

2020

Gestor Multiple de Bases de Datos



Pablo Brañuelas Castellano

08/06/2020

PROYECTO FIN DAM

INDICE

1. Memoria:
 - a. Diseño de la aplicación.
 - b. Funcionamiento de la aplicación.
 - c. Seguridad.
 - d. Diseño.
2. Líneas de Investigación.
3. Accesibilidad.
4. Bibliografía.

1. MEMORIA

a) Diseño de la aplicación



El diseño está basado en contenedores y cajas, usando un sistema de GridLayout.

La idea es que se pueda ver fácilmente las bases de datos y al clicar sobre ellas aparezcan sus tablas y vistas.

Mientras que en el lado derecho es donde se realizan las consultas y donde se muestran los resultados obtenidos y también el número de filas obtenidas.

En la parte superior hay un menú donde se pueden establecer la conexión a los diferentes tipos de bases de datos (MySQL, MariaDB, SQLITE, Firebird, SQL Server, PostgreSQL), consultar la sintaxis de cada sistema de base de datos, dar formato al texto, exportar e importar bases de datos, guardar los resultados de las consultas realizadas, cargar archivos para mostrar gráficos de datos, actualizar y vaciar los diferentes contenedores, una conexión a un Web Service escrito en Java Spring y una opción de salir de la aplicación.

Las versiones utilizadas en este proyecto son:

- Python 3.8.2.
- MySQL 8.0.19.
- MariaDB 10.4.
- SQLITE 3.
- Dremio 3.3.0.
- Firebird 3.0.
- SQL SERVER 2017.
- PostgreSQL 11.7.

b)Funcionamiento de la aplicación

La aplicación está basada en los diferentes gestores de base de datos que existen; por ejemplo DBeaver.

La aplicación te permite conectarte a distintos gestores de bases de datos como MySQL, MariaDB, SQLITE, Firebird, SQL Server, PostgreSQL ,Dremio.

Algunas de las funcionalidades que incluye son:

- Exportar e importar bases de datos de cualquier gestor usado en la aplicación.
- Guardar las consultas realizadas en distintos formatos: CSV, PDF, HTML.
- Cargar archivos csv y xls para obtener gráficos a partir de la información que contienen.
- Refrescar bases de datos y tablas.
- Dar estilo a las distintas ventanas cambiando el tamaño, tipo y color de la letra además del fondo.
- Sistema de ayudas saber la sintaxis correcta a usar dependiendo del gestor que estés utilizando.

- Api de Google para saber cuántas veces se ha buscado una palabra en cada región.
- Palabras más buscadas al año.
- Web Service escrito en Java y Spring ,al cual se llama desde cada una de las opciones escritas en Python: Crear Usuario, Ver todos los Usuarios, Buscar Usuario, Actualizar Usuario y Eliminar Usuario.

c) Seguridad

Para poder acceder a las bases de datos y sus diferentes opciones necesitas haber iniciado sesión.

Para ello hay una ventana de inicio de sesión conectada a distintas librerías de python, para verificar el usuario y contraseña.

A nivel interno cada gestor tiene su propia forma de controlar que puede hacer y que permisos tiene cada usuario a través de roles y permisos.

2. Línea de Investigación

Para realizar el proyecto he tenido que aprender:

1. Como conectar los distintos gestores de bases de datos y que no entren en conflicto al usarlos en otros sistemas operativos como Linux.
2. Librerías de Python que permitan guardar las consultas en distintos formatos.
3. Librerías de Python para poder leer los archivos csv y xls y poder mostrar sus columnas y dibujar los gráficos.
4. Las características propias y la sintaxis de cada uno de los gestores utilizados y como realizar por línea de comandos las exportaciones y las importaciones de las bases de datos.
5. Librería para usar Google Trends con Python.
6. Como conectar y consumir un Web Service escrito en Java Y Spring desde Python .
7. Distintas opciones de consultas sobre archivos csv y xls que permite Dremio a través de Azure Storage y creación de tablas en Dremio.

8. En futuras ampliaciones se puede añadir:
- a. MongoDB.
 - b. Apache Cassandra.
 - c. TimeScaleDB.
 - d. Apache Hadoop.
 - e. DB2.
 - f. Oracle.
 - g. Dremio Ampliado con Azure Storage.

3. Accesibilidad

Para mejorar la accesibilidad en la aplicación he añadido una opción para que cada contenedor pueda configurar el estilo que el usuario necesite.

Además de agregar dos configuraciones específicas para personas con problemas de visión y daltonismo.

Los criterios seguidos son:

- Para personas con problemas de visión según los criterios de accesibilidad tiene que usarse:
 - Alto contraste. Para ello he establecido como fondo el color negro y color de fuente el verde.
 - Tipo de fuente y tamaño de letra. No hay un criterio preestablecido. He optado por Arial y tamaño 30 pixeles.

