Employment Agency
DATABASE-BASED APPLICATION

Principles Of Database Management | April 1st 2019

Table of Contents

Introduction

The main objective of this project is to create an employment agency application which assists businesses in meeting their human resources requirements while also serving the needs of job-seekers. This software aims to help companies find the right person for a job by providing information about applicant's education, work experience while also provide job candidates details on the firm, which they consider to apply for a position at. The final purpose of this application is to reduce cost and time for both human resources management of businesses and job-seekers alike.

Project Overview

- Name of the project:

    - Employment Agency Application

- Members of the team:

    - **Chung Minh Nhat - ITITIU17012**

    - **Nguyen Tran Chi Hieu - ITITIU17110**

    - **Chiem Quoc Hung - ITITIU17067**

- Tools in use:

    - MySQL

    - Qt Creator C++

    - Microsoft Word (report)

    Functionalities of Employment Agency Application:

1/ Provide login account for both recruiters and job-seekers.

2/ Login account contains basic information of a person such as Name, Gender, Date of Birth and specific details depend on which type of user.

3/ Job-seekers can edit their profiles, CVs, educations and work experiences.

4/ Recruiters can edit their profiles, job titles and facts about companies for which they work.

5/ Provide search engine to help seekers and recruiters find potential work or applicants.

6/ Recruiters can add, modify and delete their vacancies.
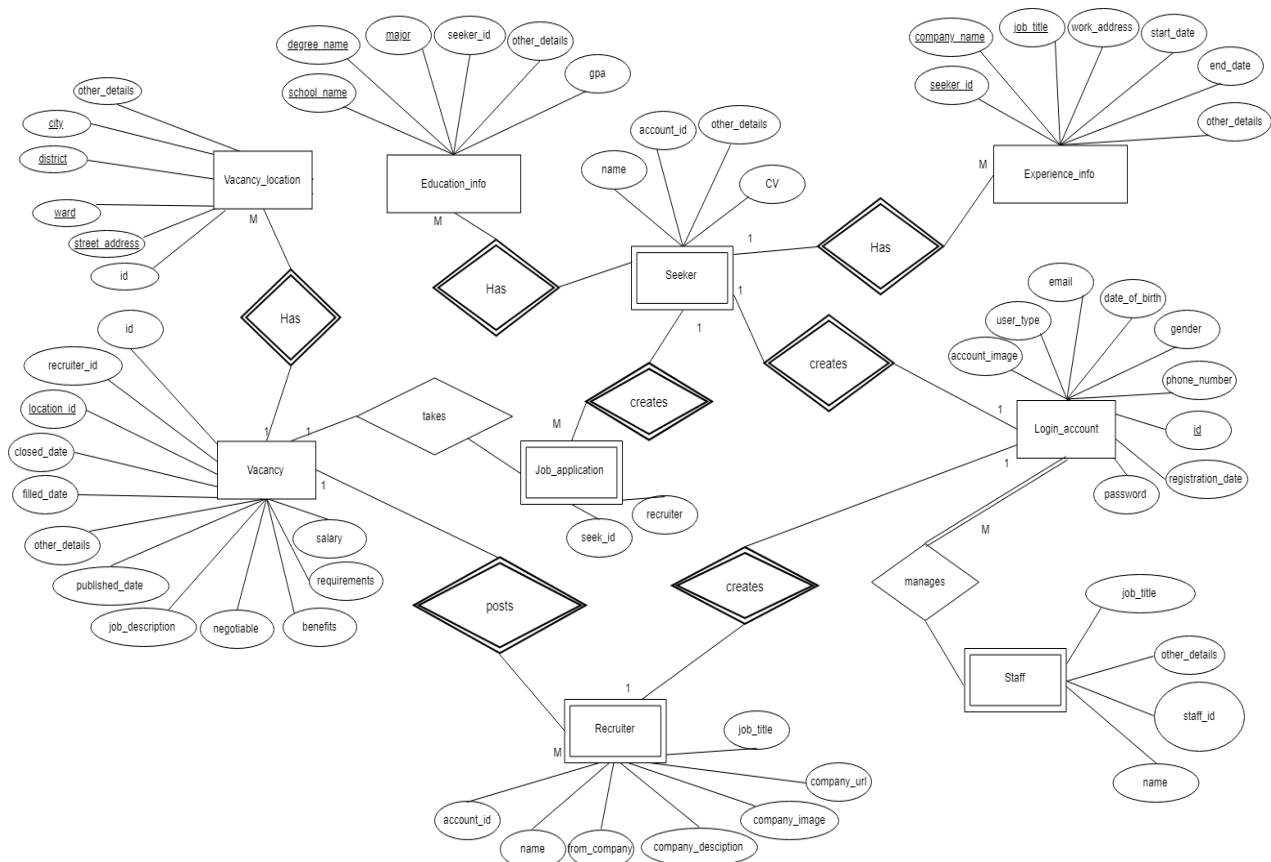
7/ Job-seekers can apply for the jobs they want.

