FoodWasteReducer

Rodica-Elena Roșca

Faculty of Computer Science,
"Alexandru Ioan Cuza" University,
General Berthelot 16, 700483 Iași, Romania
https://www.info.uaic.ro

Abstract. FoodWasteReducer is a minimal project which implements the exchange of products between restaurants and people. This implementation uses Transmission Control Protocol for transmitting data between the server and the client(s).

1 Introduction

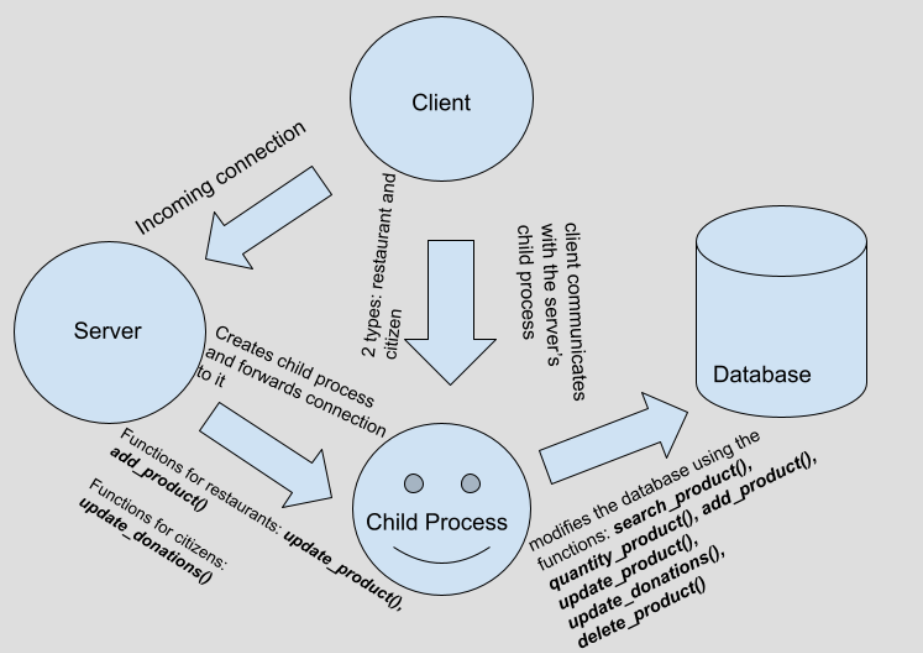
FoodWasteReducer is a minimal implementation of a food waste reducer system. It has the ability to ask clients which type of client they are and, depending on this, to request a certain product and a certain quantity of that product if it is possible. This is the case when the client is a normal person. The case when the client is a restaurant implies another options like donating a certain product and a certain quantity of that product.

Being implemented concurrently, it can handle multiple clients at the same time. Everything about network communication will be presented in the **Technologies Used**. Also, in the **Program Arhitecture** I will show you a diagram which should make you a clear vision about the entire project. After seeing all of this, I will present to you the details of the client and the server implementation in the **Implementation Details** section. In the end, all the conclusions will be reflected in the **Conclusions** section.

2 Technologies Used

Transmission Control Protocol(TCP) is the protocol used for transmitting data between the server and the client(s). **FoodWasteReducer** uses a connection-oriented model for communication over the network. The **Transmission Control Protocol(TCP)** was the one that I choosed because I can not affort any lost of data. The project is designed to continuously interact with a `database` and the modifications made on the database must be done in the correct order and with the correct parameters. The server program uses *multiprocessing* for handling connections with multiple clients and it makes **TCP** the best option. For the database I choosed **SQLite**. It is an open-source relational database engine and it has all I need for this `project`.

3 Program Arhitecture



4 Implementation Details

The project has two executable binary programs: *the server* and *the client*. In the next two subsections, both the server and the client will be analyzed.

4.1 The Server

The Server is responsible with dealing with multiple clients. *The Server* does the following
requirements:

- *Multiprocessing* for delegating the responsibility of handling the connection with a client to a child process

- *Options* for choosing to donate or to ask for a certain product/quantity. For this work,
*search_product(char *, sqlite3 *)* provides information about finding or not the product in the database,
*quantity_product(char *, sqlite3 *)* provides information about finding or not the quantity of a certain product in the database. The function *add_product(char *, int, sqlite3 *)* takes the product from the restaurant and modifies the database by inserting the information if the product does not exist and *update_product(char*

**, int, sqlite3 *)* updates the database using UPDATE command if the product already exists. If the client asks for a product and the given quantity is maximum possible, the function *delete_product(char* , sqlite3 *)* deletes the product from the database, otherwise, if the given quantity is less then the maximum quantity that is found in the database, we use the function *update_donations(char *aliment, int cantitate, sqlite3 *db)*.

- *I/O operations* for receiving the client's options and modifying the database

4.2 The Client

The Client is responsible with the initiation of the connection with the server. *The Client* does the following requirements:

- *I/O operations* which will be transmitted to the server via a socket descriptor and, depending on the case, they can modify or not the database

5 Conclusions

The FoodWasteReducer project provides the basic features needed for connecting restaurants with people in order to reduce the food waste. The potential improvements of the proposed solution would be to add a guide for the ones who are on a diet and will give them `suggestions`. It could also create a new table for each category of products.

References

SQLite : *How to integrate SQLite code in C program?*
`https://zetcode.com/db/sqlitec/`

Conccurency : *How to create a conccurent Client-Server implementation?*
`https://profs.info.uaic.ro/~gcalancea/lab7/servTcpConc.c`

SQLite : *SQL Injection*
`https://stackoverflow.com/questions/6646731/how-does-sprintf-protect-against-sql-injection`