

**Due dates:** March 17, April 7, and April 21, 2016

### Project description

In October 2015, New Orleans ranked #1 City for fastest median wage growth because the range of wage data start from 2005<sup>1</sup>. In July 2015, New Orleans ranked #1 in America for creative professionals<sup>2</sup> and #1 in workforce training. Currently, for computer-science majors job seems not a problem. But do you know that the seven technology jobs received the hardest hit by layoffs in 2015 were computer hardware engineers, network and computer system administrators, computer programmers, computer support, Web developers, computer and information research scientists, information security analysts<sup>3</sup>? No worry, the unemployment rates of these types of workers were only between 3-5% and large demands of later four workers are strong in the next decade according to the forecast of the same report. No wonder we still see jobless people, even painfully see jobless college graduates. While creating one's own business is a solution, finding a job is a more common way to start a college graduate's career. To be qualified for a job, one often has to retool him/herself by learning and training. In this project, your task is to build an information system that can help optimize training programs, job assistance services, career planning, and company recruitment. The central efforts for this task is to design a database and a number of database applications that support the services provided by this system. If your design is good and if Louisiana wants to keep the crown of workforce training in nation, the State has to use your design.

Workers are classified as staff (salary worker), wage-workers, and volunteers. Companies and organizations hold jobs. Workers acquire knowledge and skills through two types of learning activities, academic courses and job trainings. The element of these two types of activities is course. The mission of learning is to fill up knowledge/skill gaps between the workers' possession and the job's requirements. In industry, practitioners refer this kind of learning activities as *training*, so do we. Generally, academic courses help students learn knowledge; training courses help trainees gain skills. Each course covers a set of knowledge/skills. For example, CSCI4125 covers: data modeling (medium level), relational models (medium level), SQL (medium level), JDBC programming (beginner level), RDBMS design (medium level), database concurrency control (advanced level), and relational object database (medium level).

A person can work on one or more jobs. A job is a slot that pays the worker. The nature of every job is described by a "job profile" that specifies many things including the required knowledge/skills. Typically, there are multiple jobs of the same job profile. There are standards of job profile descriptions.

Some important entities along with their important attributes are listed below. The attributes surrounded by a pair of braces ( { } ) may have multiple values, the attribute that carries a plus (+) is a composite field which you should flatten in relational modeling. Note, the purpose of the following information is to describe the systems, **not** to give you an actual data schema. For example, "company" in "Job(job\_code, type, pay\_rate, pay\_type, company)" means the identifier of a company, rather than the company object. The main focus of your design is the database schema configuration. In your database, you should consider the nature of the data and the queries to process. You will need additional relationships and entities defined by yourselves.

Person(per\_id, name, address<sup>+</sup>, zip\_code, email, gender, {phone})  
where per\_id is a the primary key.

Job\_profile(pos\_code, title, description, avg\_pay, {required\_skill})  
where pos\_code is a unique identifier, and required\_skill refers to one or multiple skills represented by k\_code.

---

<sup>1</sup> <http://gnoinc.org/news/awards-rankings/new-orleans-ranks-1-city-for-fastest-median-wage-growth/>

<sup>2</sup> <http://gnoinc.org/news/awards-rankings/new-orleans-ranks-1-in-america-for-creative-professionals/>

<sup>3</sup> [http://www.informationweek.com/strategic-cio/7-tech-jobs-hardest-hit-by-layoffs-in-2015-/d/d-id/1324526?elq=a328144e627e45418631f9833716ed1d&elqTrackId=d22c4ae72f0642c0a4b61dda9786c10a&elqaid=68254&elqat=1&elqCampaignId=19950&image\\_number=9](http://www.informationweek.com/strategic-cio/7-tech-jobs-hardest-hit-by-layoffs-in-2015-/d/d-id/1324526?elq=a328144e627e45418631f9833716ed1d&elqTrackId=d22c4ae72f0642c0a4b61dda9786c10a&elqaid=68254&elqat=1&elqCampaignId=19950&image_number=9)

**Job(job\_code, type, pay\_rate, pay\_type, company)**

where job\_code is a unique identifier, type can be “full-time”, “part-time”, and pay\_type can be wage or salary, pay\_rate is the hourly rate for wage or annual pay for salary. Note, the relationship between jobs and job profiles is many-to-one. In our system, we consider each job will be of one job profile only. (This is often not true for high-end jobs.)

