage process for the extracted data.	The system successfully stores the data on the local server, maintaining the integrity of special characters and encodings.
5	Verify that the stored data maintains character integrity.	The stored data is retrievable and matches the input, maintaining the integrity of special characters and encodings.
Post-Conditions	Stored data is retrievable and matches the input for character integrity.	
Associated Requirements	RUT-3	

Test Type: Unit	Context: Resume Data Update	Test ID: UT-24
-----------------	-----------------------------	----------------

Description	Ensure the system can update existing resume data in local server when a resume is re-uploaded with significant changes.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-26">https://jack7aylor.atlassian.net/browse/KAN-26</a>	
Preconditions	<ul style="list-style-type: none"><li>An existing resume in local server has been updated and re-uploaded.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Log in as an HR staff member.	HR staff member successfully logs into the system.
2	Navigate to the resume upload section.	HR staff member reaches the resume upload section.
3	Select the updated version of the existing resume.	The system recognizes the selected file.
4	Click on the upload button to upload the updated resume file.	The system uploads the file and begins processing.
5	Verify that the system processes and stores the updated resume data.	The system updates the existing record in DynamoDB with the new data without duplicating entries.
Post-Conditions	Local server contains the updated data only	
Associated Requirements	RUT-3	

Test Type: System	Context: Concurrent Data Storage	Test ID: UT-25
Description	Test the system's ability to handle concurrent data storage requests without data loss or corruption.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-26">https://jack7aylor.atlassian.net/browse/KAN-26</a>	
Preconditions	<ul style="list-style-type: none"><li>Multiple HR users upload and process different resumes simultaneously.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Log in as multiple HR staff members.	Multiple HR staff members successfully log into the system.
2	Each HR staff member navigates to the resume upload section.	HR staff members reach the resume upload section independently.
3	Each HR staff member selects a different resume file for upload.	The system recognizes the selected files from multiple sources.
4	Simultaneously trigger the upload and storage process for each resume.	The system uploads and begins processing each file concurrently.
5	Verify that the system correctly handles all concurrent storage requests.	The system handles all requests concurrently, correctly storing all data in the local server without errors or data mix-ups.

6	Confirm that data from each resume is independently and correctly stored.	Data from each resume is independently and correctly stored, maintaining data integrity for all concurrent requests.
Post-Conditions	Data from each resume is independently and correctly stored.	
Associated Requirements	RUT-3	

Test Type: Unit	Context: Invalid Input Handling	Test ID: UT-26
Description	Verify that the system correctly handles invalid inputs (e.g., special characters in numeric fields, extremely long text inputs).	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-27">https://jack7aylor.atlassian.net/browse/KAN-27</a>	
Preconditions	<ul style="list-style-type: none"><li>HR user is in the job criteria definition section.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Log in as an HR staff member.	HR staff member successfully logs into the system.
2	Navigate to the job criteria definition section.	HR staff member reaches the job criteria definition section.
3	Enter invalid data types for required skills (numbers), years of experience (text), and educational qualifications (special characters).	The system does not accept these inputs, displays an error message detailing the invalid data, and requests correct inputs.
4	Attempt to save the invalid data.	The system prevents the saving of invalid data and prompts the HR user to correct the inputs.
5	Verify that no incorrect data is saved in the system.	No incorrect data is saved; the system maintains data integrity by only allowing valid inputs to be stored.
Post-Conditions	No incorrect data is saved in the system.	
Associated Requirements	RUT-4	

Test Type: Unit	Context: Update Existing Criteria	Test ID: UT-27
Description	Ensure that previously defined job-specific criteria can be updated correctly.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-27">https://jack7aylor.atlassian.net/browse/KAN-27</a>	
Preconditions	<ul style="list-style-type: none"><li>Job-specific criteria already exist and are stored in the system.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Log in as an HR staff member.	HR staff member successfully logs into the system.
2	Navigate to the job criteria definition section.	HR staff member reaches the job criteria definition section.

3	Access the existing job-specific criteria.	The system displays the existing job-specific criteria.
4	Update the required skills, years of experience, and educational qualifications.	The system accepts the updates.
5	Save the new criteria.	The system saves the new criteria and confirms the successful update.
6	Verify that the updated criteria are correctly reflected in the system.	The updated criteria are correctly reflected in the system, ensuring that the new requirements are stored accurately.
Post-Conditions	Updated criteria are correctly reflected in the system.	
Associated Requirements	RUT-4	

Test Type: Unit	Context: Role-Based Access Control	Test ID: UT-28
Description	Test that only authorized HR users can define and modify job-specific criteria.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-27">https://jack7aylor.atlassian.net/browse/KAN-27</a>	
Preconditions	<ul style="list-style-type: none"><li>Different users (HR and non-HR) attempt to access the criteria definition section.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Log in as a non-HR user.	Non-HR user successfully logs into the system.
2	Navigate to the job criteria definition section.	The system restricts access and displays an authorization error message to the non-HR user.
3	Attempt to define job-specific criteria as a non-HR user.	The system restricts access and prevents the non-HR user from defining job-specific criteria.
4	Log in as an HR user.	HR user successfully logs into the system.
5	Navigate to the job criteria definition section.	HR user reaches the job criteria definition section.
6	Verify that only HR users can access and modify job-specific criteria.	The system allows HR users to define and modify job-specific criteria, while restricting non-HR users.
Post-Conditions	Job-specific criteria remain secure and unaltered by unauthorized access.	
Associated Requirements	RUT-4	

Test Type: Unit	Context: Criteria Deletion	Test ID: UT-29
Description	Ensure the system handles the deletion of existing job-specific criteria correctly.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-27">https://jack7aylor.atlassian.net/browse/KAN-27</a>	

Preconditions	• Job-specific criteria have been defined and stored.	
Test Execution Steps		
Step	Action	Expected Response
1	Log in as an HR staff member.	HR staff member successfully logs into the system.
2	Navigate to the job criteria definition section.	HR staff member reaches the job criteria definition section.
3	Select the set of job-specific criteria to delete.	The system recognizes the selected criteria.
4	Click on the delete button to remove the criteria.	The system successfully removes the criteria and confirms the deletion.
5	Verify that the deleted criteria are no longer retrievable.	The deleted criteria are no longer retrievable from the system, confirming the deletion.
Post-Conditions	The deleted criteria are no longer retrievable from the system.	
Associated Requirements	RUT-4	



Test Type: System	Context: Simultaneous Criteria Updates	Test ID: UT-30
Description	Test the system's handling of simultaneous updates to job-specific criteria from multiple HR users.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-27">https://jack7aylor.atlassian.net/browse/KAN-27</a>	
Preconditions	<ul style="list-style-type: none"><li>Multiple HR users are logged in and attempt to update the same job criteria simultaneously.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Log in as multiple HR staff members.	Multiple HR staff members successfully log into the system.
2	Navigate to the job criteria definition section.	HR staff members reach the job criteria definition section.
3	Each HR staff member attempts to update the same job criteria simultaneously.	The system manages concurrent updates, ensuring data consistency.
4	Each HR staff member clicks the save button to submit their updates.	The system processes and saves the updates, ensuring there are no conflicts or data corruption.
5	Verify that all updates are accurately reflected in the system.	All updates are accurately reflected in the system without overlap or error, maintaining data integrity.
Post-Conditions	All updates are reflected accurately without overlap or error.	
Associated Requirements	RUT-4	

Test Type: Unit	Context: Ranking with Incomplete Criteria	Test ID: UT-31
Description	Verify the system's behavior when job-specific criteria are partially defined or incomplete.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-28">https://jack7aylor.atlassian.net/browse/KAN-28</a>	
Preconditions	<ul style="list-style-type: none"><li>Job-specific criteria are only partially saved (e.g., required skills are defined, but years of experience are not).</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Log in as an HR staff member.	HR staff member successfully logs into the system.
2	Navigate to the job criteria definition section.	HR staff member reaches the job criteria definition section.
3	Define partial job-specific criteria (e.g., only required skills).	The system saves the partially defined criteria.
4	Attempt to rank resumes based on the incomplete criteria.	The system either ranks resumes based on the available data or alerts the user that the criteria are incomplete and ranking might be inaccurate.

