CS1555: Team 16 Project Report

Team 16: Alexi Green, Caleb Tucker-Loewe, Gordon Lu

Manual:

Installation and Startup

To get started:

* Go to <https://github.com/glu99331/CS1555-Term-Project>.
* Download the repository onto your local machine.  This can be done in one of the following ways::
  + Use git clone <https://github.com/glu99331/CS1555-Term-Project.git> in Git Bash in the directory you want to store the files for the application in.
  + Download the repository with a GUI like GitHub Desktop.
  + Download a ZIP containing the files for the repository and extract the files into your desired directory.

Compile Program: javac -cp .;postgresql-42.2.18.jar PittTours.java

Run Program: java -cp .;postgresql-42.2.18.jar PittTours

In the PittTours.java file, in the main method, in order to establish a connection, modify the parameters of the lines:

props.setProperty(“user”, “yourusername”);

props.setProperty(“password”, “yourpassword”);

Otherwise, a JDBC connection will not be established and the program will crash.

Initial Menu

This menu is meant to let the user distinguish what kind of user they are (Administrator or Customer).

* The user will continuously be prompted until they enter option 3 to quit.

Administrator Menu

If a user tries to access this menu, they will be asked to supply a username and password to verify that they are actually an administrator.

* The admin has the option to view more options, this will print out the customer menu. The admin will also have the option to “go back one level”, which will go to the corresponding admin options.
* The user will have the option to exit the program by entering “10” from the Administrator menu, and “15” from the Customer Menu inside of the Administrator menu.

All functions as described in the project description have been implemented to the best of our ability.

Customer Menu

Any user can access this menu, whether they are a customer that is already in the database or not.

* It is assumed that a user who enters the customer menu will not backtrack and log in as an admin.

All functions as described in the project description have been implemented to the best of our ability.

Limitations:

All functions should be implemented properly, however there could be unforeseen errors that have not been handled properly (i.e. user input, parsing errors, or concurrent transactions). In our testing, there didn’t appear to be any unhandled errors.

Triggers for corrections to the database contents upon updates, deletes, or inserts should handle all situations described in the project description properly.  However, it is difficult to test for extreme edge cases in terms of thinking of insert/update/delete combinations that the trigger would handle.

Possible Improvements:

Aside from runtime improvements, the structure of the schema could be modified so that the necessary queries in the Java Application would be simplified.

Additionally, because of the current layout of the schema, a lot of the functions, procedures, and triggers require additional overhead in order to query information from multiple tables and perform a series of calculations in order to function properly.  If the schema were a bit more condensed (i.e., some of the attributes from different tables could be merged together into a single table, or, at the very least, some tables could contain a few additional foreign keys referencing other tables), we might be able to eliminate some of that overhead.