

Đặc tả bài toán:

Cho mô hình lớp một phần của ứng dụng hệ thống quản lý tuyển dụng nhân sự trong công ty, được mô tả như sau: Công ty có nhiều vị trí công việc (*Position*) cần tuyển dụng như: Manager, Developer, Tester, Business Analyst, ... Mỗi vị trí có thể tuyển nhiều nhân sự, mỗi ứng viên ứng tuyển (*Candidate*) chỉ có thể ứng tuyển (*APPLY*) cho một vị trí công việc.

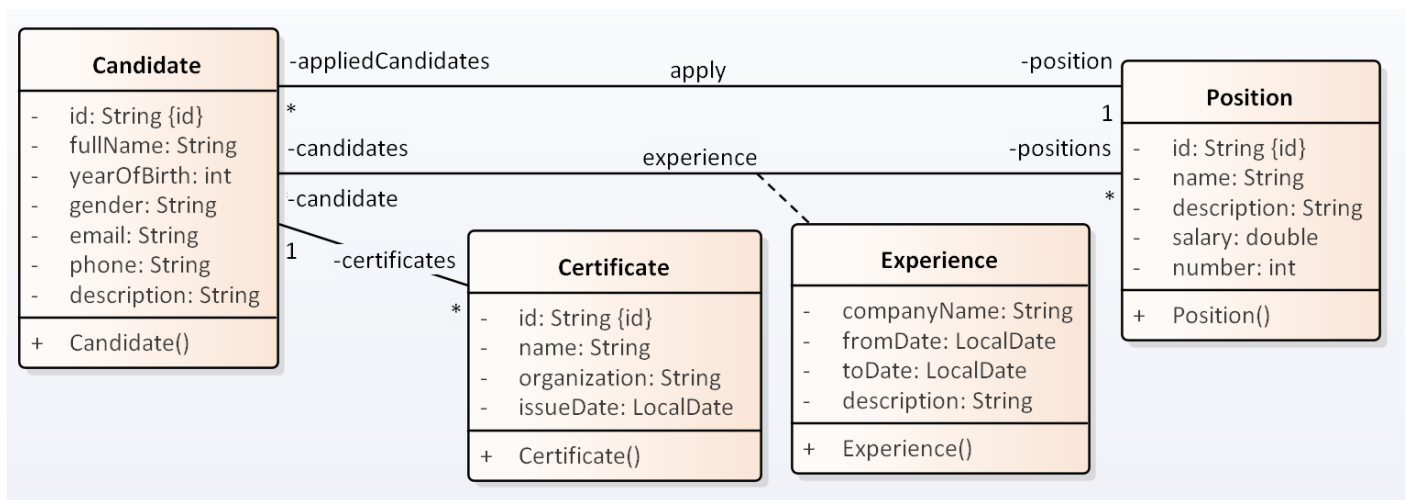
Mỗi ứng viên có các bằng cấp (*Certificate*) mà họ đạt được nhất định và cũng có thể có những kinh nghiệm làm việc (*Experience*) tại các công ty khác nhau. Mỗi kinh nghiệm trên một vị trí làm việc sẽ gồm có một số thông tin như: tên công ty (*companyName*), từ thời gian (*fromDate*), đến thời gian (*toDate*), mô tả công việc (*description*).

Thông tin về vị trí công việc bao gồm: Mã số (*id*), tên vị trí (*name*), mô tả công việc (*description*), lương cơ bản (*salary*) và số lượng tuyển dụng (*number*).

