**Project checkpoint 2**

[Link](https://www.kaggle.com/datasets/ravindrasinghrana/job-description-dataset)

1. **A description of the application and its features. State as clearly as possible what you want to do (bullet points preferred). This can be extracted from your Checkpoint 1 write-up.**

This application will serve as a user interface for searching job listings within specified parameters. User accounts will allow the user to store, track, and access information concerning job listings. Application features will be incorporated into a standalone web application using QT. The following features will be:

* User account creation and management.
* Tracking saved job listings (allowing the user to add/modify/delete saved jobs from the account)
* Searching databases using filters (i.e. salary, proximity, etc.).
* Viewing company profiles associated with job listings.
* Compare multiple job listings to evaluate.
* Easy navigation to and from job listings and user accounts.

**2. The ER diagram of your application domain, plus the English description.**

Entity sets:

* Job
  + ID
  + Title
  + Experience
  + Qualifications
  + Salary
  + Work\_Type
  + Role
  + Description
  + Benefits
  + Skills
  + Responsibilities
* Company
  + Name
  + size
* Recruiter
  + Name
  + Contact\_number
* Location
  + City
  + Country
  + Latitude
  + longitude
* Job Portal
  + Website-Domain

Relationships:

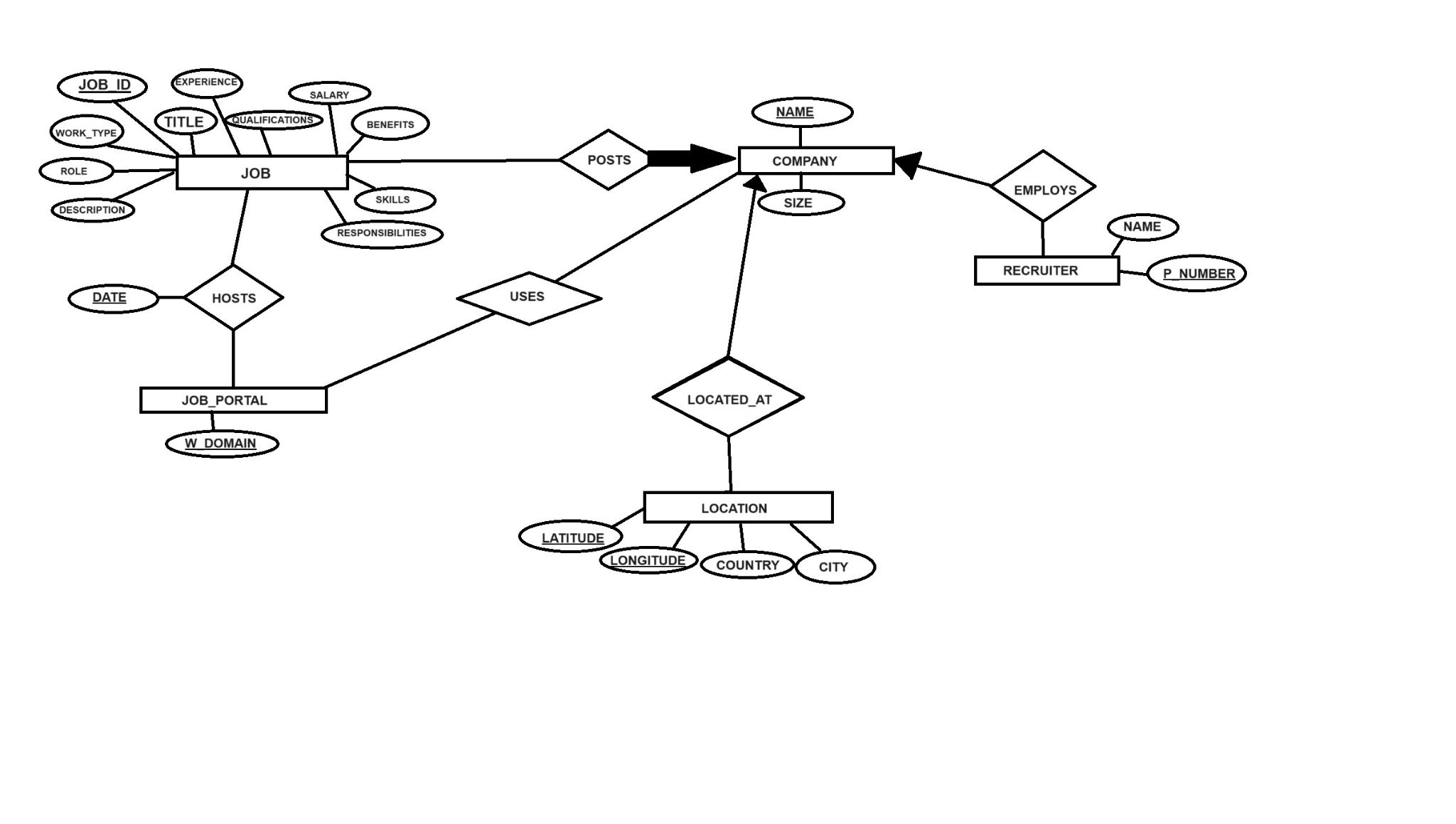
Company posts Job - one-to-many (←)