5	Verify the system's response to the incomplete criteria.	The system either ranks the resumes based on partial criteria or prompts the user to complete the criteria setup.
Post-Conditions	Resumes are either ranked based on partial criteria or the user is prompted to complete the criteria setup.	
Associated Requirements	RUT-5	

Test Type: Unit	Context: Ranking Stability with Duplicate Criteria Scores	Test ID: UT-32
Description	Test the stability and consistency of the ranking process when multiple resumes score identically based on the job-specific criteria.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-28">https://jack7aylor.atlassian.net/browse/KAN-28</a>	
Preconditions	<ul style="list-style-type: none"><li>Multiple resumes that meet the criteria identically are uploaded.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Log in as an HR staff member.	HR staff member successfully logs into the system.
2	Navigate to the resume upload section.	HR staff member reaches the resume upload section.
3	Upload multiple resumes that meet the job-specific criteria identically.	The system recognizes and processes the uploaded resumes.
4	Initiate the ranking process for the resumes.	The system ranks the resumes, ensuring that those with identical scores are ranked in a stable and consistent order.
5	Verify the stability and consistency of the ranking process.	The system consistently ranks the identical resumes in a repeatable and predictable manner, possibly using secondary criteria or timestamp.
Post-Conditions	Resumes with identical scores are ranked in a repeatable and predictable manner.	
Associated Requirements	RUT-5	

Test Type: Unit	Context: Edge Cases in Criteria Matching	Test ID: UT-33
Description	Ensure the system can handle edge cases in job-specific criteria (e.g., extreme values, boundary conditions).	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-28">https://jack7aylor.atlassian.net/browse/KAN-28</a>	
Preconditions	<ul style="list-style-type: none"><li>Resumes with edge case values (e.g., maximum or minimum years of experience possible) are uploaded.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response

1	Log in as an HR staff member.	HR staff member successfully logs into the system.
2	Navigate to the resume upload section.	HR staff member reaches the resume upload section.
3	Upload resumes with edge case values (e.g., maximum or minimum years of experience).	The system recognizes and processes the uploaded resumes.
4	Initiate the ranking process for the resumes.	The system accurately processes and ranks the resumes, reflecting correct prioritization based on the criteria.
5	Verify the system handles edge case values without errors.	The edge case resumes are correctly ranked and stored without any processing errors.
Post-Conditions	Edge case resumes are correctly ranked and stored.	
Associated Requirements	RUT-5	

Test Type: System	Context: Ranking Performance Under High Load	Test ID: UT-34
Description	Evaluate the performance and responsiveness of the ranking system under a high load of resume uploads and ranking requests.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-28">https://jack7aylor.atlassian.net/browse/KAN-28</a>	
Preconditions	<ul style="list-style-type: none"><li>A large number of resumes are uploaded simultaneously, and the ranking process is initiated.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Log in as an HR staff member.	HR staff member successfully logs into the system.
2	Navigate to the resume upload section.	HR staff member reaches the resume upload section.
3	Upload a large number of resumes simultaneously.	The system recognizes and processes all the uploaded resumes.
4	Initiate the ranking process for the high volume of resumes.	The system processes and ranks the high volume of resumes without significant delays, maintaining performance.
5	Verify the system's performance and responsiveness under high load.	System performance metrics are logged, and all resumes are ranked as expected, demonstrating the system's robustness.
Post-Conditions	System performance metrics are logged, and all resumes are ranked as expected.	
Associated Requirements	RUT-5	

Test Type: Unit	Context: Ranking Repeatability and Consistency	Test ID: UT-35
Description	Confirm that the ranking process is repeatable and consistent when performed multiple times under identical conditions.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-28">https://jack7aylor.atlassian.net/browse/KAN-28</a>	
Preconditions	<ul style="list-style-type: none"><li>• A set of resumes is uploaded and ranked.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Log in as an HR staff member.	HR staff member successfully logs into the system.
2	Navigate to the resume upload section.	HR staff member reaches the resume upload section.
3	Upload a set of resumes and initiate the ranking process.	The system processes and ranks the uploaded resumes based on the specified criteria.
4	Repeat the ranking process multiple times with the same set of resumes and criteria.	Each ranking operation yields the same result, demonstrating the system's consistency and repeatability.
5	Verify that the rankings are consistent across trials.	The rankings are consistent across multiple trials, with no variation in ranking order or scores.
Post-Conditions	Rankings are consistent across trials; no variation in ranking order or score.	
Associated Requirements	RUT-5	

Test Type: System	Context: Analytics Dashboard	Test ID: UT-36
Description	As an HR staff member, I want a dashboard that displays analytics so that captured data can be analyzed thoroughly.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-31">https://jack7aylor.atlassian.net/browse/KAN-31</a>	
Preconditions	<ul style="list-style-type: none"><li>• The user must be authenticated and logged in as an HR staff member.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Access the dashboard page.	The system displays the analytics dashboard interface.
2	Ensure resume data is available for analysis.	The dashboard displays analytics based on the available resume data.
3	Interact with various dashboard elements.	The system updates and displays analytics based on user interactions.
Post-Conditions	<ul style="list-style-type: none"><li>• The extracted data (skills, educational background, and professional experience) is stored locally.</li><li>• The candidate's profile is updated with the structured data.</li></ul>	
Associated Requirements	<ul style="list-style-type: none"><li>• RUT-8 The system must provide a dashboard that displays analytics of the resume screening process.</li></ul>	

Test Type: Unit	Context: Dashboard Data Accuracy	Test ID: UT-37
Description	Verify that the dashboard accurately displays extracted data from resumes	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-31">https://jack7aylor.atlassian.net/browse/KAN-31</a>	
Preconditions	HR user is logged in, and resumes have been uploaded and processed.	
Test Execution Steps		
Step	Action	Expected Response
1	Navigate to the analytics dashboard.	System displays the dashboard with resume analytics.
2	Review the displayed data for accuracy.	Data on the dashboard matches the expected extracted data.
3	Refresh the dashboard.	Updated data is accurately reflected on the dashboard.
Post-Conditions	Dashboard displays correct and complete analytics of the resume data.	
Associated Requirements	RUT-8	

Test Type: Unit	Context: Dashboard Interactivity	Test ID: UT-38
Description	Test interactive elements of the dashboard for dynamic data analysis.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-31">https://jack7aylor.atlassian.net/browse/KAN-31</a>	
Preconditions	HR user is logged in and on the dashboard page.	
Test Execution Steps		
Step	Action	Expected Response
1	Access a filter tool on the dashboard.	System displays various filter options (e.g., skills).
2	Apply a specific filter to the data.	Dashboard updates to only show data relevant to filter.
3	Interact with a graphical element.	System displays detailed analytics related to the element.
Post-Conditions	User interactions lead to updated analytics display.	
Associated Requirements	RUT-8	

Test Type: Unit	Context: Dashboard Real-time Update	Test ID: UT-39
Description	Ensure the dashboard reflects real-time updates as new resumes are processed.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-31">https://jack7aylor.atlassian.net/browse/KAN-31</a>	
Preconditions	HR user is logged in with prior resumes processed.	
Test Execution Steps		
Step	Action	Expected Response
1	Observe the initial data presented.	Dashboard displays current resume analytics.
2	Upload and process a new resume.	System processes the resume and updates the database.
3	Observe the dashboard for updates.	Dashboard immediately reflects new data from the resume.
Post-Conditions	New data is immediately represented in the dashboard upon processing.	
Associated Requirements	RUT-8	

Test Type: Unit	Context: Dashboard Summary and Detail Toggle	Test ID: UT-40
Description	Test the dashboard's ability to toggle between summary and detailed analytics views.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-31">https://jack7aylor.atlassian.net/browse/KAN-31</a>	
Preconditions	HR user is logged in and data from multiple resumes is available.	
Test Execution Steps		
Step	Action	Expected Response
1	Navigate to the analytics dashboard.	System displays the default view (e.g., summary analytics).
2	Toggle to the detailed analytics view.	System updates to display detailed analytics of resumes.
3	Toggle back to the summary view.	Dashboard returns to showing summary analytics.
Post-Conditions	The user can view both summary and detailed analytics as needed.	
Associated Requirements	RUT-8, RUT-9	

Test Type: System	Context: Recruitment Status Dashboard	Test ID: UT-41
Description	As an HR staff member, I want a dashboard that displays summaries of the resume screening process so that an overview of the overall process, rather detailed data, is provided.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-32">https://jack7aylor.atlassian.net/browse/KAN-32</a>	
Preconditions	<ul style="list-style-type: none"><li>• AWS Comprehend must be integrated and functional.</li><li>• AWS DynamoDB is set up and accessible for data storage.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Access the summary dashboard page.	The system displays the summary dashboard interface.
2	Ensure resume data is available for summaries.	The dashboard displays summaries based on the available resume data.
3	Verify overall process metrics and summaries.	The system displays an accurate overview of the resume screening process.
4	Check navigation and usability of the dashboard.	The dashboard is easy to navigate and use, displaying clear summaries.
Post-Conditions	<ul style="list-style-type: none"><li>• Extracted data is stored in locally</li><li>• Stored data is retrievable and correctly formatted for use.</li></ul>	
Associated Requirements	RUT-9	

Test Type: Unit	Context: Recruitment Trends Visualization	Test ID: UT-42
Description	Verify the dashboard's ability to display visual trends of the recruitment process.	

Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-32">https://jack7aylor.atlassian.net/browse/KAN-32</a>	
Preconditions	HR user is logged in, and sufficient recruitment data is available.	
Test Execution Steps		
Step	Action	Expected Response
1	Navigate to the recruitment trends section.	System displays visual trends (graphs, charts).
2	Examine a specific trend (e.g., monthly hires).	Graph shows the correct trend based on past recruitment data.
3	Interact with the trend graph to view specifics.	System displays detailed data for selected points on the graph.
Post-Conditions	Visual trends correctly reflect the historical data of recruitment activities.	
Associated Requirements	RUT-9	
Test Type: Unit	Context: Data Consistency and Refresh	Test ID: UT-43
Description	Ensure that the dashboard reflects consistent and up-to-date recruitment data.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-32">https://jack7aylor.atlassian.net/browse/KAN-32</a>	
Preconditions	HR user is logged in and dashboard has previously displayed data.	
Test Execution Steps		
Step	Action	Expected Response
1	Observe the displayed recruitment data.	Data is consistent with AWS DynamoDB records.
2	Trigger a manual refresh of the dashboard.	Dashboard data updates and reflects the latest input.
3	Check for auto-refresh feature.	Dashboard automatically updates at scheduled intervals.
Post-Conditions	Dashboard data is refreshed and consistent with database records.	
Associated Requirements	RUT-9	



Test Type: Unit	Context: Recruitment Summary Filtering	Test ID: UT-44
Description	Test the dashboard's filtering capabilities to view recruitment summaries based on various criteria.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-32">https://jack7aylor.atlassian.net/browse/KAN-32</a>	
Preconditions	HR user is logged in and multiple recruitment summaries are available.	
Test Execution Steps		
Step	Action	Expected Response
1	Access filtering options on the dashboard.	System presents various criteria for filtering (e.g., department, time period).
2	Apply a specific filter (e.g., last quarter).	Dashboard updates to display summaries for the last quarter.
3	Remove filters to view all data.	Dashboard resets to show all available recruitment summaries.
Post-Conditions	Filtering provides accurate and relevant summaries based on selected criteria.	
Associated Requirements	RUT-9	

Test Type: Unit	Context: Recruitment Status Notifications	Test ID: UT-45
Description	Verify that the dashboard can send real-time notifications to HR users about critical recruitment status updates.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-32">https://jack7aylor.atlassian.net/browse/KAN-32</a>	
Preconditions	HR user is logged in and notification settings are configured.	
Test Execution Steps		
Step	Action	Expected Response
1	Set up notification preferences on the dashboard.	System confirms settings have been saved.
2	Simulate a change that meets notification criteria (e.g., a high-priority position filled).	System sends a notification to the HR user.
3	Check notification reception and content accuracy.	HR user receives a notification with accurate details about the change.
Post-Conditions	HR user receives notifications reflecting real-time recruitment changes.	
Associated Requirements	RUT-15, RUT-9	

Test Type: System	Context: Categorization Feedback Mechanism	Test ID: UT-46
Description	Test the feedback mechanism for HR users to refine and correct the categorization process.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-33">https://jack7aylor.atlassian.net/browse/KAN-33</a>	
Preconditions	HR user is logged in and has permissions to modify categorization rules.	
Test Execution Steps		
Step	Action	Expected Response
1	Identify a misclassified resume.	System shows the resume in the incorrect category.
2	Provide feedback to correct the category.	System accepts feedback and updates the categorization logic.
3	Re-upload the same resume.	System re-processes the resume and categorizes correctly.
Post-Conditions	System updates categorization logic based on user feedback and improves accuracy.	
Associated Requirements	RUT-10	

Test Type: System	Context: Categorize resumes into predefined job categories.	Test ID: UT-47
Description	As an HR staff member, I want to categorize resumes into predefined job categories based on the extracted skills and experiences using natural language processing so that categorizing resumes can be simplified.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-33">https://jack7aylor.atlassian.net/browse/KAN-33</a>	
Preconditions	<ul style="list-style-type: none"><li>• The user must be logged in as an HR staff member with permission to access job criteria settings.</li><li>• The system interface for defining job criteria must be operational.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Upload a resume with specific skills and experiences.	The system processes the resume and extracts the relevant skills and experiences.
2	Ensure predefined job categories are available in the system.	The system lists all predefined job categories.
3	Trigger the categorization process.	The system categorizes the resume into the appropriate job category.
4	Verify the categorized resume.	The resume appears in the correct job category based on extracted skills and experiences.
Post-Conditions	<ul style="list-style-type: none"><li>• Job-specific criteria are stored in the system’s database.</li><li>• Stored criteria are retrievable and accurately reflect the user inputs.</li></ul>	
Associated Requirements	<ul style="list-style-type: none"><li>• RUT-10 The system shall automatically categorize resumes into predefined job categories based on the extracted skills and experiences using natural language processing.</li></ul>	

Test Type: Unit	Context: Job Category Accuracy	Test ID: UT-48
Description	Test the accuracy of the system in categorizing resumes into the correct job categories based on extracted skills and experiences.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-33">https://jack7aylor.atlassian.net/browse/KAN-33</a>	
Preconditions	HR user is logged in and predefined job categories are set up.	
Test Execution Steps		
Step	Action	Expected Response
1	Upload a resume known for specific skills.	System processes and extracts skills and experiences.
2	Check the category assigned by the system.	The system assigns the resume to the appropriate category.
3	Verify correctness by comparing with manual assessment.	Assigned category matches the manual categorization.
Post-Conditions	Resumes are correctly categorized, and the categories reflect the resume content accurately.	
Associated Requirements	RUT-10	

Test Type: Unit	Context: Multi-category Classification	Test ID: UT-49
Description	Verify the system's capability to classify resumes into multiple categories if applicable.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-33">https://jack7aylor.atlassian.net/browse/KAN-33</a>	
Preconditions	HR user is logged in, and resumes with diverse skills that span multiple categories are available.	
Test Execution Steps		
Step	Action	Expected Response
1	Upload a resume with diverse skills.	System processes and extracts diverse skills.
2	Trigger categorization.	System categorizes the resume into multiple categories.
3	Review categorized resume in each category.	Resume appears correctly in each assigned category.
Post-Conditions	Resumes are correctly classified into all relevant categories.	
Associated Requirements	RUT-10	

Test Type: Unit	Context: Category Update and Resume Reclassification	Test ID: UT-50
Description	Ensure the system can update job categories and reclassify resumes accordingly.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-33">https://jack7aylor.atlassian.net/browse/KAN-33</a>	
Preconditions	HR user is logged in and previous job categories and resume classifications exist.	
Test Execution Steps		
Step	Action	Expected Response
1	Modify an existing job category.	System updates the category and confirms the change.
2	Trigger reclassification of existing resumes.	System processes existing resumes for new categorization.
3	Verify updated classifications.	Resumes appear under updated categories correctly.
Post-Conditions	Updated categories are reflected and resumes are reclassified appropriately.	
Associated Requirements	RUT-10	

Test Type: System	Context: Resume Match Percentage	Test ID: UT-51
Description	As an HR staff member, I want to generate a match percentage for each resume so that it is simple to see how closely the resume meets the predefined job-specific criteria.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-34">https://jack7aylor.atlassian.net/browse/KAN-34</a>	
Preconditions	<ul style="list-style-type: none"><li>Job-specific criteria must be defined and saved within the system.</li><li>Resumes ready for testing are available for upload.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Upload a resume with defined job-specific criteria.	The system processes the resume and extracts relevant information.
2	Ensure job-specific criteria are available in the system.	The system lists all predefined job-specific criteria.
3	Trigger the match percentage calculation process.	The system calculates and displays the match percentage for the resume.
4	Verify the accuracy of the match percentage.	The match percentage accurately reflects how closely the resume meets the criteria.
Post-Conditions	<ul style="list-style-type: none"><li>Resumes are ranked according to the predefined criteria.</li><li>The ranked list of resumes is available for review and further action.</li></ul>	
Associated Requirements	<ul style="list-style-type: none"><li>RUT-11 The system shall generate a match percentage for each resume, indicating how closely the resume meets the job-specific criteria set by the HR user.</li></ul>	