**Company(comp\_id, address<sup>+</sup>, zip\_code, primary\_sector, {speciality}, website)**

Where primary\_sector refers to the business sector the company belongs to, such as tourism, oil and mining, entertainment, service, education, technology. Specialities can be construction, logistics service, computer hardware, software, electronics, basic materials, IT consulting, transportation.

**Course (c\_code, title, level, description, status, retail\_price)**

where c\_code is a unique identifier, and status is either active or expired.

**Section (c\_code, sec\_no, complete\_date, year, offered\_by, format, price)**

Where the triple (c\_code, sec\_no, year) form a unique identifier for every section, format can be (classroom, online-sync, online-selfpaced, or correspondence), offered\_by refers to a university or a training company.

**Knowledge\_Skill(ks\_code, title, description, level)**

where ks\_code is a unique identifier (you may use the skill code defined by the Department of Labor), and level can be “beginner”, “medium” or “advanced”.

In a person’s career, working on multiple jobs is common. Sometimes, a person’s working history can make him/her invaluable at certain situation. Recruiters often seek applicants’ experience. For this reason, the database should not only represent the job a person is currently working holding, it should also track every worker’s job history.

The information system will support at least two services, the *career planning service* and the *training service*. The *career planning service* needs information about workers’ education, training and skills in order to help find workers suitable for a job. The *training service* can recommend the courses for a person who pursues a job by recognizing the missing skills of the person for the job.

## **Team organization**

You are required to form a team made of two people. Each team member should work on one or more different service. Every team member should perform every aspect in the development process of database applications such as E-R modeling, SQL, and Java. Do not divide your tasks horizontally such as one write SQL, the other writes Java. Such kind of task division will give the Java guy serious disadvantages in earning a fair grade.

## **Queries**

The following queries are given to illustrate the capabilities of your system. Your database should support at least the following queries:

1. List a company’s workers by names.
2. List a company’s staff by salary in descending order.
3. List companies’ labor cost (total salaries and wage rates by 1920 hours) in descending order.
4. Find all the jobs a person is currently holding.
5. List a person’s knowledge/skills in a readable format.
6. List the skill gap of a worker between his/her job(s) and his/her skills.
7. List the required knowledge/skills of a job profile in a readable format.
8. List a person’s missing knowledge/skills for a specific job in a readable format.
9. Find the courses each of which alone can cover a given skill set.
10. List the courses (course id and title) that each alone teaches all the missing knowledge/skills for a person to pursue a specific job.
11. Suppose the skill gap of a worker and the requirement of a desired job can be covered by one course. Find the “quickest” solution for this worker. Show the course, **section information** and the completing date.

12. If query #9 returns nothing, then find the course **sets** with the minimum number of courses that their combination covers the given skill set. The considered course sets will not include more than three courses.
13. List the course sets that their combination covers all the missing knowledge/skills for a person to pursue a specific job. The considered course sets will not include more than three courses.
14. Find the cheapest course choices to make up one's skill gap by showing the courses to take and the total cost (by **section prices**). The considered course sets will not include more than three courses.
15. List all the job profiles that a person is qualified for.
16. Find the job with the highest pay rate for a person according to his/her skill qualification.
17. List all the names along with the emails of the persons who are qualified for a job profile.
18. When a company cannot find any qualified person for a job, a secondary solution is to find a person who is almost qualified to the job. Make a "missing-one" list that lists people who miss only one skill for a specified job profile.
19. List the skillID and the number of people in the missing-one list for a given job profile in the ascending order of the people counts.
20. Suppose there is a new job profile that has nobody qualified. List the persons who miss the least number of skills and report the "least number".
21. For a specified job profile and a given small number  $k$ , make a "missing- $k$ " list that lists the people's IDs and the number of missing skills for the people who miss only up to  $k$  skills in the ascending order of missing skills.
22. Given a job profile and its corresponding missing- $k$  list specified in Question 22. Find every skill that is needed by at least one person in the given missing- $k$  list. List each skillID and the number of people who need it in the descending order of the people counts.
23. In a local or national crisis, we need to find all the people who once held a job of the special job-profile identifier.
24. Find all the unemployed people who once held a job of the given job-profile identifier.
25. Find out the biggest employer in terms of number of employees or the total amount of salaries and wages paid to employees.
26. Find out the job distribution among business sectors; find out the biggest sector in terms of number of employees or the total amount of salaries and wages paid to employees. (required for graduate students only)
27. Find out the ratio between the people whose earnings increase and those whose earning decrease; find the average rate of earning improvement for the workers in a specific business sector. (required for graduate students only)
28. Find the job profiles that have the most openings due to lack of qualified workers. If there are many opening jobs of a job profile but at the same time there are many qualified jobless people. Then training cannot help fill up this type of job. What we want to find is such a job profile that has the largest difference between vacancies (the unfilled jobs of this job profile) and the number of jobless people who are qualified for this job profile.
29. Find the courses that can help most jobless people find a job by training them toward the job profiles that have the most openings due to lack of qualified workers.
30. **List all the courses, directly or indirectly required, that a person has to take in order to be qualified for a job of the given profile, according to his/her skills possessed and courses taken.**