System and Design Specification

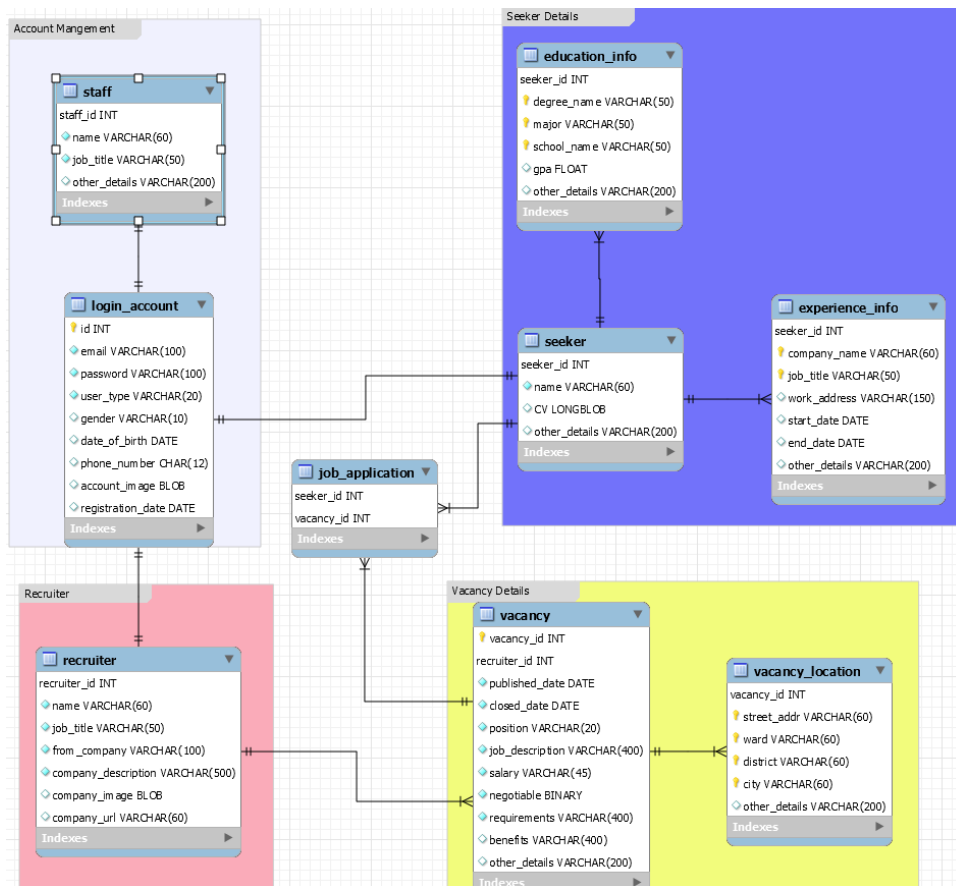- Database

Entity Relationship Diagram:

This database is based on Boyce-Codd Normal Form.

We decided to use an ERD, since it can illustrate the view of the entire system

in a way that facilitates the understanding of such system and therefore ease the implementation of our database. Moreover, ERD can be easily drawn, using tools that support making flowcharts.

Relationship diagram

We provide the notation as follow:

(part of) primary key

foreign key and (part of) primary key

not null

can be null

## Project Implementation

1. Database

Employment Agency database was first created following exactly what the ERD and the relational diagram show. We used basic SQL queries to initialize involved tables and set up relationships between them. At this time, we didn't need to put any records into the database, since the application would then carry on with the work.

