JobAssist Database Design Guide

1. Purpose and Overview

The JobAssist database serves two primary functions:

1. **Contact Data Standardization:** To normalize raw, inconsistent company data into single, reliable profiles for accurate reporting and enrichment.
2. **Job Application Tracking:** To record specific job application events, link them directly to the standardized company profiles, and manage associated document metadata.

The architecture is split into two logical areas connected by the central companies table.

2. Entity Relationship Diagram (Conceptual)

The design follows a central hub-and-spoke model where the companies table is the hub for all data integrity and reporting.

- **Left Side (Standardization):** contacts $\rightarrow$ company_name_mapping $\rightarrow$ **companies**
- **Right Side (Tracking): companies** $\rightarrow$ applications $\leftarrow$ job_titles / job_documents

3. Schema Details: Standardization Core

These tables manage the import, normalization, and enrichment of company and contact data.

3.1. Table: contacts (Source/Raw Contact Data)

Field Name	Data Type	Description
id	integer	Primary Key (Contact ID).
first_name	varchar(50)	Contact's first name.
last_name	varchar(50)	Contact's last name.
company	varchar(100)	**Raw company name** (FK to company_name_mapping). The string that needs cleaning.
position	varchar(100)	Contact's job title/position.

connected_on	date	Date the contact was made.

3.2. Table: company_name_mapping (Standardization Link)

Field Name	Data Type	Description
raw_name	varchar(100)	**Primary Key / Foreign Key** (referenced by contacts.company). The unique, unstandardized company name string.
company_id	integer	**Foreign Key** (to companies.company_id). The ID of the clean company profile this raw name belongs to.

3.3. Table: companies (Clean, Managed Profiles) - The Central Hub

Field Name	Data Type	Description
company_id	integer	Primary Key (Clean Company ID).
company_name_clean	varchar(100)	The standardized company name (e.g., "Google").
target_interest	boolean	Flag indicating sales/marketing interest.
size_employees	integer	Estimated number of employees.
annual_revenue	numeric	Annual revenue (numeric part).
revenue_scale	varchar(10)	Scale of the revenue (e.g., 'M', 'B').

notes	text	Qualitative notes.

4. Schema Details: Application Tracking Feature

These tables extend the model by recording individual job application events, linking directly back to the centralized companies table.

4.1. Table: job_titles (Job Title Lookup)

This lookup table stores unique job titles to facilitate analysis on the types of roles targeted.

Field Name	**Data Type**	**Description**
job_title_id	BIGSERIAL	Primary Key. Auto-incrementing, efficient lookup ID.
title_name	varchar(255)	The exact job title string from the application. **Must be UNIQUE.**
standardized_title	varchar(255)	Optional field for grouping titles (e.g., mapping all 'Sr. Software Eng' variations to 'Senior Software Engineer').
created_at	timestamp	Timestamp of when the title was first added.

4.2. Table: applications (Core Application Record)

This is the main transaction table for every application event.

Field Name	**Data Type**	**Description**
application_id	UUID	Primary Key. Used for universal uniqueness across systems.
company_id	integer	**Foreign Key to companies.company_id**

		(links application to the clean profile).
job_title_id	BIGINT	**Foreign Key to job_titles.job_title_id** (links application to the specific title).
user_id	varchar(100)	The unique ID of the user who owns this application.
date_applied	date	The date the application was submitted.
current_status	varchar(50)	Current stage (e.g., 'Applied', 'Interviewing', 'Rejected').
updated_at	timestamp	Last modification time (uses a trigger for auto-update).

4.3. Table: job_documents (Document Metadata)

This table tracks the necessary metadata for files uploaded related to a specific application. **It does not store the files themselves.**

Field Name	**Data Type**	**Description**
document_id	UUID	Primary Key.
application_id	UUID	**Foreign Key to applications.application_id** (ON DELETE CASCADE).
document_type	Custom ENUM	Specific type of document (e.g., 'RESUME', 'COVER_LETTER', 'JOB_DESCRIPTION').
file_path	varchar(512)	The secure, unique filename on the file system

		(e.g., a UUID or hash). **Must be UNIQUE.**
original_filename	varchar(255)	The name the user uploaded (for display purposes).
mime_type	varchar(100)	The detected MIME type (e.g., 'application/pdf').
upload_timestamp	timestamp	Time of the upload.

5. Design Rationale (Unified System)

1. **Centralized Company Data:** By linking applications.company_id to companies.company_id, all application tracking immediately benefits from the standardized company name and all enrichment data, ensuring consistent reporting across the entire platform.
2. **Application History:** The structure supports the required **1 Company** $\rightarrow$ **Many Applications** $\rightarrow$ **Many Documents** hierarchy, allowing users to track multiple job submissions to the same company over time.
3. **Data Integrity (Constraints):**
   - Foreign Key constraints prevent deleting a core company or job_title if an application depends on it (ON DELETE RESTRICT).
   - The job_documents metadata is automatically cleaned up when the parent application is deleted (ON DELETE CASCADE).
4. **Performance:** Using a dedicated lookup table for job_titles improves query performance and reduces redundant data storage in the main applications table.