

|  |
| --- |
| Business Template  **Recruıtment agency** |
|  |

Contents

[1 Business Description 3](#_Toc62212630)

[1.1 Business background 3](#_Toc62212631)

[1.2 Problems. Current Situation 3](#_Toc62212632)

[1.3 The benefits of implementing a database. Project Vision 3](#_Toc62212633)

[2 Model description 3](#_Toc62212634)

[2.1 Definitions & Acronyms 3](#_Toc62212635)

[2.2 Logical Scheme 3](#_Toc62212636)

[2.3 Objects 3](#_Toc62212637)

# 

# Business Description

## Business background

The recruitment industry serves as a bridge between talented individuals seeking employment and organizations searching for qualified professionals.

This Recruitment Agency aims to streamline the hiring process by maintaining a structured relationship among candidates, recruiters, and job opportunities.

Unlike traditional agencies that only focus on matching candidates to open positions, this agency also provides career development services such as resume writing, interview coaching, and skills improvement training. These services help candidates enhance their employability and align better with the requirements of modern job markets.

As the number of candidates, recruiters, and available job postings continues to grow, managing all information manually through spreadsheets or documents has become increasingly inefficient and error-prone. This project was initiated to create a centralized database system that supports all core recruitment activities while also integrating the agency’s additional career support services.

## Problems. Current Situation

Before implementing a structured database, the agency faces several issues that hinder its daily operations:

1. Fragmented data storage – Candidate profiles, recruiter details, and job postings are stored separately in multiple documents, making it difficult to update or cross-reference information.
2. Manual tracking of applications and services – Applications are processed through emails and spreadsheets, leading to inconsistent and incomplete records of candidate progress.
3. Lack of connection between candidates and their skills or services – The agency cannot efficiently identify which skills a candidate has or which services they have used.
4. Limited recruiter insight – Recruiters cannot easily view which candidates are qualified for which jobs, or which applicants are still in process.
5. Data redundancy and inaccuracy – Repeated information about candidates, contacts, and services increases the risk of inconsistency.
6. Poor reporting and decision support – Generating summaries like “top demanded skills,” “most requested services,” or “successful placement rate” is time-consuming and unreliable.

Overall, the lack of a unified data management system causes delays in the recruitment process, inefficient communication, and reduced operational transparency across the agency.

## the Benefits of implementing a database. Project Vision

The implementation of a **relational database** will create a structured, reliable, and scalable system for managing all recruitment activities and candidate development programs.

#### **Benefits:**

1. **Centralized Management:** All data — including candidates, recruiters, job postings, services, and applications — will be organized in one place.
2. **Data Consistency and Integrity:** Relationships between entities (e.g., candidates–skills, candidates–services, recruiters–jobs) are clearly defined, reducing redundancy.
3. **Faster Access and Processing:** Staff can quickly retrieve information about candidate skills, applied jobs, and service history.
4. **Improved Candidate Matching:** The system can identify candidates whose skills match job requirements more efficiently.
5. **Enhanced Reporting:** Managers can generate detailed reports and performance statistics about candidates, recruiters, and services.
6. **Scalability and Integration:** The database design in 3rd Normal Form (3NF) allows easy integration with a web interface or an AI-based matching system in the future.
7. **Professionalism and Accuracy:** Having structured, reliable data enhances trust between the agency, its candidates, and partner companies.

#### **Project Vision:**

The vision of this project is to develop a **fully normalized recruitment agency database** that not only automates the hiring and candidate tracking process but also integrates the agency’s **career development services** into the same system.

The database will support:

* Managing **candidate information, skills, and applications**
* Managing **recruiters and their job postings**
* Recording **services provided to each candidate**
* Ensuring **clear and accurate relationships** between all entities

In the long term, this database will serve as the foundation for an **intelligent recruitment platform** capable of recommending candidates for suitable jobs, tracking their service history, and providing data-driven insights into the agency’s operations.

# Model description

## Definitions & Acronyms

PK Primary Key

FK Foreign Key

Candidate A person registered the recruitment agency seeking employment.

Recruiter An employee posting job vacancies and reviewing applications.

