

# Data and Applications - Project Phase 1

Aryaman Kolhe (2022121002), Prakhar Jain (2022121008), and Vrinda Agarwal (2021101110)

## Introduction to the Mini World

In today's fast paced world, it has become increasingly difficult for people working in HR to find a good fit for their institution or company. Our *Only Jobs* mini world solves this issue by providing job seekers and hiring managers a one stop solution to all their problems.

It helps store various qualifications and projects worked on by people, along with available jobs, their minimum requirements and much more.

**Keywords:** Data Models, Hiring Database, Mini World

## 1. Purpose and Users of the Database

*Only Jobs* can be used to provide employment to people according to which job suits them best, in an efficient and accurate manner. Companies can use it to find talent for internships, part time/full time jobs, and aspiring students and others can use it to find a satisfying job.

Note that our dataset is not limited to the corporate world. We have introduced a variety of job descriptions that can be used for other fields as well. E.g. Scientists looking for research positions/ applying for the post of a professor in a University.

## 2. Applications of the Database

Our database has special applications that have been tailor made for both hiring managers and job seekers. E.g. An R&D company like IBM Research can find a researcher working in a specific field, with a minimum number of research projects/patents published with ease.

## 3. Database Requirements

### 3.1 Assumptions

We've used a few job descriptions as proof of concept, but this could be generalized to any job, and any company regardless of a person's origin, gender, and job requirements.

### 3.2 Strong Entity Types

1. Person	a. Person_Id	Artificial key, primary key, candidate key, Integer
	b. Name	Varchar(50), null
	c. Aadhar_Number	Unique, not null, candidate key, Integer
	d. Job_Id	Foreign Key(Job_Id), Can be null (unemployed)
	e. Phone_Number	Composite (Country_Code: varchar(10), Number: varchar(20)). Not all countries use 10 digit phone nums
	f. Date_of_Birth	Date type, not null
	g. Age	Derived from Date_of_Birth, nonnegative integer
	h. Gender	varchar(10)
2. Project	a. Project_Id	Artificial Key, Primary Key, Integer
	b. Type	varchar(20), belongs to ('Paper', 'Patent', 'Coding', ...)
	c. Description	varchar(200), not null
	d. Affiliation	Affiliated to a lab or company
	e. Field	varchar(50), multivalued
3. Educational_Institutes	a. Educational_Institute_Id	Artificial Primary Key, integer
	b. Name	varchar(50), not null
	c. Courses_Offered	varchar(50), multivalued
	d. Departments	varchar(50), multivalued
	e. Type	varchar(50), must belong to ('School', 'University', ...)
4. Institute	a. Institute_Id	Artificial Primary Key, Integer
	b. Name	varchar(50), not null
	c. Type	varchar(50), must belong to ('Company', 'University', 'Hospital', ...)
	d. Project_Id	Foreign Key(Project_Id), Integer, multivalued

5. Jobs	a. Job_Id	Artificial Primary Key, Integer
	Job_Description	varchar(200), not null
	c. Institute_Id	Foreign Key(Institute_Id), Integer, not null
	d. Salary	Composite (CTC: float, Base: float)
	e. Start_Date	Date, not null
	f. End_Date	Date, can be null
	g. Type	varchar(50), must belong to ('Software Developer', 'Project Manager', 'Researcher', ...)
	h. Duration_Worked	Derived attribute (if end date is null: Current Date - Start Date. Else: End Date - Start Date)

### 3.3 Weak Entity Types

1. Educational_Qualifications	a. Person_Id	Partial Key, cannot be null, unique
	b. Educational_Institute_Id	Foreign Key(Educational_Institute_Id), can be null (you can get a job without a degree)
	c. Description	varchar(200), can be null
	d. CGPA	nonnegative float, subject to constraint [0, 10], can be null
	e. Qualifications	varchar(50), must belong to ('PhD', 'Masters', 'Bachelors', 'Diploma', 'School', ...), can be null

2. Institute_Departments	a. Institute_Id	Partial key
	b. Department_Id	unique, varchar(50)
	c. Name	varchar(50), cannot be null
	d. Project_Id	Foreign Key(Project_Id), multivalued
3. Minimum_Job_Requirements	a. Job_Id	Partial Key
	b. Min_Years_of_Experience	float
	c. Min_Qualification	varchar(50) <sup>1</sup>

### 3.4 Relationship Types (Degree <= 3)

1. Point_of_Contact	Degree	2
	Entity Types	Person, Person ( <i>recursive</i> )
	Cardinality	[(1,n) (1,1)]
2. Work_Experience	Degree	2
	Entity Types	Person, Jobs
	Cardinality	[(1,1), (0, n)]
3. References	Degree	3
	Entity Types	Person, Institute, Job) <sup>2</sup>
	Cardinality	[(1,1), (1,n), (1,1)]
4. Studied_In	Degree	2
	Entity Types	Person, Educational_Institute
	Cardinality	[(1,1), (0,n)]

5. Offers	Degree	2
	Entity Types	Institute, Jobs
	Cardinality	[(1,1), (0,n)]
6. Department_Works_on_Project	Degree	3
	Entity Types	Institute, Institute_Departments, Project <sup>3</sup>
	Cardinality	[(1,1), (1,n), (0,n)]

### 3.5 Relationship Types (Degree > 3)

1. Associated_With	Degree	4
	Entity Types	Person, Institute, Jobs, Project
	Cardinality	[(1,n), (1,1), (1,1), (1,n)]

Footnotes:

- 1 - We're storing qualifications with an hierarchy: PhD > Masters > Bachelors > Diploma > School Must belong to (PhD, Masters, Bachelors, Diploma, School)
- 2 - A point of contact works in a company, and refers the job to a job seeker
- 3 - Primary Key for this table = (Institute\_Id, Department\_Id)

## 4. Functional Requirements

### 4.1 Retrievals

- Selection - Print all person details of people who studied in IIIT-H
- Projection - Special Opportunities to improve Gender Diversity - Print names of all people who fit the criteria
- Aggregate - Find the average package of an alumni belonging to a particular institute
- Search - Filter out candidates for a particular job, based on whether they clear the minimum job qualifications or not

All these use joins, selection, and projection.

### 4.2 Modifications

- Insert + Check for Integrity Constraints - Add a new job to the market, such that the company offering the job is legitimate. Add a new person to Person (example, when a fresher joins an educational institute)
- Update - If a person graduates, update his/her qualifications
- Delete - Remove invalid/expired jobs from Jobs

## 5. Summary

Therefore, we have defined our own *Only Jobs* Database, which can be used by both employers and employees for better prospects and job fits, which will lead to better job satisfaction.

We've also included tailor made functional requirements, that can be extended even further to help the users find the information they need.

— Thank you —