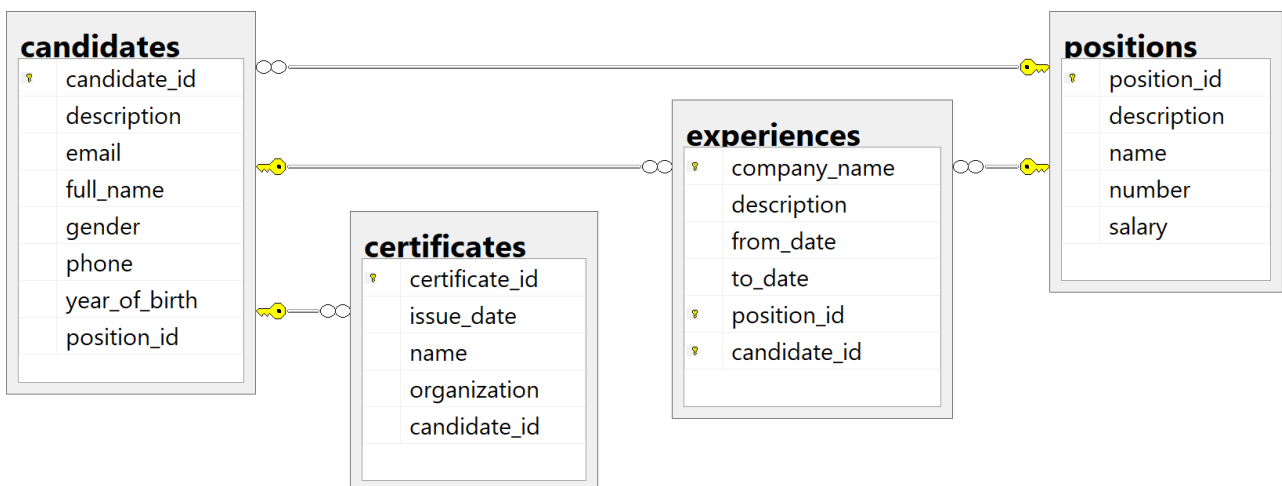
Thông tin về ứng viên bao gồm: Mã số (*id*), họ tên (*fullName*), năm sinh (*yearOfBirth*), giới tính (*gender*), email, số điện thoại (*phone*), mô tả bản thân (*description*).

Thông tin về bằng cấp bao gồm: Mã số (*id*), tên bằng cấp (*name*), nơi cấp (*organization*), ngày cấp (*issueDate*).

Class diagram (mô hình lớp)



Database Diagram (mô hình CSDL)



Tạo các project tên gồm: HọTênSV_MãSốSV_ SốMáy[Client]. Dùng ngôn ngữ lập trình JAVA kết nối CSDL quan hệ và hiện thực các yêu cầu sau:

Câu 1: Dùng JPA ORM ánh xạ các thực thể trong mô hình lớp sang mô hình cơ sở dữ liệu quan hệ (*tên database là [tên và mã số của sinh viên]*).

Câu 2: Chạy dữ liệu mẫu được cho sẵn vào các bảng tương ứng.

Câu 3: Hiện thực chương trình dựa trên mô hình client server (*Client-Server* với *Socket*), thiết lập một server triển khai trên mạng LAN với các chức năng như sau (*port* là 4 số cuối MSSV, *host* là máy SV làm bài):

a) Liệt kê danh sách các vị trí công việc khi biết tên vị trí (*tìm tương đối*) và mức lương khoảng từ, kết quả sắp xếp theo tên vị trí công việc.

+ listPositions(name: String, salaryFrom: double, salaryTo: double): List<Position>

b) Liệt kê danh sách các ứng viên và số công ty mà các ứng viên này từng làm.

+ listCandidatesByCompanies() : Map<Candidate, Long>

c) Tìm danh sách các ứng viên đã làm việc trên một vị trí công việc nào đó có thời gian làm lâu nhất.

+ listCandidatesWithLongestWorking(): Map<Candidate, Position>

d) Thêm một ứng viên mới. Trong đó mã số ứng viên phải bắt đầu là C, theo sau ít nhất là 3 ký số.

+ addCandidate(candidate: Candidate) : boolean

e) Tính số năm làm việc trên một vị trí công việc nào đó khi biết mã số ứng viên.

+ listYearsOfExperienceByPosition(candidateID: String): Map<Position, Integer>

f) Liệt kê danh sách các ứng viên và danh sách bằng cấp của từng ứng viên.

+ listCandidatesAndCertificates(): Map<Candidate, Set<Certificate >>

Câu 4: Dùng Junit 5, viết lớp kiểm thử, tạo các test case cho các phương thức CRUD trên.

Câu 5: Hiện thực chương trình client để trao đổi dữ liệu từ xa giữa client và server.

Hướng dẫn chấm:

Chỉ đạt tối đa 50% tổng điểm toàn bài, nếu không tách các layers.

Trừ 0.5 điểm / 1 thành phần sai (*đặt tên project, tên database, host name, port*)

Trừ 0.25 điểm cho mỗi vị trí thiếu ràng buộc: not null, unique hoặc sai data type

Câu 2 - Không chấm điểm nếu sinh viên không chạy được dữ liệu mẫu cho sẵn

Câu 3 - Trừ 0.5 điểm cho mỗi câu nếu viết không theo Method Signature cho trước

Điểm tối đa là 0.5/câu nếu không viết câu truy vấn đầy đủ mà duyệt trên tập đối tượng để xử lý.

Câu 4: Phần kiểm thử dữ liệu - Chỉ chấm những câu xử lý viết đúng từ 50% trở lên.

Câu 5: Phần client - Không chấm nếu không tách project riêng biệt, đóng gói thừa trừ 0.5 điểm