Final Project Documentation: BDII

Repository: https://github.com/JeanPaulRF/Whisky

1st Daniel Sequeira Retana
Escuela de ingeniería en computación(of Aff.)
ITCR (of Aff.)
San José, Costa Rica
danielsretana@gmail.com

2nd Jean Paul Rodriguez Flores
Escuela de Ingenieria en Computacion (of Aff.)
ITCR (of Aff.)
San Jose, Puriscal
jeanpaulrf2@gmail.com

***Abstract*—This document is textual documentation of a final database project about a website to sell whisky. It composed of three different stores that have its own inventory and that any user can order on them.**

I. Introduction

A. Definition of important concepts

- Spatial data: Comprise the relative geographic information about the earth and its features.
- Encryption: It is the process of encoding information.
- Decryption: The conversion of encrypted data into its original form
- Aged: Do something old. In wine and some foods it implies its alteration over time, either to improve or deteriorate.
- Back-end: Logical area of every web page. We refer to the internal architecture of the site that ensures that all elements perform the correct function. It is not visible to the user and does not include any type of graphic element.
- Front-end: It is the part that the user sees and in which the design line and the graphic elements of the page are included, unlike in the Back-End.

B. Topic context

Some clients want to implement a website for all Whiskey fans for the countries of Scotland, the United States and Ireland. They want a 24/7 system, in which the server is always available and any customer can make a purchase or a subscription to the page at any time. They have decided to hire us as the company in charge of developing said program. After numerous meetings to gather requirements, the clients concluded the functionalities and general restrictions that they want.

C. Who is it for and why?

This project is aimed at all fans of whiskey in the aforementioned countries, so that they can consult the products available for different concepts in stores and thus buy them and/or make a subscription.

II. Methods

A. Proposal description

To meet the needs of the client, a unified web page will be developed, which has access to the central database and to the stores, where the different queries will be made at the user's request. You will access procedures located in the different databases, corresponding to the tables located in them. The user will be able to enter with an encrypted password, he will be able to see the products available in the stores and on the page in general, he will be able to consult different areas, he will be able to make a subscription that grants different benefits within the site. You can also make purchases of products, as well as request the shipment of these, review the products and workers.

B. Used tools

- Hackolade
- Sql Server
- Postgresql
- Git Kraken
- GitHub
- Overleaf
- Browsers
- Word
- PHP
- Javascript
- CSS
- XAMPP

C. Application architecture

The software consist in a web page divided in two parts: the store and the administration. Both sites work with 2 data base engines, SQL server where are the stores and everything relate and Postgres where is the master data base who manage all the administrative and sales part.

Based on the requisites, the Store has multiple interfaces, and the user interact depending of what he wants to do, starting with sign in, to view reviews or buying his own whisky. The administrative part is created for high-level users, in this section the user is able to work directly with the database using the cruds and other consults with an exclusive interface for each function.

The data bases have multiple tables related to each other with foreign keys, and stored procedures to make basic and advanced functions, this procedures are used work with the web page to make strong insert thanks to the validations that they have.

The web page is highly user friendly, with a modern design and easy to navigate inside him. Multiple languages and similar technologies are working together to bring to the user a good experience.

D. Database desing

As requested by the client, an independent database was created to store the products, sales and workers; along with their respective type tables. In the same way, in this database there are the necessary tables to store the deliveries, the reviews of the products and the workers ones, as well as another to respond to any complaint presented in the worker's qualification.

For the stores, a database was created for each one of them, where the client, user and subscription information is stored; these with their respective types tables. As well as to store store inventory.

For each database, the necessary procedures or functions were elaborated to show all the requested queries, as well as the realization of the CRUD (Create, Read, Update and Delete) of tables such as product, worker, subscription... Also to carry out the log in for each user and that they can make purchases. Also to be able to present the reviews of products and workers, see the response of these. As well to see the delivery of the product.

- Master DB Design: https://acortar.link/zDN8Bs
- Store DB Desing: https://acortar.link/hBinag

III. Analysis of results

A. Problems found

When you install Postgres it use your current keyboard language to config the COLLATE and the CTYPE, you can't create a database with differente collate and ctype if you have preconfigured Postgres with a different encoding, the participants of this project were using en-US.UTF-8 and Spanish-Costa Rica.1252 as collate and ctype so to solve the different configuration of postgres it was necessary to use a template to create the data base.

A common problem in the development of the project was the connection to the databases, mainly for the different credentials between the developers, creating the necessity of changing the string connection depending who is working with the database.

Another problem was an intern error of SQL Server making impossible to use openquery in stored procedures, so making a consult of Postgres in SQL Server was not possible, to solve this the table of sales was move to Postgres, making it unnecessary to use openquery.

B. What was achieved and what was not?

Starting for the achieved, we were capable of creating a good documentation, with all the required for the correct description of the project, the clean code was present for all the project, with good practices and efficient functions to complete their goals. Everything related with the first segment of the evaluation was completed.

For the second part, the inventory, the CRUDs, the billing and the consults are working perfectly in back end and front end

It was not possible to make the function to send emails when new product enters the system

IV. Conclusion

Applying all the knowledge acquired throughout the semester was a great exercise to test everything learned, its difficulty forced us to review what was not clear at first, making this time well learned to deliver a project of good quality. On the subject of whiskey, it was very entertaining to learn about this curious drink, it is incredible all the culture that it has and that was ignored.

Making a web page will always help to learn to improve the programing skills, since it takes many languages and technologies behind to work in a single system, it can be concluded that the project fulfilled the objective of applying the topics and teaching new ones for the training of students according to current market needs

V. Bibliography

- https://docs.microsoft.com/en-us/sql/relational-databases/spatial/spatial-data-sql-server?view=sql-server-ver16
- https://www.sqlshack.com/an-overview-of-the-column-level-sql-server-encryption/
- https://www.postgresqltutorial.com/postgresql-date-functions/postgresql-now/
- https://www.youtube.com/watch?v=YpRiAk5oAVA