CS 5/7330 Fall 2021

Project – NoSQL Databases

The goal of this project is for you to get some exposure of NoSQL database by developing a small application using a NoSQL database.

Application – Research paper database

I want to build a database to keep track of research papers. Each paper has the following information that need to be stored:

- Title: a string. A paper can only have one title, and you can assume it is unique.
- Authors: a paper can have many authors. Each author should have the following information stored
    - Last name
    - First name
    - Affiliation: the name of his/her employer (a company or a school, you don't have to store the type of the affiliation). Notice that each author may change employers, so for each person, you need to record the start and end date for each employer that employ him/her. You should further assume the following:
        - Each author can have only at most one employer at any given time
        - It is possible that an author is unemployed for some period
    - Notice that if the paper have more than one author, the order of the authors need to be recorded.
- Publication: Where the paper is published. There are two types
    - Conferences: you need to record the following info
        - Name of the conference
        - The number of times it is held (e.g., 1st, 17th, etc.)
        - The year that is held
        - The location that it is held (just a string is fine)
    - Journal: you need to record the following info
        - The name of the journal
        - The year and month where the paper is published
        - Some (but not all) journal have a volume number associated with it (just a string is fine, some volume # have complicated format that you don't have to worry about in this project)
- Some optional information about each paper
    - URL where the paper can be download
    - Page number (if it is published as part of a book/magazine) (string is fine)

Task 1 – Database selection and design (25% of the project score)

You are to choose a NoSQL database system to implement a database to store the above information. You will need to design the schema that is used to store the data.

You can choose amount MongoDB, Neo4j and Redis. If you want to use another NoSQL database system you need to let me know by 10/29 (Fri) at noon. I will get back to you by 11/1 (Mon) at class time to inform me my decision.

You will need to submit a report by Monday (11/8) 11:59pm detailing your database design and schema. I will get back to each group via zoom (probably 5-10 minutes) to discuss that (hopefully within 3 days of submission).

You will be given a 10% bonus for this part if you hand in your report by 11/3 (Wed).

Task 2 – Implementation of system (75% of the project score)

You are to implement a program that allow user to enter data, as well as query information about the system. Your program can be implemented in either C, C++, Java, Python, Perl or Ruby. If you want to implement it in any other programming language, you need to get my approval (by 11/8) first.

Your system should do the following:

Data Entry:

The system will provide a GUI for the user to input papers. It may be necessary to provide places for users to input other info (such as authors, publications etc.). It should ensure the consistency of data (to the best of ability – for instance, you should assume paper title, conference name, journal name to be unique. However, many authors may have the same name).

Query

You should implement the following queries via your GUI.

- The program should get the name of a paper and return all relevant info for each paper
- The program should get the name of an author (just the name), and list of the papers for that author.
- The program should get the name of a publication, and a year range, and list of papers that is published within that range.

Bonus

There are 2 possible bonuses, each award an extra 10% to the total grade. 7000 level students are required to implement at least one of the two

Bonus 1

Implement the following query:

- The program should get the name of an author, and return the list of papers that he/she authored. However, if there is reason to believe that there are multiple authors that have the same name (e.g. they serve at different places in the same year, one should separate them. (Notice that sometime you may have to guess which paper belong to which author. It is ok, but specify how you make the guess).

Bonus 2

Implement the following query:

- The program should get the name of an author, and then return all authors that have a co-auther number of at most 3. A co-author number is defined in the following number (let the input author be A)
    - If an author co-authored with A, then the cu-author number = 0
    - If an author did not co-authered with A, but co-authored with someone that has a co-author number of 0, then his/her co-author number is 1
    - In general, if an author did not co-authered with A or anyone with a co-author number < k, but he co-authored a paper with someone with co-author number k, then his/her co-author number is k+1

Due Date

Each group should make a 10 minute presention that demo your system. It can be either done in class on either 12/1 or 12/6, or you can make a video and submit to me on or before 11/29 (Mon) at 11:59pm and I will play it during one of the two lectures I will make comments and ask you to do certain things. You will than have a few days to make final changes. The project is due 12/10 (Thur) 11:59pm.

Notice that the majority of the grade will be based on what have been achieved by the presentation. You will need to provide resources needed to run your program (except materials/code/libraries that can be downloaded from other sites). You should also provide a 2-4 page user manual, and a 2-4 page developer manual (contain enough information for other people to continue your task).