Test Type: Unit	Context: Match Percentage with Varying Criteria	Test ID: UT-52
Description	Test the system's ability to generate accurate match percentages as job-specific criteria are varied.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-34">https://jack7aylor.atlassian.net/browse/KAN-34</a>	
Preconditions	HR user is logged in, job-specific criteria are editable, and multiple resumes are available for testing.	
Test Execution Steps		
Step	Action	Expected Response
1	Adjust job-specific criteria (e.g., increase required years of experience).	System updates criteria.
2	Re-calculate match percentages for a set of resumes.	System shows new match percentages reflecting updated criteria.
3	Verify accuracy of new match percentages.	New percentages correctly indicate how well resumes meet the updated criteria.
Post-Conditions	Match percentages accurately reflect adjustments in job-specific criteria.	
Associated Requirements	RUT-11	

Test Type: Unit	Context: Batch Processing of Resumes	Test ID: UT-53
Description	Ensure the system can handle batch processing of resumes and generate match percentages simultaneously.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-34">https://jack7aylor.atlassian.net/browse/KAN-34</a>	
Preconditions	HR user is logged in, multiple resumes are uploaded, and job-specific criteria are established.	
Test Execution Steps		
Step	Action	Expected Response
1	Upload a batch of resumes.	System accepts and processes the batch of resumes.
2	Trigger match percentage calculation for the batch.	System calculates match percentages for all resumes.
3	Review match percentages for accuracy.	All resumes display a match percentage accurately reflecting the criteria.
Post-Conditions	All processed resumes have a calculated match percentage.	
Associated Requirements	RUT-11	

Test Type: Unit	Context: Match Percentage Adjustment by HR	Test ID: UT-54
Description	Test the HR user's capability to manually adjust the match percentage calculation based on additional insights.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-34">https://jack7aylor.atlassian.net/browse/KAN-34</a>	
Preconditions	HR user is logged in, resumes have been processed, and initial match percentages are available.	
Test Execution Steps		
Step	Action	Expected Response
1	Select a resume with an initial match percentage.	System displays the initial match percentage.
2	Manually adjust the match percentage based on additional qualitative analysis.	System allows adjustment and updates the match percentage.
3	Save the new match percentage.	System confirms the new percentage and updates the record.
Post-Conditions	Adjusted match percentages are saved and reflect HR user modifications	
Associated Requirements	RUT-11, RUT-12 (manual override)	

Test Type: Unit	Context: Dynamic Criteria Updates	Test ID: UT-55
Description	Verify the system's responsiveness to dynamic updates in job-specific criteria and its impact on real-time match percentage calculations.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-34">https://jack7aylor.atlassian.net/browse/KAN-34</a>	
Preconditions	HR user is logged in, job-specific criteria are dynamic, and resumes are continuously being uploaded.	
Test Execution Steps		
Step	Action	Expected Response
1	Update job-specific criteria during active resume processing.	System acknowledges criteria updates.
2	Observe the effect on ongoing match percentage calculations.	System recalculates match percentages in real-time.
3	Verify the updated match percentages for consistency with new criteria.	Updated match percentages accurately reflect the new criteria.
Post-Conditions	Real-time updates to match percentages reflect changes in job criteria.	
Associated Requirements	RUT-11	



Test Type: System	Context: Display System- Generated Match Percentage	Test ID: UT-56
Description	As an HR staff member, I want to override system generated resume match percentage so that exceptional circumstances (e.g., candidate has skills not previously identified) can be accommodated.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-35">https://jack7aylor.atlassian.net/browse/KAN-35</a>	
Preconditions	<ul style="list-style-type: none"><li>• The HR staff member is logged into the HR system.</li><li>• The system contains resumes with system-generated match percentages.</li><li>• The HR staff member has the necessary permissions to override match percentages.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Access a resume with a system-generated match percentage.	The system displays the match percentage for the resume.
2	Enter a new match percentage manually.	The system accepts the new match percentage input.
3	Save the overridden match percentage.	The system updates and saves the new match percentage.
Post-Conditions	<ul style="list-style-type: none"><li>• The overridden match percentage is saved and displayed correctly.</li><li>• The system correctly uses the overridden match percentage for subsequent actions.</li></ul>	
Associated Requirements	RUT-12	

Test Type: System	Context: Modify Candidate Skill Set	Test ID: UT-57
Description	As an HR staff member, I want to add additional skills to a candidate's profile so that the system-generated match percentage accurately reflects all relevant skills.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-35">https://jack7aylor.atlassian.net/browse/KAN-35</a>	
Preconditions	<ul style="list-style-type: none"><li>• The HR staff member is logged into the HR system.</li><li>• The system contains candidate profiles with system-generated match percentages.</li><li>• The HR staff member has the necessary permissions to modify candidate profiles.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Access a candidate's profile.	The system displays the current skills and match percentage for the candidate.
2	Add new skills to the candidate's profile.	The system accepts the new skills input.
3	Save the updated candidate profile.	The system updates and recalculates the match percentage based on the new skills.
Post-Conditions	<ul style="list-style-type: none"><li>• The updated skills and match percentage are saved and displayed correctly.</li><li>• The system uses the updated match percentage for subsequent actions.</li></ul>	
Associated Requirements	RUT-12	

Test Type: System	Context: Override System Flagged Candidate	Test ID: UT-58
Description	As an HR staff member, I want to manually flag a candidate as a top match despite a low system-generated match percentage so that the candidate is considered for roles based on exceptional qualifications.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-35">https://jack7aylor.atlassian.net/browse/KAN-35</a>	
Preconditions	<ul style="list-style-type: none"><li>• The HR staff member is logged into the HR system.</li><li>• The system contains candidate profiles with system-generated match percentages.</li><li>• The HR staff member has the necessary permissions to flag candidates.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Access a candidate's profile with a low match percentage.	The system displays the match percentage for the candidate.
2	Manually flag the candidate as a top match.	The system accepts the flagging action.
3	Save the flagged candidate profile.	The system updates the candidate status to top match.
Post-Conditions	<ul style="list-style-type: none"><li>• The candidate's flagged status is saved and displayed correctly.</li><li>• The system uses the flagged status for subsequent candidate recommendations.</li></ul>	
Associated Requirements	RUT-12	

Test Type: System	Context: Update Candidate Experience	Test ID: UT-59
Description	As an HR staff member, I want to update a candidate's experience details so that the system-generated match percentage reflects the most recent and relevant experience.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-35">https://jack7aylor.atlassian.net/browse/KAN-35</a>	
Preconditions	<ul style="list-style-type: none"><li>• The HR staff member is logged into the HR system.</li><li>• The system contains candidate profiles with system-generated match percentages.</li><li>• The HR staff member has the necessary permissions to update candidate profiles.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Access a candidate's profile.	The system displays the current experience and match percentage for the candidate.
2	Update the candidate's experience details.	The system accepts the new experience details.
3	Save the updated candidate profile.	The system updates and recalculates the match percentage based on the new experience.
Post-Conditions	<ul style="list-style-type: none"><li>• The updated experience and match percentage are saved and displayed correctly.</li><li>• The system uses the updated match percentage for subsequent actions.</li></ul>	
Associated Requirements	RUT-12	

Test Type: System	Context: Add Candidate Recommendations	Test ID: UT-60
Description	As an HR staff member, I want to add recommendations to a candidate's profile so that the system-generated match percentage takes these endorsements into account.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-35">https://jack7aylor.atlassian.net/browse/KAN-35</a>	
Preconditions	<ul style="list-style-type: none"><li>• The HR staff member is logged into the HR system.</li><li>• The system contains candidate profiles with system-generated match percentages.</li><li>• The HR staff member has the necessary permissions to add recommendations.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Access a candidate's profile.	The system displays the current match percentage for the candidate.
2	Add new recommendations to the candidate's profile.	The system accepts the new recommendations input.
3	Save the updated candidate profile.	The system updates and recalculates the match percentage based on the new recommendations.
Post-Conditions	<ul style="list-style-type: none"><li>• The updated recommendations and match percentage are saved and displayed correctly.</li><li>• The system uses the updated match percentage for subsequent actions.</li></ul>	
Associated Requirements	RUT-12	