## Development Tasks

- (1) Make an E-R diagram showing your data model. In the diagram, show the important attributes of each entity. For every relationship, show the cardinality and the participation status, as well as any important attributes associated with the relationship. Be sure to indicate the primary key of each strong entity. Note, this is a data-model E-R diagram, not a detailed relation schema diagram for implementation. I will accept hand-draw diagrams. Feel free to discuss your E-R diagram in class or during my office hours.
- (2) Reduce the data-model E-R diagram to the database relations schema. Draw a relation-schema diagram for your implementation.
- (3) Enumerate the concerned functional dependencies (to be learned in Chapter 8) in your system.
- (4) Revise the database schema design by producing a lossless-join 3NF schema that preserves functional dependencies (to be learned in Chapter 8).
- (5) Populate the tables with adequate data such that all the questions can be demonstrated. Be sure to write your INSERT statements in SQL scripts. You are also required to write a SQL script that cleans up everything in the database; this

will be a good exercise to realize the dependencies enforced by foreign key references. In your development process, you must re-populate your database after cleaning up it for multiple times.

- (6) Write the SQL statements that carry out the 25 of the 30 queries listed above. Write an interface in Java to drive each of these queries with JDBC. GUI will be appreciated.
- (7) Design and implement necessary Java classes such as Course, Job\_profile, Job, and Person with the JDBC technique. These Java classes should support creation (such as creating a course, a job profile, or a job) and deletion (such as removing a job or setting a course *inactive*.)
- (8) Design and implement the database application that manages **business processes**. Write application programs to support business **processes** such as the following:
  - a) A company accepts a new employee;
  - b) A person's job hunting;
  - c) A company finds the right people for a job profile with training;
  - d) !Evaluate the opportunities in all the business sectors for the career planning service;

### Due Dates

**Due date of Phase 1 (10%) – March 22:** The deliverables of task (1).

**Due date of Phase 2 (50%) – April 7:** The deliverables of tasks (2), (5) and (6).

**Due date of Phase 3 (40%) – April 21:** The entire project report including everything. Every group will sign up for an hour-long project inspection with me in the week of April 25. At least two test cases should be prepared and documented for queries 7 through 29, as well as for services (a) through (d) specified in Development Task (8). Each test case must include the prepared input data/setting and a specification of the expected output/results.

**Graduate Students Are Required to Complete Additional Work (CSCI 5125)** [Undergraduate students will earn extra points by working on the following requirements.]

(1) Some job profiles require one or more certificates. One can obtain a certificate by passing a set of exams provided by some companies. Most certificates can expire. Many certificates are bound to a tool. A set of certificates are potential for CSCI 4125 students including Oracle Database Developer I and Oracle Database Developer II. These two certificates bound to tool Oracle 11g. To obtain a certificate, one has to pass a number of courses. The courses are among the courses we considered previously. They could be offered either by colleges or training companies. A conceptual entity of certificate is given below.

Certificate(cer\_code, title, description, expire\_date, issued\_by, t\_code)

where cer\_code is a unique identifier, t\_code shows the bounded tools of this certificate if any, and issued\_by refers to a company.

Graduate students' implementation of the project should manage certificates. The queries regarding job profiles, jobs and skills should handle considerations of certifications.

(2) Implementation of Queries 26 and 27 are mandatory for graduate students, optional for undergraduate students for bonus points.

(3) Design and implement experiments on concurrency control using this database system. Each experiment can be demonstrated in SQLPLUS or in JDBC implementations. Every experiment must be documented with problem specification, applied solution, and presented with three test cases (with input data/setting and the expected output/results). Every graduate student must have at least four different experiments prepared and documented.