CS166 Quarter Project

MovieNet

PHASE #3: System Implementation

Professor:
Vassilis Tsotras

Teaching Assistants:
Moloud Shahbazi
Anumeha Bhasker

Department of Computer Science
University of California Riverside

*S*pring 2013

Contents

1 Introduction

In this phase you are provided a package including SQL schema and dataset. The dataset consists of data records that should be loaded into your database system. You can use Lab 5 instruction on how to use java to connect to the Postgres database and how to execute SQL queries.
You are free to change the dataset, the relational schema and the initial Java file but you have to explain it why you changed. You can also implement the project in any language that you prefer. But keep it in mind that the TA might not help you if you use a different language than JAVA.
For the remove/delete functionalities, you can either create other tables to store deleted values or flag the deleted tuples. You can also delete the tuple, but most of the times it is a better idea to store the deleted data.
Additional extra points will be awarded to projects that use GUI interfaces and user-friendly error messages.
Please follow the below steps to get started:

1.1 How to use the package provided:

- Download project.tar.gz
- Place it in the proper position and execute the following command : tar zxvf project.tgz
- You will see that a directory named project will be created. This directory contains all necessary files to get started. More specifically it contains the following things:
  - project/script → contains the scripts you will need to create your database
  - project/script/create_tables.sql → SQL script creating the database relational schema. It also includes the commands to drop these tables.
  - project/script/load_data.sql → SQL script loading the data in your tables. The script loads each text file into the appropriate table. Note that the file paths have to be changed based on your own configuration in order to make it work.
  - project/dataset → contains data for your tables. You should load these entries in the database with the copy script.

1.2 About the Client Interface in Java

If this is the first time you work with Java there is no need to be worried. A template user interface in JAVA is supplied for your lab5 assignment. You are expected to extend this basic interface and evolve it into a complete system by the addition of functionality. An Introduction to Java/JDBC can also be found in your Textbook (Sections 6.2 6.3 of Database Management Systems (Third edition)).
In this phase we basically want to create an interactive psql console for non-sql users (i.e. users of MovieNet). You should therefore pay special attention in making the interface as intuitive as possible.
The following figure shows where the Java program comes into play. It connects to the Postgres Database (which must be up and running) and from there a user may perform various SQL queries.

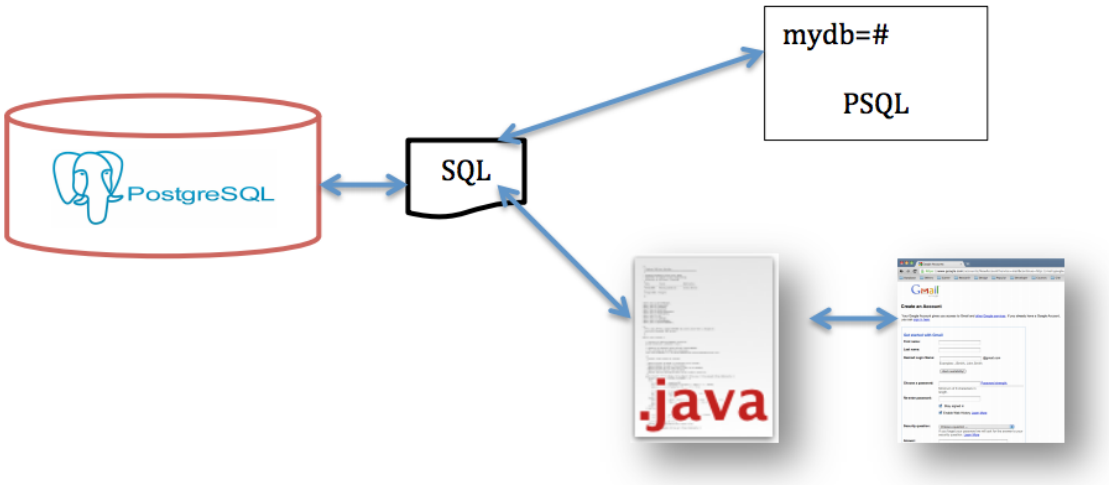

Figure 1

2 Functionalities of MovieNet

The basic requirements in terms of system functionality which has been explained in phase 1 of the project should be implemented. On top of them you may implement as many additional features as you want. Throughout the project you should keep in mind that MovieNet users are not SQL-aware. You should therefore make the user interface as intuitive as possible.

3 Extra Credit

3.1 Good User Interface

A good user interface will bring more users to your application, and also more points in this phase. A user interface is good if it is:

- Easy for users to explore features.
- Robust for exceptions, like unexpected inputs.
- Graphic interface supporting all features required.

3.2 Triggers and Stored Procedures

Instead of processing the workflow step by step, triggers and stored procedures can be used to handle a sequence of procedures. For example, when a user want to delete his/her account, the system can use trigger to delete referring records in other tables. You can also find other opportunities for triggers and stored procedures.
To submit your triggers and stored procedures with your project, please include them in the following location: project/script/triggers_sp.sql

3.3 Performance Tuning

The performance can be improved if index are used properly. In order to show the performance improvement by using index, you should defend why you chose to use an index at some particular table. For your submission, you should put the declarations of indexes in: project/script/indexes.sql

3.4 Any Other Fancy Stuff

Please feel free to show any of your ideas to improve your project! The only thing required for these will be clear documentation on them!

4 Submission Guideline

4.1 Project Report

You should provide a high level description (1-2 pages) of your implementation. You should describe in which part each of you has worked and any problems you

found along your way. In this document you should also include whatever is asked in the below individual descriptions. You should submit everything together using the iLearn System at least 1 hour before your presentation. The available slots for the presentation are between. Demo appointment slots will be online through iLearn.

4.2 Files

The following files should be submitted:

- Create Tables, Bulkload Data Scripts: if you have any modification over the schema or tables, you should include your changes into the files, and leave necessary comments for them. This includes the following files:
  - project/scripts/create_tables.sql
  - project/scripts/load_data.sql
- System Implementation User Interface: submit your source file(s). You should make sure that your code can compile and run successfully. Any special requirement for compiling and running should be stated clearly in your project report, or a readme file with your source code.
  Please include all your source code within the directory of: project/java/
- Other scripts, like triggers and store procedures, and index: You should have descriptions over these features and also all your scripts within the following directory: project/script/

5 Demo

Please send an email to your TA to schedule a 20-min demo on your system. The available slots will be posted through iLearn soon.