Test Type: System	Context: Add Notes/Tags	Test ID: UT-61
Description	As an HR staff member, I want to add notes or tags to individual resumes so that I can include additional notes (e.g., referred by employee, top candidate...).	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-36">https://jack7aylor.atlassian.net/browse/KAN-36</a>	
Preconditions	<ul style="list-style-type: none"><li>• The HR staff member is logged into the HR system.</li><li>• The system contains resumes accessible to HR staff members.</li><li>• The HR staff member has the necessary permissions to add notes or tags to resumes.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Access a resume from the system.	The system displays the resume details.
2	Enter and save a note or tag for the resume.	The system accepts the note or tag input, updates, and saves it. A confirmation message is displayed.
3	Verify that the system correctly updates and displays the note or tag with the resume details.	The note or tag is displayed correctly with the resume details and used in subsequent system actions.
Post-Conditions	<ul style="list-style-type: none"><li>• The note or tag is saved and displayed correctly.</li><li>• The system correctly uses and displays the note or tag with the resume details.</li></ul>	
Associated Requirements	RUT-13	

Test Type: System	Context: Highlight Key Experience	Test ID: UT-62
Description	As an HR staff member, I want to highlight key experience on a resume so that it is prominently displayed for consideration in the hiring process	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-36">https://jack7aylor.atlassian.net/browse/KAN-36</a>	
Preconditions	<ul style="list-style-type: none"><li>• The HR staff member is logged into the HR system.</li><li>• The system contains resumes accessible to HR staff members.</li><li>• The HR staff member has the necessary permissions to highlight key experiences.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Access a resume from the system.	The system displays the resume details.
2	Highlight key experience on the resume.	The system accepts the highlighted experience and saves it.
3	Verify that the system correctly displays the highlighted experience.	The key experience is prominently displayed on the resume.
Post-Conditions	<ul style="list-style-type: none"><li>• The highlighted experience is saved and displayed correctly.</li><li>• The system correctly uses and displays the highlighted experience in subsequent actions.</li></ul>	
Associated Requirements	RUT-13	

Test Type: System	Context: Add Custom Rating	Test ID: UT-63
Description	As an HR staff member, I want to add a custom rating to a resume so that I can provide an overall evaluation score based on my assessment.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-36">https://jack7aylor.atlassian.net/browse/KAN-36</a>	
Preconditions	<ul style="list-style-type: none"><li>• The HR staff member is logged into the HR system.</li><li>• The system contains resumes accessible to HR staff members.</li><li>• The HR staff member has the necessary permissions to add custom ratings.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Access a resume from the system.	The system displays the resume details.
2	Enter and save a custom rating for the resume.	The system accepts the custom rating input, updates, and saves it. A confirmation message is displayed.
3	Verify that the system correctly updates and displays the custom rating with the resume details.	The custom rating is displayed correctly with the resume details and used in subsequent system actions.
Post-Conditions	<ul style="list-style-type: none"><li>• The custom rating is saved and displayed correctly.</li><li>• The system correctly uses and displays the custom rating with the resume details.</li></ul>	
Associated Requirements	RUT-13	



Test Type: System	Context: Add Interview Feedback	Test ID: UT-64
Description	As an HR staff member, I want to add interview feedback to a candidate's profile so that my observations and recommendations are included in the hiring process.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-36">https://jack7aylor.atlassian.net/browse/KAN-36</a>	
Preconditions	<ul style="list-style-type: none"><li>• The HR staff member is logged into the HR system.</li><li>• The system contains candidate profiles accessible to HR staff members.</li><li>• The HR staff member has the necessary permissions to add interview feedback.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Access a candidate's profile.	The system displays the candidate's profile details.
2	Enter and save interview feedback for the candidate.	The system accepts the feedback input, updates, and saves it. A confirmation message is displayed.
3	Verify that the system correctly updates and displays the interview feedback with the candidate's profile.	The feedback is displayed correctly with the candidate's profile and used in subsequent system actions.
Post-Conditions	<ul style="list-style-type: none"><li>• The interview feedback is saved and displayed correctly.</li><li>• The system correctly uses and displays the feedback with the candidate's profile.</li></ul>	
Associated Requirements	RUT-13	

Test Type: System	Context: Update Candidate Availability	Test ID: UT-65
Description	As an HR staff member, I want to update a candidate's availability status so that the system reflects their current availability for interviews and hiring.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-36">https://jack7aylor.atlassian.net/browse/KAN-36</a>	
Preconditions	<ul style="list-style-type: none"><li>• The HR staff member is logged into the HR system.</li><li>• The system contains candidate profiles accessible to HR staff members.</li><li>• The HR staff member has the necessary permissions to update candidate availability.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Access a candidate's profile.	The system displays the candidate's profile details.
2	Update the candidate's availability status.	The system accepts the new availability status input, updates, and saves it. A confirmation message is displayed.
3	Verify that the system correctly updates and displays the candidate's availability status.	The updated availability status is displayed correctly with the candidate's profile and used in subsequent system actions.
Post-Conditions	<ul style="list-style-type: none"><li>• The updated availability status is saved and displayed correctly.</li><li>• The system correctly uses and displays the updated status with the candidate's profile.</li></ul>	
Associated Requirements	RUT-13	

Test Type: System	Context: Send Email	Test ID: UT-66
Description	As an HR staff member, I want to schedule and send automated email communications to candidates so that I can ensure notifications are sent and allow time to concentrate on other efforts (e.g., rejection letter).	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-37">https://jack7aylor.atlassian.net/browse/KAN-37</a>	
Preconditions	<ul style="list-style-type: none"><li>• The HR staff member is logged into the HR system.</li><li>• The HR staff member has the necessary permissions to schedule and send emails.</li><li>• The system has the capability to send emails.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Access the email communication scheduling page.	The system displays the email scheduling interface.
2	Enter the details for an email, including the recipient, subject, body, and scheduling information.	The system accepts the email details and scheduling inputs.
3	Save the scheduled email.	The system saves the scheduling information and confirms the operation. A confirmation message is displayed.
4	Verify that the email is sent at the scheduled time and that the status is updated appropriately.	The system sends the email at the specified time, updates the status to "Sent," and the recipient receives the email.
Post-Conditions	<ul style="list-style-type: none"><li>• The scheduled email is sent at the specified time.</li><li>• The system updates the status to reflect that the email has been sent.</li></ul>	
Associated Requirements	RUT-14	

Test Type: System	Context: Schedule Interview	Test ID: UT-67
Description	As an HR staff member, I want to schedule interviews with candidates and send automated email invitations so that candidates are informed about interview details.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-37">https://jack7aylor.atlassian.net/browse/KAN-37</a>	
Preconditions	<ul style="list-style-type: none"><li>• The HR staff member is logged into the HR system.</li><li>• The HR staff member has the necessary permissions to schedule interviews and send emails.</li><li>• The system has the capability to send emails.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Access the interview scheduling page.	The system displays the interview scheduling interface.
2	Enter the details for the interview, including the candidate, date, time, and location.	The system accepts the interview details.
3	Save the scheduled interview and send the email invitation.	The system saves the interview details, schedules the interview, and sends the email invitation. A confirmation message is displayed.
4	Verify that the email invitation is sent and that the status is updated appropriately.	The system sends the email invitation, updates the status to "Sent," and the candidate receives the email.
Post-Conditions	<ul style="list-style-type: none"><li>• The email invitation is sent at the specified time.</li><li>• The system updates the status to reflect that the email has been sent.</li></ul>	
Associated Requirements	RUT-14	

Test Type: System	Context: Follow-Up Email	Test ID: UT-68
Description	As an HR staff member, I want to send follow-up emails to candidates after interviews so that they are kept informed about the next steps in the hiring process.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-37">https://jack7aylor.atlassian.net/browse/KAN-37</a>	
Preconditions	<ul style="list-style-type: none"><li>• The HR staff member is logged into the HR system.</li><li>• The HR staff member has the necessary permissions to send follow-up emails.</li><li>• The system has the capability to send emails.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Access the candidate's profile.	The system displays the candidate's profile details.
2	Enter the details for the follow-up email, including the recipient, subject, and body.	The system accepts the email details.
3	Send the follow-up email.	The system sends the follow-up email and confirms the operation. A confirmation message is displayed.
4	Verify that the follow-up email is sent and that the status is updated appropriately.	The system sends the email, updates the status to "Sent," and the candidate receives the email.
Post-Conditions	<ul style="list-style-type: none"><li>• The follow-up email is sent successfully.</li><li>• The system updates the status to reflect that the email has been sent.</li></ul>	
Associated Requirements	RUT-14	