2. Application
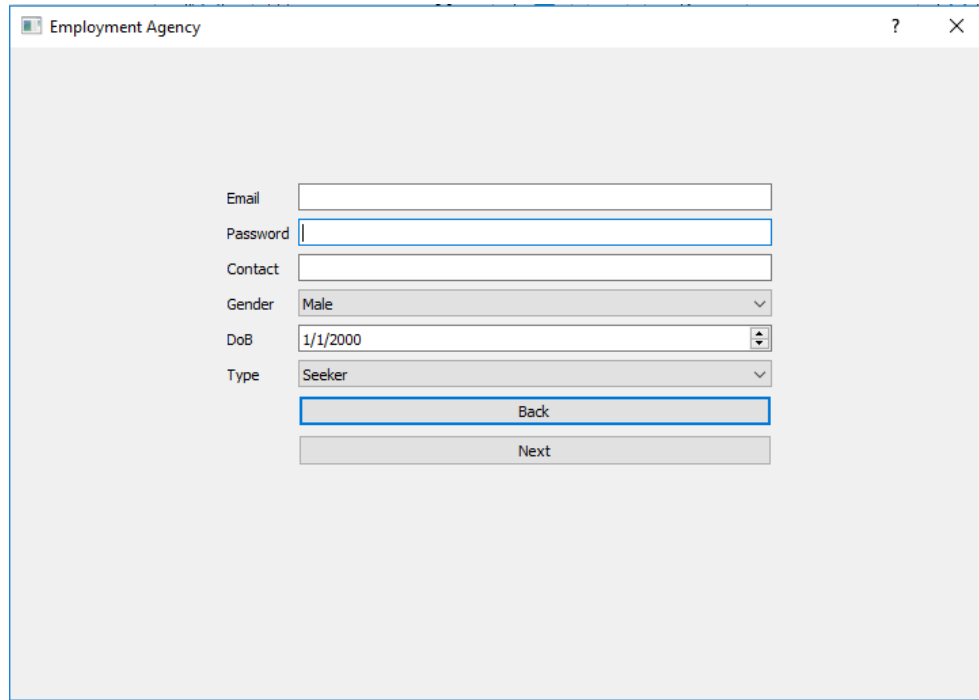
- Mainwindow: when the application (mainwindow) is open, the database must be connected so any implementation on signing in or signing up is saved to the database. After spending some days finding a proper way to connect Qt with EmploymentAgency database, we came up with the code segment:

```
// connect the database
db = QSqlDatabase::addDatabase("QSQLITE");
db.setDatabaseName("C:/Users/Admin/Documents/GitHub/EmploymentAgency/Database/EmploymentAgency");
if (db.open()) {}
else {
    QMessageBox::information(this, "Login call", "Database failed");
}
```