- Para personas con daltonismo el criterio seguido es buscar características comunes que no afecten a los distintos tipos de daltonismo.

Para ello he usado la web

<https://www.toptal.com/designers/colorfilter> donde te filtra según tipo de daltonismo y te muestra como lo vería cada persona los colores.

- Color de Fondo. El elegido es “Sky blue”. Ya que las personas con daltonismo si ven ese color (salvo aquellas que sufren un daltonismo acromático).
- Fuente de la letra. Como no hay un criterio preestablecido he escogido Arial 20 píxeles.
- Color de Letra. Bajo el mismo criterio del color de fondo; es decir un color que sea fácilmente visible y buscando un buen contraste, para facilitarles la visión he escogido el color negro.

4. BIBLIOGRAFÍA

- **MYSQL**

- Conectar Python con Mysql:
<https://unipython.com/acceso-a-bases-de-datos-con-pyodbc/>

- Documentación:
<https://www.w3schools.com/sql/default.asp>
<https://dev.mysql.com/doc/refman/8.0/en/>

- **MARIADB**

- Conectar Python con MariaDB:
<https://unipython.com/acceso-a-bases-de-datos-con-pyodbc/>
Lo único que cambia es el puerto Mysql escucha en el 3306 .MariaDB en el 3310

- Documentación:
<https://mariadb.com/kb/en/documentation/>

- **SQLITE**

- Conectar Python con SQLITE:
<https://pynative.com/python-sqlite/>
- Documentación:
https://www.sqlite.org/keyword_index.html
<https://www.sqlite.org/docs.html>

- **FIREBIRD**

- Conectar python con Firebird:
https://firebirdsql.org/file/documentation/drivers_documentation/python/fdb/getting-started.html
- Documentación:
 - Crear bases de datos:
<https://www.firebirdsql.org/manual/es/qsg15-es-creating.html>
 - Agregar usuarios :
 - <https://firebird21.wordpress.com/2015/07/11/agregando-el-usuario-sysdba-en-firebird-3/>

- <https://firebird21.wordpress.com/2013/04/21/agregando-modificando-y-borrando-usuarios/>
- Mostrar tablas y vistas:
<https://firebird21.wordpress.com/2013/03/04/nombres-de-todas-las-tablas-y-de-todas-las-vistas/>
- Estructura de una tabla:
<https://firebird21.wordpress.com/2013/03/04/nombres-de-todas-las-columnas-de-una-tabla/>

- **SQL SERVER**

- Conectar Python con SQL SERVER:
<https://docs.microsoft.com/es-es/sql/connect/python/pyodbc/step-3-proof-of-concept-connecting-to-sql-using-pyodbc?view=sql-server-ver15>
- Documentación:
<https://docs.microsoft.com/es-es/sql/sql-server/?view=sql-server-ver15>
- Realizar copias de seguridad:
<https://docs.microsoft.com/es-es/sql/linux/sql->

[server-linux-backup-and-restore-database?view=sql-server-2017](https://www.server-linux-backup-and-restore-database?view=sql-server-2017)

- **POSTGRESQL**

- Conectar Python con PostgreSQL:
<https://www.neoguias.com/como-conectarse-postgresql-python/>
- Documentación: <https://www.postgresql.org/docs/>
 - Ver Bases de datos:
<http://www.postgresqltutorial.com/postgresql-show-databases/>
 - Mostrar tablas y vistas:
<http://www.postgresqltutorial.com/postgresql-show-tables/>
 - Mostrar campos de una tabla:
<http://www.postgresqltutorial.com/postgresql-describe-table/>

- **DREMIO**

- Conectar Dremio con Python:
<https://docs.dremio.com/client-applications/python.html>
- Documentación: <https://docs.dremio.com/>
 - Crear Tablas:
<https://docs.dremio.com/sql-reference/sql-commands/tables.html>
 - Cargar archivos csv y xls y realizar consultas SQL desde la interfaz web:
<https://www.dremio.com/tutorials/visualizing-your-first-dataset-with-tableau-and-dremio/>

- **ACCESIBILIDAD**

- Baja Visión:
<http://accesibilidadweb.dlsi.ua.es/?menu=deficit-visual-baja-vision>
- Daltonismo:
<https://cuidatuvista.com/daltonismo-problemas-vision-de-colores-rojo-verde/>

- Web Efecto Daltonismo:

<https://www.toptal.com/designers/colorfilter>

- Mejorar Accesibilidad para daltónicos:

<http://accesibilidadweb.dlsi.ua.es/?menu=deficit-visual-daltonismo>