Test Type: System	Context: Offer Letter Email	Test ID: UT-69
Description	As an HR staff member, I want to send offer letters to selected candidates via email so that they receive their job offers promptly and can respond accordingly.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-37">https://jack7aylor.atlassian.net/browse/KAN-37</a>	
Preconditions	<ul style="list-style-type: none"><li>• The HR staff member is logged into the HR system.</li><li>• The HR staff member has the necessary permissions to send offer letters.</li><li>• The system has the capability to send emails.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Access the candidate's profile.	The system displays the candidate's profile details.
2	Enter the details for the offer letter email, including the recipient, subject, and body.	The system accepts the email details.
3	Send the offer letter email.	The system sends the offer letter email and confirms the operation. A confirmation message is displayed.
4	Verify that the offer letter email is sent and that the status is updated appropriately.	The system sends the email, updates the status to "Sent," and the candidate receives the email.
Post-Conditions	<ul style="list-style-type: none"><li>• The offer letter email is sent successfully.</li><li>• The system updates the status to reflect that the email has been sent.</li></ul>	
Associated Requirements	RUT-14	

Test Type:	Context: Rejection Letter Email	Test ID: UT-70
Description	As an HR staff member, I want to send rejection letters to unsuccessful candidates via email so that they are promptly informed about the decision and can seek other opportunities.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-37">https://jack7aylor.atlassian.net/browse/KAN-37</a>	
Preconditions	<ul style="list-style-type: none"><li>• The HR staff member is logged into the HR system.</li><li>• The HR staff member has the necessary permissions to send rejection letters.</li><li>• The system has the capability to send emails.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Access the candidate's profile.	The system displays the candidate's profile details.
2	Enter the details for the rejection letter email, including the recipient, subject, and body.	The system accepts the email details.
3	Send the rejection letter email.	The system sends the rejection letter email and confirms the operation. A confirmation message is displayed.
4	Verify that the rejection letter email is sent and that the status is updated appropriately.	The system sends the email, updates the status to "Sent," and the candidate receives the email.
Post-Conditions	<ul style="list-style-type: none"><li>• The rejection letter email is sent successfully.</li><li>• The system updates the status to reflect that the email has been sent.</li></ul>	
Associated Requirements	RUT-14	

Test Type: System	Context: High-Priority Resume Matches	Test ID: UT-59
Description	As an HR staff member, I want to receive real-time notifications when a resume matches high-priority job criteria so that application can be processed immediately.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-38">https://jack7aylor.atlassian.net/browse/KAN-38</a>	
Preconditions	<ul style="list-style-type: none"><li>• The HR staff member is logged into the HR system.</li><li>• High-priority job criteria are defined and configured in the system.</li><li>• The HR staff member has the necessary permissions to receive notifications.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Upload a resume that matches high-priority criteria.	The system processes the resume and evaluates it against high-priority criteria.
2	Ensure high-priority criteria are defined in the system.	The system lists all high-priority criteria and confirms their presence.
3	Trigger the notification process.	The system sends a real-time notification when a match is found.
4	Verify the real-time notification.	The real-time notification is received and displayed to the HR user.
Post-Conditions	<ul style="list-style-type: none"><li>• A real-time notification is sent and received by the HR staff member when a resume matches high-priority criteria.</li><li>• The system updates the status of the notification appropriately.</li></ul>	
Associated Requirements	RUT-15	



Test Type: System	Context: High-Priority Candidate Alert	Test ID: UT-71
Description	As an HR staff member, I want to receive real-time alerts for high-priority candidates so that immediate action can be taken to engage with top talent.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-38">https://jack7aylor.atlassian.net/browse/KAN-38</a>	
Preconditions	<ul style="list-style-type: none"><li>• The HR staff member is logged into the HR system.</li><li>• High-priority candidate criteria are defined and configured in the system.</li><li>• The HR staff member has the necessary permissions to receive alerts.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Access the candidate alert configuration page.	The system displays the alert configuration interface.
2	Define high-priority candidate criteria.	The system accepts the criteria and saves the configuration.
3	Upload a resume that matches the defined criteria.	The system processes the resume and evaluates it against the criteria.
4	Verify the real-time alert.	The real-time alert is received and displayed to the HR user.
Post-Conditions	<ul style="list-style-type: none"><li>• A real-time alert is sent and received by the HR staff member when a resume matches high-priority candidate criteria.</li><li>• The system updates the status of the alert appropriately.</li></ul>	
Associated Requirements	RUT-15	

Test Type: System	Context: Critical Skill Match Notification	Test ID: UT-72
Description	As an HR staff member, I want to receive notifications when resumes with critical skills are uploaded so that these candidates can be prioritized in the hiring process.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-38">https://jack7aylor.atlassian.net/browse/KAN-38</a>	
Preconditions	<ul style="list-style-type: none"><li>• The HR staff member is logged into the HR system.</li><li>• Critical skills criteria are defined and configured in the system.</li><li>• The HR staff member has the necessary permissions to receive notifications.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Access the critical skills configuration page.	The system displays the critical skills configuration interface.
2	Define critical skills criteria.	The system accepts the criteria and saves the configuration.
3	Upload a resume that matches the defined criteria.	The system processes the resume and evaluates it against the criteria.
4	Verify the notification for critical skill match.	The notification is received and displayed to the HR user.
Post-Conditions	<ul style="list-style-type: none"><li>• A notification is sent and received by the HR staff member when a resume matches critical skills criteria.</li><li>• The system updates the status of the notification appropriately.</li></ul>	
Associated Requirements	RUT-15	

Test Type:	Context: Immediate Candidate Referral Alert	Test ID: UT-73
Description	As an HR staff member, I want to receive immediate alerts when a candidate is referred by a high-priority source so that these candidates are given prompt attention.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-38">https://jack7aylor.atlassian.net/browse/KAN-38</a>	
Preconditions	<ul style="list-style-type: none"><li>• The HR staff member is logged into the HR system.</li><li>• High-priority referral sources are defined and configured in the system.</li><li>• The HR staff member has the necessary permissions to receive alerts.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Access the referral alert configuration page.	The system displays the referral alert configuration interface.
2	Define high-priority referral sources.	The system accepts the criteria and saves the configuration.
3	Upload a referred candidate's resume.	The system processes the resume and evaluates it against the referral source criteria.
4	Verify the immediate referral alert.	The referral alert is received and displayed to the HR user.
Post-Conditions	<ul style="list-style-type: none"><li>• An immediate referral alert is sent and received by the HR staff member when a resume matches high-priority referral criteria.</li><li>• The system updates the status of the alert appropriately.</li></ul>	
Associated Requirements	RUT-15	

Test Type: System	Context: Priority Position Match Notification	Test ID: UT-74
Description	As an HR staff member, I want to receive notifications when resumes match criteria for high-priority positions so that these candidates can be quickly moved through the hiring process.	
Jira Epic	<a href="https://jack7aylor.atlassian.net/browse/KAN-38">https://jack7aylor.atlassian.net/browse/KAN-38</a>	
Preconditions	<ul style="list-style-type: none"><li>• The HR staff member is logged into the HR system.</li><li>• High-priority position criteria are defined and configured in the system.</li><li>• The HR staff member has the necessary permissions to receive notifications.</li></ul>	
Test Execution Steps		
Step	Action	Expected Response
1	Access the high-priority position configuration page.	The system displays the high-priority position configuration interface.
2	Define high-priority position criteria.	The system accepts the criteria and saves the configuration.
3	Upload a resume that matches the defined criteria.	The system processes the resume and evaluates it against the criteria.
4	Verify the high-priority position match notification.	The notification is received and displayed to the HR user.
Post-Conditions	<ul style="list-style-type: none"><li>• A notification is sent and received by the HR staff member when a resume matches high-priority position criteria.</li><li>• The system updates the status of the notification appropriately.</li></ul>	
Associated Requirements	RUT-14	

## 5. Test Results

### 5.1 Test Results Overview

The unit test scripts are primarily written using Jest for testing JavaScript and React components, as well as Pytest for testing Python scripts involved in resume parsing and data extraction. These scripts are designed to validate individual components in isolation, ensuring that each function and module behaves as expected under various conditions. The tests cover a range of scenarios, including edge cases, to ensure robustness.