Company employs recruiter - one-to-many (←)

Company located\_at Location - one to many (←)

Job Portal hosts Job with attribute: Date - many-to-many (–)

Company uses Job Portal - many-to-many (–)



The database will keep track of job listings that include the following attributes: job ID, title, experience, qualifications, salary, work type, role, description, benefits, skills, and responsibilities. A company can post many job listings, but each job listing has one specific company. Companies include name and size. Recruiters work at a company; there are multiple recruiters per company. A name and a contact number describe a recruiter. Companies are at one or many locations. Locations are described by their city, country, latitude, and longitude. Companies post their job listings on many Job Portals. Job Portals are identified by their website domain. Job Portals host job listings that are first posted on a specific date.

**3. A description of the functionalities that you plan to offer. At the minimum, you must handle inserting, deleting, and modifying tuples in the database and handle queries that involve joins (all joins), projections, selections, and aggregations (e.g., sum, average, max, min, etc.). Note: notions such as aggregations will be covered shortly in the class.**

* Inserting job listing entities into favorites
* Deleting job listing entities from favorites
* Modifying job listing entities from favorites
* Comparing job listings to other job listings
* The ability to view the company after searching, and go back to the search
* The ability to filter by qualifications listed in the user’s profile and indicate whether qualifications are fulfilled for a job listing
* Handling queries that involve joins, projections, selections, and aggregations

4. **Specify the kind of queries that will be commonly asked in your application in English and how they match the functionalities you plan to offer. For example, if you are developing a database to manage books and want to provide two features: search, and review. For the search feature, you will provide the functionalities to search by authors, genre, user rating, and date. For the review feature, you will provide the functionalities to display the average rating for a book or enable users to enter a review for a specific book.**

* Our program will offer a search feature that you can use to search by:
  + Job Title/type
  + Radius distance from point
  + Salary
  + Qualifications
  + Role
  + Company name
  + Responsibilities
* After searching, the following functionalities will be displayed:
  + Other search feature options about the company
  + The ability to save/delete job listings to a favorites
* The ability to modify job listings
* The ability to add comments/notes
* Functionality to compare job listings by the following criteria:
  + Salary
  + Distance from point
  + Qualifications
  + Responsibilities
  + Company Name

5. **Discuss the planned user interface to your database. At the minimum, this user interface must be a text-based one. It is preferred to be a Web-based or mobile interface.**

For this application, we’ll be utilizing QT for C++. This GUI plugin offers many built-in ease-of-use features to provide code in a standalone application. Upon booting up the program, the user is prompted to enter an existing account or create a new account. These accounts will be stored locally on the user’s computer. After completion, the user is sent to the home page, displaying a search function in the middle of the screen. In the top corner, there will be a button to access the user’s account anytime. In the user’s account, there will be a table storing saved/favorite listings; the user can modify/delete these listings at any time. The search function will use a drop-down menu to search by filters. Once the user selects a filter, they will enter their query into the search bar. The function will display a list of all job listings associated with the filter. A user can click on an individual listing to see details pertaining to the specific example. A job listing will also include a button to favorite the listing. These favorites are what’s displayed on the user’s profile.

6. **A description of software platforms/languages that you plan to use.**

We plan to use C++ for the language, and QT for the implementation. QT has been used by many companies, including Mercedes and LGE; thus, the software is complex and has few limitations in design. MySQL will be used for the database as it works well with C++.