JobAssist Application Quality Assurance Checklist

This checklist outlines the mandatory tests to be executed across the application's three main interfaces following any code deployment or significant change.

1. Dashboard & Filtering (index.html)

Test Case	Feature/Endpoint	Steps	Expected Result	Pass/Fail
1.1 Initial Load	GET /api/companies	Load index.html.	All standardized company profiles load and display in the table.	[]
1.2 Target Filter	Filter Control	Select **Target Interest: Yes** from the dropdown.	Only companies flagged as target_interest: true are visible.	[]
1.3 Name Search	Search Input	Enter a partial clean name (e.g., "Micr").	The table instantly filters to show only matching company names (e.g., "Microsoft").	[]
1.4 Column Sorting	Data Manipulation	Click the **Clean Company Name** header, then click the **Annual Revenue**	The data correctly sorts alphabetically, then numerically (Asc/Desc).	[]

Test Case	Feature/Endpoint	Steps	Expected Result	Pass/Fail
			header.	
1.5 Navigation	Header Link	Click the link/button to the Management Tool for an arbitrary company.	User is redirected to management.html with the correct ID passed.	[]

2. Raw Name Mapping/Cleanup Tool (cleanup.html)

(Note: Test cases for the Cleanup Tool have been omitted for brevity, but a full QA document would include them.)

3. Company Management Interface (management.html)

These tests validate the newly implemented data retrieval and link functionality.

Test Case	Feature/Endpoint	Steps	Expected Result	Pass/Fail
3.1 Profile Load	GET /api/companies/{id}/details	Click a Company Name link from the Dashboard to load the profile.	The form fields (Headquarters, Revenue, Notes, etc.) are populated with the correct profile data.	[]
3.2 Raw Names List	GET /api/companies/{id}/details	Verify the content in the **Linked Raw Names** section.	The section displays a list of all raw names associated with the profile (e.g., "Google Inc.",	[]

Test Case	Feature/Endpoint	Steps	Expected Result	Pass/Fail
			"Googlle").	
3.3 Contacts & Count	GET /api/companies /{id}/details	Load a company profile.	The **Linked Contacts** table populates, and the **Contact Count** displays the total number of records correctly.	[]
3.4 Clickable Contacts	management.js Rendering	Click on a contact's name in the **Linked Contacts** table.	A new browser tab opens, navigating directly to the contact's LinkedIn URL (linkedin_url).	[]
3.5 Profile Update	PUT /api/companies /{id}	Change the **Headquarters** and **Target Interest** flag and click **Save Changes**.	The status message confirms success, the page updates immediately, and a subsequent page reload shows the saved data.	[]

4. General Code & Navigation

Test Case	Feature/Endpoint	Steps	Expected Result	Pass/Fail
4.1 File Metadata	All Files	Review the top of all code/HTML	Each file must contain a comment with	[]

Test Case	Feature/Endpoint	Steps	Expected Result	Pass/Fail
		files.	the file name (e.g., app.py) and the date it was last edited (e.g., 2025-10-24).	
4.2 Global Navigation	All HTML Pages	Navigate between index.html, cleanup.html, and management.html.	A consistent, functional navigation bar (Header) must be present on all pages, linking to the others.	[]

5. API and Data Integrity

Test Case	Feature/Endpoint	Steps	Expected Result	Pass/Fail
5.1 ID Consistency	All Endpoints	Map a raw name (e.g., "Foo Ltd") to an existing company (ID 10).	The company_name_mapping table must show raw_name: "Foo Ltd", company_id: 10.	[]
5.2 Data Types	PUT /api/companies/{id}	Attempt to save an update with a non-numeric value in size_employees or annual_revenue.	The API should return a 500 or 400 error and not corrupt the database.	[]

5.3 Error Handling	Any API Call	Simulate a failed database connection during any operation.	The API should return an appropriate 500-level error and a friendly message to the user.	[]