### 5.2 Coverage Analysis

Using Istanbul and Jest, the team initially faced challenges in achieving accurate code coverage, with some files showing incomplete coverage. Over time, targeted test cases improved coverage, addressing earlier gaps. Despite these improvements, the shift to demo preparation in the final sprint resulted in partial or incomplete testing of some new features.

### 5.3 Bug/Fault/Failure Report

During system testing, issues were found with session management, particularly in maintaining session continuity across multiple tabs.

Early in testing, the login feature exhibited inconsistencies with handling repeated failed login attempts, which required additional refinement.

The analytics dashboard sometimes failed to display real-time updates accurately, especially under high data load conditions.

#### Explanation of How These Issues Were Addressed:

- **Session Continuity:** This issue was addressed by refining session management logic, ensuring that user sessions were maintained consistently across all open tabs and devices.
- **Login Functionality:** Additional tests were implemented to specifically target edge cases in the login process, including handling for repeated failed login attempts. The logic was adjusted to improve reliability and security.
- **Dashboard Data Inconsistencies:** The team focused on optimizing data retrieval and display functions, particularly under high load scenarios. This involved refining database queries and ensuring that the front-end accurately reflected real-time changes in the underlying data.

These efforts contributed to the overall stability and reliability of the system, though some issues remain partially addressed, particularly in areas related to newer features that were deprioritized in the final sprint.

## 6. Traceability Matrix

### 6.1 Traceability Matrix Overview

The traceability matrix is a critical tool in software testing that ensures every requirement is covered by one or more test cases. Its primary purpose is to provide a clear and organized mapping between software requirements, user stories, architectural elements, and corresponding test cases. This mapping allows for a systematic verification process, ensuring that all specified functionalities are tested and that any gaps in testing coverage are identified and addressed. By linking each test case to its related requirement, the traceability matrix also facilitates easier tracking of testing progress and validation of the system's compliance with its intended design and requirements.

### 6.2 Traceability Matrixes

#### Sprint 1

Use Case	Func. Req.	User Story	Arch. Element	Commit	Test ID	Test Status	Comment
KAN-23	RUT-1	Resume Upload	User Interface	<a href="#">LINK</a>	UT-01	PASSED	Verifies successful upload in different formats.
KAN-23	RUT-1, RUT-3	Resume Upload	File Upload Handling	<a href="#">LINK</a>	UT-02	PARTIALLY PASSED	Ensures proper handling of large files.
KAN-23	RUT-2, RUT-3	Resume Upload	Data Extraction and Storage	<a href="#">LINK</a>	UT-03	PASSED	Tests data parsing across multiple formats.
KAN-23	RUT-1	Resume Upload	User Interface	<a href="#">LINK</a>	UT-04	NOT DOING	Checks system capability for handling multiple uploads.
KAN-23	RUT-1	Resume Upload	User Interface	<a href="#">LINK</a>	UT-05	PASSED	Verifies user feedback during upload.

KAN-29	RUT-6	User Accounts	Account Management		UT-06	NOT TESTED	Tests account creation functionality.
KAN-29	RUT-7	User Accounts	Authentication	<a href="#">LINK</a>	UT-07	PASSED	Ensures login functionality.
KAN-29	RUT-6, RUT-7	User Accounts	Account Management		UT-08	NOT DOING	Verifies password reset process.
KAN-29	RUT-6, RUT-7	User Accounts	Security	<a href="#">LINK</a>	UT-09	PARTIALLY PASSED	Tests system response to repeated failed login attempts.
KAN-29	RUT-6, RUT-7	User Accounts	Account Management		UT-10	NOT TESTED	Verifies functionality for deleting accounts.
KAN-30	RUT-7	Application Login	Session Management		UT-11	PARTIALLY PASSED	Ensures secure logout and re-login procedures.
KAN-30	RUT-7	Application Login	Session Management	<a href="#">LINK</a>	UT-12	PARTIALLY PASSED	Checks session continuity across multiple tabs.
KAN-30	RUT-7	Application Login	Authentication	<a href="#">LINK</a>	UT-13	PASSED	Validates case sensitivity of the username field.
KAN-30	RUT-7	Application Login	Security		UT-14	NOT TESTED	Tests redirection to login page for unauthorized access.

KAN-30	RUT-7	Application Login	Authentication		UT-15	NOT DOING	Verifies multi-factor authentication process.
--------	-------	-------------------	----------------	--	-------	-----------	---

## Sprint 2

Use Case	Func. Req.	User Story	Arch. Element	Commit	Test ID	Test Status	Comment
KAN-25	RUT-1, 2, 3	Extraction	Extraction	<a href="#">LINK</a>	UT-16	PASSED	Upload Resume with Rich Formatting
KAN-25	RUT-2	Extraction	Extraction	<a href="#">LINK</a>	UT-17	PARTIALLY PASSED	Upload Empty Resume, feedback needed
KAN-25	RUT-1, 2	Extraction	Extraction	<a href="#">LINK</a>	UT-18	PASSED	Upload of Incorrectly Formatted PDF
KAN-25	RUT-1, 2, 3	Extraction	Extraction	<a href="#">LINK</a>	UT-19	PARTIALLY PASSED	Resume Update and Re-extraction
KAN-25	RUT-1, 2, 3	Performance	Performance	<a href="#">LINK</a>	UT-20	PASSED	Upload of Large Resume File
KAN-26	RUT-3	Storage	Storage	<a href="#">LINK</a>	UT-21	PASSED	Storage of Large Resume Data



KAN-26	RUT-3	Storage	Consistency	<a href="#">LINK</a>	UT-22	PASSED	Data Consistency on Rapid Re-upload
KAN-26	RUT-3	Storage	Encoding	<a href="#">LINK</a>	UT-23	PARTIALLY PASSED	Handling of Special Characters and Encoding
KAN-26	RUT-3	Storage	Update	N/A	UT-24	NOT DOING	Resume Data Update
KAN-26	RUT-3	Storage	Concurrency	<a href="#">LINK</a>	UT-25	PASSED	Concurrent Data Storage
KAN-27	RUT-4	Criteria	Validation	<a href="#">LINK</a>	UT-26	PASSED	Invalid Input Handling
KAN-27	RUT-4	Criteria	Update	<a href="#">LINK</a>	UT-27	PASSED	Update Existing Criteria
KAN-27	RUT-4	Criteria	Access Control	N/A	UT-28	NOT DOING	Role-Based Access Control
KAN-27	RUT-4	Criteria	Deletion	N/A	UT-29	NOT DOING	Criteria Deletion

KAN-27	RUT-4	Criteria	Concurrency	N/A	UT-30	NOT DOING	Simultaneous Criteria Updates
KAN-28	RUT-5	Rank	Incomplete	<a href="#">LINK</a>	UT-31	PASSED	Ranking with Incomplete Criteria
KAN-28	RUT-5	Rank	Stability	N/A	UT-32	NOT DOING	Ranking Stability with Duplicate Criteria Scores
KAN-28	RUT-5	Rank	Edge Cases	<a href="#">LINK</a>	UT-33	PASSED	Edge Cases in Criteria Matching
KAN-28	RUT-5	Rank	Performance	N/A	UT-34	NOT DOING	Ranking Performance Under High Load
KAN-28	RUT-5	Rank	Consistency	N/A	UT-35	NOT DOING	Ranking Repeatability and Consistency

### Sprint 3

Use Case	Func. Req.	User Story	Arch. Element	Commit	Test ID	Test Status	Comment
KAN-31	RUT-8	Analytics Dashboard	User Interface	NULL	UT-36	NOT TESTED	Display a dashboard with history of previous uploaded candidates