The argument "QSQLITE" specifies the type of the database driver used for the connection. We then passed a directory to .setDatabaseName to direct the application to EmploymentAgency database. The if-else lines of code is to check if the database db has been connected yet.
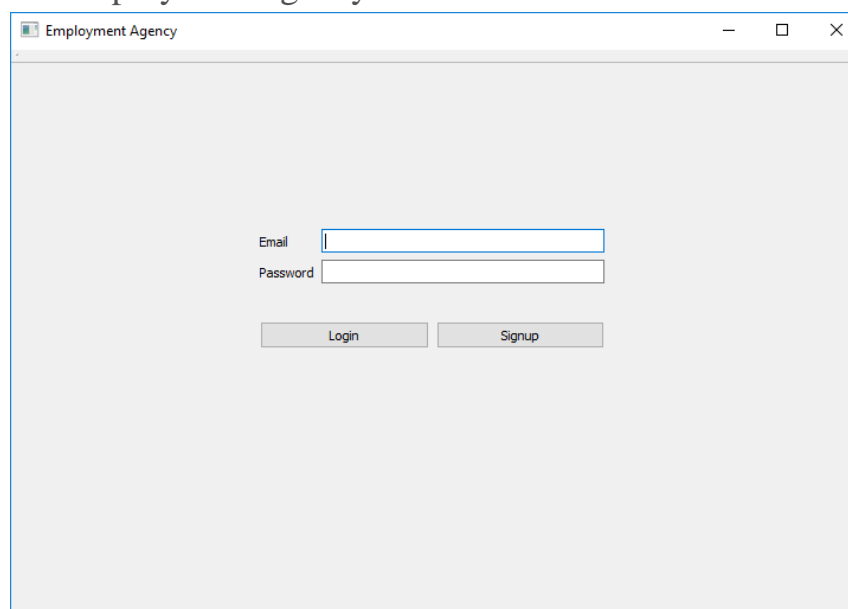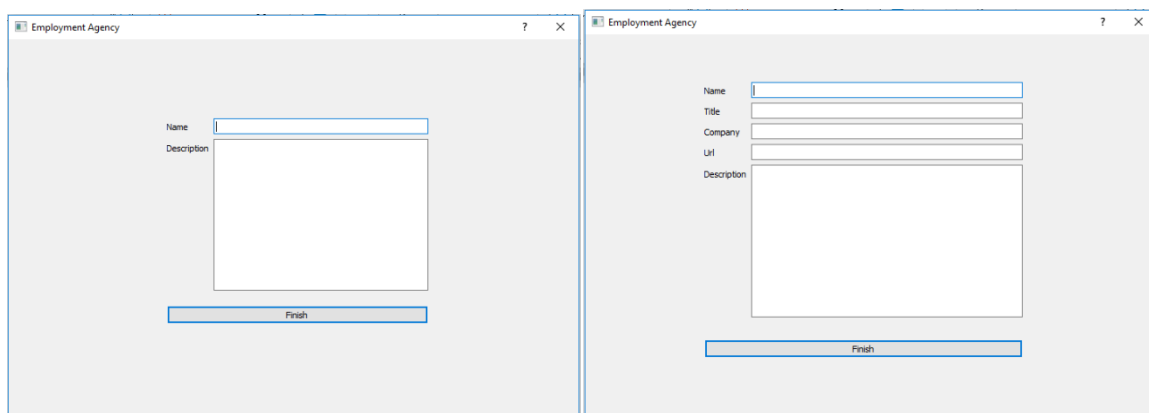
- Sign up window: this window is showed if the user wants to create an account by clicking on Sign up button from Mainwindow. Subsequently, a new window pops up, letting users type in required information.

Once Next button is clicked, the typed data will be saved into the EmploymentAgency database, then another window shows up corresponding to the user type (Seeker/ Recruiter).

- User profile and searching tool window: is showed after a successful log in. The window shows user's profile and let the user update their information (education, experience, vacancy) and search for vacancy/seeker profile.

Figure 1 - update information, search information window

From this window, users can do more interactions with it: update further information and search for vacancy/ seeker. Looking at the image above, the right part of the window is for a seeker to add their education information (university, major, degree name...) and experience information (company name, title, duration...). By clicking on Add button, another window shows up, enabling the user to add such information.

From now on, windows for a seeker will be displayed. Recruiter's windows are pretty much the same to those.

Figure 2 - add education/ experience window

The recruiter's profile also shows basic details. On the right-hand side, first table view is for job applications which other seekers sent, second table view shows vacancies that the recruiter created.

The button Add let recruiter create new vacancies.

Dialog   ?   ✕

Position

Published Date   1/1/2000

Closed Date   1/1/2000

Job Description

Salary   ☐ Negotiable

Requirements

Benefits

Other details

OK   Cancel

The recruiter can fill in all the information about the new vacancy. On the right, this is supposed for vacancy's location. This cannot be used currently. Recruiter has to create the vacancy first then go back to profile, double click the newly-created vacancy.

This shows details of the vacancy and a Add button with which recruiter can add an address for the vacancy.

Moreover, user can search for vacancy offerings by clicking on Search button, yielding:

The Load button helps shows available vacancies so seeker can choose which to apply by double clicking on one. Another different window is shown displaying the vacancy details:

Figure 3- search and apply window

If the seeker finds the appropriate job and wants to apply for it, he or she will just click on Apply button to apply.

Last but not least, this application needs staff who can manage all login accounts, their information. For the moment, they can only check every information avaiable on the database.

The staff can search using SQL query.

Conclusion

From the project, we achieved deeper understanding about the procedure of building an application and how to connect an application to a database.

Future Work:

- Add new work categories, skill sets
- Make ability to connect with internet