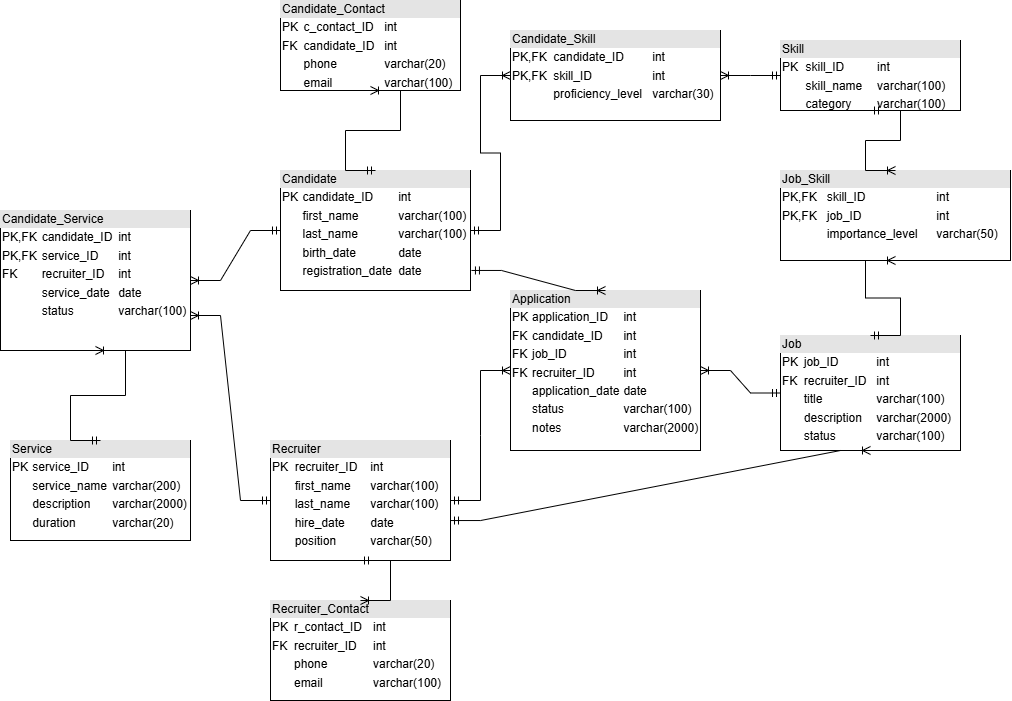
Application A submission made by a candidate for a specific job.

Service Additional support (resume writing, coaching, skills development) offered to candidates.

Skill A professional ability or competency possessed by a candidate or required by a job.

3NF Third Normal Form ensures data consistency and eliminates redundancy.

## Logical Scheme



## Objects

Below are the tables (entities) in the model with their key fields, relationships, and data types. This section describes all the main objects (tables) included in the database model.

Each table is defined with its purpose, key attributes, data types, and relationships.

All tables are normalized to Third Normal Form (3NF) to eliminate redundancy and ensure data consistency.

**Candidate**

Stores personal and registration information for each candidate registered with the recruitment agency.  
Each candidate can have multiple contacts, applications, skills, and services used.

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| Candidate | Candidate\_ID | Unique ID for each candidate, PK | INT |
| First\_name | Candidate’s first name | Varchar(100) |
| Last\_name | Candidate’s last name | Varchar(100) |
| Birth\_date | Candidate’s birth date | DATE |
| Registration\_date | Date when candidate registered | DATE |

Each candidate can have multiple contact records, skills, and applications.

Example with data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Candidate\_ID | First\_name | Last\_name | Birth\_date | Registration\_date |
| 10001 | John | Smith | 21/12/1997 | 10/05/2025 |

**Candidate\_Contact**

Contains contact details of candidates. A candidate may have multiple contact entries, for example if they have different phone numbers or emails.

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| Candidate\_Contact | C\_contact\_ID | Unique contact record ID, PK | INT |
| Candidate\_id | Foreign key referencing Candidate, FK | INT |
| Phone | Candidate’s phone number | Varchar(20) |
| Email | Candidate’s email address | Varchar(100) |

One to many relationship with candidate. Candidate\_ID is a foreign key.

|  |  |  |  |
| --- | --- | --- | --- |
| C\_contact\_ID | Candidate\_ID | Phone | Email |
| 1 | 1001 | +905874231625 | jsmith@mail.com |

**Candidate\_Skill**

Bridge table linking candidates and their skills.

Represents the many-to-many relationship between Candidate and Skill, and includes each candidate’s proficiency level.

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| Candidate\_Skill | Candidate\_ID | Foreign key referencing Candidate, PK, FK | INT |
| Skill\_ID | Foreign key referencing Skill, PK, FK | INT |
| Proficiency\_level | Candidate’s phone number | Varchar(30) |

A candidate can have multiple skills, and each skill can belong to multiple candidates.

|  |  |  |
| --- | --- | --- |
| Candidate\_ID | Skill\_ID | Proficiency\_level |
| 1001 | 10 | ADVANCED |

**Candidate\_Service**

Tracks which additional services (resume writing, interview coaching, etc.) have been used by candidates.

Represents a many-to-many relationship between Candidate and Service.