KAN-31	RUT-8	Dashboard Data Accuracy	User Interface	NULL	UT-37	PARTIALY PASSED	Verify that the dashboard accurately displays extracted data from resumes
KAN-31	RUT-8	Dashboard Interactivity	User Interface	NULL	UT-38	PARTIALY PASSED	Test interactive elements of the dashboard for dynamic data analysis
KAN-31	RUT-8	Dashboard Real-time Update	User Interface	NULL	UT-39	NOT TESTED	Ensure the dashboard reflects real-time updates as new resumes are processed
KAN-31	RUT-8	Dashboard Summary and Detail Toggle	User Interface	NULL	UT-40	NOT TESTED	Test the dashboard's ability to toggle between summary and detailed analytics
KAN-32	RUT-9	Recruitment Status Dashboard	User Interface, Database Integration	NULL	UT-41	NOT TESTED	Display a second dashboard that shows candidates hiring progress
KAN-32	RUT-9	Recruitment Trends Visualization	User Interface, Database Integration	NULL	UT-42	NOT TESTED	Verify the dashboard's ability to display visual trends of the recruitment process
KAN-32	RUT-9	Data Consistency and Refresh	User Interface, Database Integration	NULL	UT-43	NOT TESTED	Ensure that the dashboard reflects consistent and up-to-date recruitment data
KAN-32	RUT-9	Recruitment Summary Filtering	User Interface, Database Integration	NULL	UT-44	NOT TESTED	Test the dashboard's filtering capabilities to view recruitment summaries

KAN-32	RUT-9	Recruitment Status Notifications	User Interface, Database Integration	NULL	UT-45	NOT TESTED	Verify that the dashboard can send real-time notifications to HR users
KAN-33	RUT-10	Categorize Resumes into Predefined Job Categories	User Interface, Database Integration	NULL	UT-46	PARTIALLY PASSED	Filter resumes so the user can find a candidate in the registry
KAN-33	RUT-10	Categorization Feedback Mechanism	User Interface, Database Integration	NULL	UT-47	NOT TESTED	Test the feedback mechanism for HR users to refine the categorization process
KAN-33	RUT-10	Job Category Accuracy	User Interface, Database Integration	NULL	UT-48	PARTIALLY PASSED	Test the accuracy of the system in categorizing resumes into the correct job categories
KAN-33	RUT-10	Multi-category Classification	User Interface, Database Integration	NULL	UT-49	NOT TESTED	Verify the system's capability to classify resumes into multiple categories
KAN-33	RUT-10	Category Update and Resume Reclassification	User Interface, Database Integration	NULL	UT-50	NOT TESTED	Ensure the system can update job categories and reclassify resumes accordingly
KAN-34	RUT-11	Resume Match Percentage	Business Logic, Data Processing	NULL	UT-51	NOT TESTED	Create the algorithm for candidate-job match percentage
KAN-34	RUT-11	Match Percentage with Varying Criteria	Business Logic, Data Processing	NULL	UT-52	NOT TESTED	Test the system's ability to generate accurate match percentages as job-specific criteria are varied

KAN-34	RUT-11	Batch Processing of Resumes	Business Logic, Data Processing	NULL	UT-53	NOT TESTED	Ensure the system can handle batch processing of resumes and generate match percentages simultaneously
KAN-34	RUT-11	Match Percentage Adjustment by HR	Business Logic, Data Processing	NULL	UT-54	NOT TESTED	Test the HR user's capability to manually adjust the match percentage calculation
KAN-34	RUT-11	Dynamic Criteria Updates	Business Logic, Data Processing	NULL	UT-55	NOT TESTED	Verify the system's responsiveness to dynamic updates in job-specific criteria

#### Sprint 4

Use Case	Func. Req.	User Story	Arch. Element	Commit	Test ID	Test Status	Comment
KAN-35	RUT-12	Override resume match percentage	User Interface, Business Logic	NULL	UT-56	NOT TESTED	Allow HR users to manually override automated rankings
KAN-35	RUT-12	Add additional skills to candidate profile	User Interface, Business Logic	NULL	UT-57	NOT TESTED	Enable HR users to add new skills to a candidate's profile
KAN-35	RUT-12	Manually flag candidate as top match	User Interface, Business Logic	NULL	UT-58	NOT TESTED	Allow HR users to manually flag candidates as top matches

KAN-35	RUT-12	Update candidate's experience details	User Interface, Business Logic	NULL	UT-59	NOT TESTED	Allow HR users to update candidate's experience details
KAN-35	RUT-12	Add recommendations to candidate's profile	User Interface, Business Logic	NULL	UT-60	NOT TESTED	Allow HR users to add recommendations to a candidate's profile
KAN-36	RUT-13	Add notes or tags to resumes	User Interface, Data Storage	NULL	UT-61	NOT TESTED	Enable HR users to add and save notes or tags for resumes
KAN-36	RUT-13	Highlight key experience on resumes	User Interface, Data Storage	NULL	UT-62	NOT TESTED	Enable HR users to highlight key experience on resumes
KAN-36	RUT-13	Add custom rating to resumes	User Interface, Data Storage	NULL	UT-63	NOT TESTED	Enable HR users to add custom ratings to resumes
KAN-36	RUT-13	Add interview feedback to candidate's profile	User Interface, Data Storage	NULL	UT-64	NOT TESTED	Allow HR users to add interview feedback to candidate's profile
KAN-36	RUT-13	Update candidate's availability status	User Interface, Data Storage	NULL	UT-65	NOT TESTED	Allow HR users to update candidate's availability status
KAN-37	RUT-14	Schedule automated email communications	User Interface, Email Service	NULL	UT-66	NOT TESTED	Schedule and send automated emails to candidates

KAN-37	RUT-14	Schedule interviews and send email invitations	User Interface, Email Service	NULL	UT-67	NOT TESTED	Schedule interviews and send email invitations to candidates
KAN-37	RUT-14	Send follow-up emails to candidates	User Interface, Email Service	NULL	UT-68	NOT TESTED	Send follow-up emails to candidates after interviews
KAN-37	RUT-14	Send offer letters via email	User Interface, Email Service	NULL	UT-69	NOT TESTED	Send offer letters to selected candidates via email
KAN-37	RUT-14	Send rejection letters via email	User Interface, Email Service	NULL	UT-70	NOT TESTED	Send rejection letters to unsuccessful candidates via email
KAN-38	RUT-15	Real-time notifications for high-priority resumes	Notification Service, Business Logic	NULL	UT-59	NOT TESTED	Send real-time notifications for high-priority resume matches
KAN-38	RUT-15	Real-time alerts for high-priority candidates	Notification Service, Business Logic	NULL	UT-71	NOT TESTED	Send real-time alerts for high-priority candidates
KAN-38	RUT-15	Notifications for critical skill matches	Notification Service, Business Logic	NULL	UT-72	NOT TESTED	Send notifications for resumes with critical skills
KAN-38	RUT-15	Immediate alerts for high-priority referrals	Notification Service, Business Logic	NULL	UT-73	NOT TESTED	Send immediate alerts for candidates referred by high-priority sources

KAN-38	RUT-15	Notifications for high-priority position matches	Notification Service, Business Logic	NULL	UT-74	NOT TESTED	Send notifications when resumes match high-priority position criteria
--------	--------	--	--------------------------------------	------	-------	------------	---



## 7. Conclusion

### 7.1 Summary of Findings

The testing process for the Automated Resume Screening Tool revealed several important insights into the system's functionality and reliability. Across four sprints, 74 test cases were developed to validate the core functionalities, integration, and performance of the tool. The key findings include:

- The core features, such as resume uploading, data extraction, and account management, were tested, with the majority of test cases passing successfully. This indicates that the system handles essential tasks reliably.
- Testing of advanced features like the analytics dashboard and real-time notifications showed mixed results. While many tests passed, some partially passed or were not yet tested due to the complexity and recent implementation of these features.
- The system demonstrated solid performance in handling large and complex resumes, with test cases confirming its ability to process and store data efficiently. However, some areas, such as session management and multi-factor authentication, require further refinement.
- A few test cases, particularly those involving newer features like match percentage overrides and real-time notifications, have not been fully tested. These areas will need additional focus to ensure full coverage before deployment.

### 7.2 Overall Quality Assessment

Based on the testing results, the overall quality of the Automated Resume Screening Tool is strong, particularly in its core functionalities. The system has demonstrated reliability in processing resumes, managing user accounts, and storing data, all of which are critical for its intended purpose. However, the partially passed and untested areas suggest that there are still some challenges to address, especially in the newly added features. The tool is on track to meet its requirements, but further testing and refinement are necessary to ensure that all features are fully functional and robust.

### 7.3 Recommendations

To ensure the Automated Resume Screening Tool is fully ready for deployment, it's essential to complete the testing of advanced features like match percentage overrides and real-time notifications, address issues in partially passed test cases, and enhance the continuous integration process with automated testing. A final round of user acceptance testing with HR professionals should be conducted to validate usability, and a system for ongoing monitoring and maintenance should be established to support long-term performance and reliability (as they are our targeted audience).