

CS 5200 Database Management Systems

Milestone 6: Conclusion & Final Presentation

Team Name: Rookies

Team Members: Jiazhen Tang, Xinmeng Zhang, Jing Wang

Project Name: Rookie

Value proposition

We set our value proposition for Rookie as is an internship listing website that provides latest internship job postings categorised by area, requirements or other attributes for the computer science major students who are unhappy with the lack of specific targeting of Indeed and Glassdoor.

We have fulfilled our value proposition. Rookie has good value propositions for every part of our website, but one that is particularly impressive is the internship job posting for computer science students. In this part, our website helps students to find the job easier and faster.

Deliverables

This semester we plan to deliver a website which requires a user to register and login to the account. Users have a username and password. Furthermore, we will provide service for students and companies log in to our website. Students can register, build their profiles, upload their existing resumes, and look for internship postings matching their skill sets. Using a database of internship listing website, companies can register, post jobs, and search job seeker profiles. Companies can view profiles of registered students and contact them, initiate an interview, or perform some other action related to their post. In addition, registered users should be able to search for jobs and filter the results based on location, required skills, salary, experience level, etc.

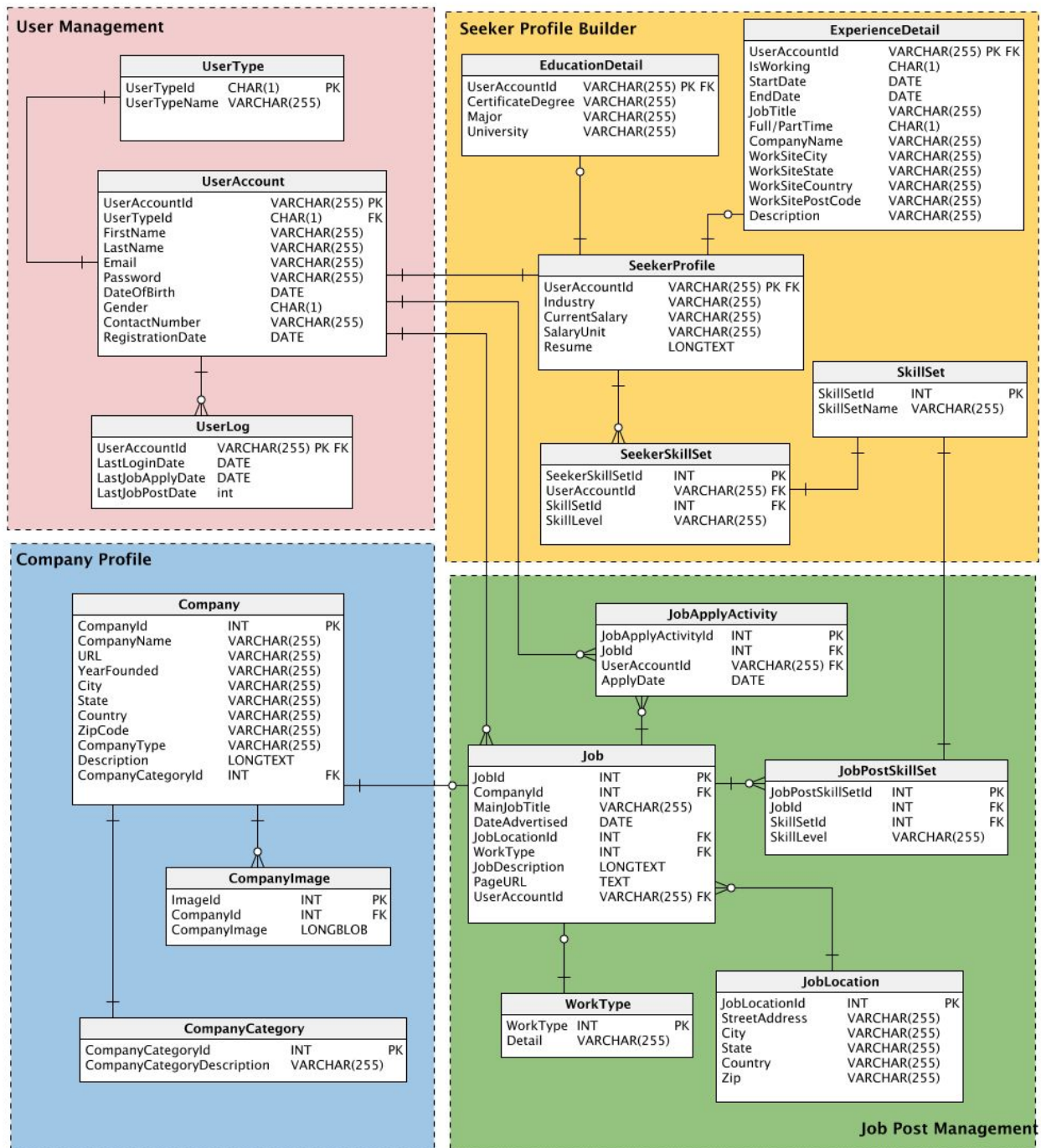
Compared to the old one, we have covered all the stuff when we plan to deliver this semester. We have user management, company profiles, seeker profiles and job post management. But we still want to add more information later. For instance, we can have a discussion board that will allow students to rate the company and provide comment. Also, a module allows users to share interview tips and experience as job rookies.

Final UML

Description of main changes:

- Changed the diagram shape from horizontal to vertical for better and clear viewing.
- Removed some unnecessary attributes.
- Changes some names of table and attributes for better understanding.
- Combining some common attributes for better management of users.

We changed our UML in PM3 when considering business insights to make the data more reasonable and manageable. Now each attribute is useful and intelligible.



Things went well

- In each milestone, the goal was clear and the approach was stated in detail on videos. So we were able to fulfill every step of the project smoothly.
- Thanks for the discussion in each module, we were able to solve some difficult configuration problems when installing MySQL and CloverDX.

- Collected data from source as csv file.
- Create an UML through teamwork using Vertabelo.
- Build a sql database for job application, inserting up to 26k rows data.
- Identify 10 questions regarding business insights and creates queries.
- Used JDBC to build the data access layer.
- Used JSP to build the web for job application.
- Created 2 ETL workflows regarding employment information for user reference.

What would we do differently?

Since we set our proposition to mainly help computer science major students to search for internships. Instead, we could also set our target users to all majors. Furthermore, instead of building our web application based on the method taught in class, we could build the generic type application and set up the Restful API server. Because our data are saved in database as resources, we could easily obtain the resources using get, put and update methods. This could be done by editing URLs to submit HTTP requests.

Future Plan

This semester we built a job searching web application including backend database, server and user interface. We build the integrated web application and could put into operation with some optimization. After this semester, we could improve our project through several aspects to solve database storage problems and server request problems.

- As the number of users on our website increases, it is impossible to run our entire database on a single server. We could build distributed database system or MySQL cluster to reduce the pressure of data storage on one machine, and avoid single point failure or security issues.
- While the number of users on our job searching website increases, the input/output requests will increase and have a high requirement for data reading speed. Therefore, we can add cache which is a in memory database with optional durability to save database reading time. To further optimize, we could add LRU cache to remove the least recently viewed job postings in the cache to save storage space. This could be implemented using LinkedHashMap.
- It is convenient to set up our servers using Amazon EC2 for hosting the website that provides grow and shrink resources. We could choose multiple types of instance with the selection of different configurations. To ensure elastic web-scale computing, EC2 could increase or decrease our capacity within seconds.
- Also, in order to reduce the server pressure of a higher QPS, we could add a load balancer which could evenly allocate requests to several backend servers and act as a proxy to hide original servers from attacks.