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| Candidate\_Service | Candidate\_ID | Foreign key referencing Candidate, PK, FK | INT |
| Service\_ID | Foreign key referencing Service, PK, FK | INT |
| Recruiter\_ID | Foreign key referencing Recruiter | INT |
| Service\_date | The date service was provided | DATE |
| Status | Status of the service | VARCHAR(100) |

Each candidate can receive multiple services from recruiters.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Candidate\_ID | Service\_ID | Recruiter\_ID | service\_date | status |
| 10001 | 5001 | 2001 | 21/12/2024 | COMPLETED |

**Service**

Lists all services offered by the recruitment agency, such as resume preparation, interview coaching, or skills training.

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| Service | Service\_ID | Unique service ID, PK | INT |
| Service\_name | Name of the service | VARCHAR(2000) |
| Description | Detailed service information | VARCHAR(2000) |
| Duration | Estimated service duration | VARCHAR(20) |

One to many relationship with Candidate\_Service.

|  |  |  |  |
| --- | --- | --- | --- |
| Service\_ID | Service\_name | description | duration |
| 5001 | RESUME WRITING | Resume writing workshop | 3 HOURS |

**Skill**

Holds information about various skills that candidates can have or that jobs can require.

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| Skill | Skill\_ID | Unique skill identifier, PK | INT |
| Skill\_name | Name of the skill | VARCHAR(100) |
| Category | The category of the skill | VARCHAR(100) |

Many-to-Many with Candidate (via Candidate\_Skill) and Job (via Job\_Skill).

|  |  |  |
| --- | --- | --- |
| Skill\_ID | Skill\_name | category |
| 10 | C++ | TECHNICAL |

**Job**

Contains job postings created by recruiters.

Each job can have multiple required skills and receive many applications.

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| Job | job\_ID | Unique job identifier, PK | INT |
| recruiter\_ID | Foreign key referencing Recruiter, FK | INT |
| Title | Job title or position name | VARCHAR(100) |
| Description | Detailed description of the job position | VARCHAR(2000) |
| Status | Indicates if the job is active or closed | VARCHAR(100) |

One-to-Many relationship with Recruiter, Many-to-Many with Skill via Job\_Skill.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Job\_ID | Recruiter\_ID | title | Description | status |
| 3001 | 2001 | Database Engineer | description | ACTIVE |

**Job\_Skill**

Bridge table defining required skills for each job posting.

Implements a many-to-many relationship between Job and Skill.

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| Job\_Skill | Job\_ID | Foreign key referencing Job, PK, FK | INT |
| Skill\_ID | Foreign key referencing Skill, PK, FK | INT |
| Importance\_level | Indicates the importance of the skill(required, optional) | VARCHAR(50) |

Each job may require multiple skills; each skill may be needed by multiple jobs.

|  |  |  |
| --- | --- | --- |
| Job\_ID | Skill\_ID | Importance\_level |
| 3001 | 10 | OPTIONAL |

**Application**

Stores all job applications submitted by candidates to recruiters.

Each record connects a candidate to a specific job and tracks application progress.

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| Application | Application\_ID | Unique ID for each application, PK | INT |
| Candidate\_ID | Foreign key referencing Candidate, FK | INT |
| Job\_ID | Foreign key referencing Job, FK | INT |
| Recruiter\_ID | Foreign key referencing Recruiter, FK | INT |
| Application\_date | Date when the candidate applied | DATE |
| Status | Current status(pending, interview, hired, rejected) | VARCHAR(100) |
| notes | Additional information about the application | VARCHAR(2000) |

Many-to-One relationship from both Candidate and Job sides.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Application\_ID | Candidate\_ID | Job\_ID | Recruiter\_ID | Application\_date | status | Notes |
| 1 | 1001 | 3001 | 2001 | 21/02/2023 | REJECTED | notes |

**Recruiter**

Stores basic information about recruiters who post job listings and provide candidate services.

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| Recruiter | Recruiter\_ID | Unique reqruiter ID, PK | INT |
| First\_name | Recruiter’s first name | VARCHAR(100) |
| Last\_name | Recruiter’s last name | VARCHAR(100) |
| Hire\_date | Date when recruiter joined the agency | DATE |
| Position | Job title/role of recruiter | VARCHAR(50) |

One-to-Many relationship with Job, Application, and Candidate\_Service.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Recruiter\_ID | First\_name | Last\_name | hire\_date | Position |
| 2001 | Anna | Dale | 03/08/2021 | HR SPECIALIST |

**Recruiter\_Contact**

Holds contact information for recruiters.

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field name | Field Description | Data Type |
| Recruiter\_Contact | R\_contact\_ID | Unique contact record ID, PK | INT |
| Recruiter\_ID | Foreign key referencing Recruiter, FK | INT |
| Phone | Recruiter’s phone number | Varchar(20) |
| Email | Recruiter’s email address | Varchar(100) |

One to many relationship with Recruiter.

|  |  |  |  |
| --- | --- | --- | --- |
| r\_contact\_ID | recruiter\_ID | Phone | Email |
| 1 | 2001 | +1234567891 | anndle@mail.com |