Instituto Tecnológico de Costa Rica

Unidad de Computación

“Proyecto 2 Bases de Datos II”

Marco Antonio Espinoza Salas

Allen Antonio Jiménez González

Lester Alexander Trejos Bermúdez

Sede San Carlos

19/11/2020

**Tabla de contenidos**

Contenido

[Introducción 3](#_Toc56639181)

[Análisis del problema 4](#_Toc56639182)

[Requerimiento #1: 4](#_Toc56639183)

[Requerimiento #2: 4](#_Toc56639184)

[Requerimiento #3: 4](#_Toc56639185)

[Requerimiento #4: 4](#_Toc56639186)

[Requerimiento #5: 5](#_Toc56639187)

[Requerimiento #6: 5](#_Toc56639188)

[Solución del problema 6](#_Toc56639189)

[Solución al requerimiento #1: 6](#_Toc56639190)

[Solución al requerimiento #2: 7](#_Toc56639191)

[Solución al requerimiento #3: 8](#_Toc56639192)

[Solución al requerimiento #4: 8](#_Toc56639193)

[Solución al requerimiento #5: 8](#_Toc56639194)

[Solución al requerimiento #6: 8](#_Toc56639195)

[Análisis de resultados 9](#_Toc56639196)

[Análisis de resultados del primer requerimiento: 10](#_Toc56639197)

[Análisis de resultados del segundo requerimiento: 10](#_Toc56639198)

[Análisis de resultados del tercer requerimiento: 11](#_Toc56639199)

[Análisis de resultados del cuarto requerimiento: 12](#_Toc56639200)

[13](#_Toc56639201)

[Análisis de resultados del quinto requerimiento: 13](#_Toc56639202)

[Análisis de resultados del sexto requerimiento: 14](#_Toc56639203)

[Conclusiones 15](#_Toc56639204)

[Recomendaciones 16](#_Toc56639205)

[Referencias Bibliográficas 17](#_Toc56639206)

## Introducción

El propósito de este informe es presentar las experiencias obtenidas en la realización del proyecto, que consiste en el desarrollo de una aplicación de escritorio, utilizando como fuente una base de datos remota o creada en el servidor local. Además, en este informe se encuentra dividido de la siguiente forma: en la sección requerimientos solicitados (Análisis del Problema) están citados punto por punto, esto para aclarar las minutas en las que se dividirá el proyecto. A su vez, la sección de las soluciones muestra la forma y el proceso en el cual se resolvió cada una de las minutas mencionadas en la sección de requerimientos solicitados. Por otra parte, en la sección de análisis, se muestran y explican los resultados obtenidos al ejecutar las respectivas estrategias planteadas en la sección de soluciones. Como antepenúltima sección, tenemos las recomendaciones, estas nos indican acciones de valiosas para un mejor desarrollo de una aplicación similar a esta en un futuro. De penúltima sección tenemos las conclusiones, las cuales dictan los puntos más significativos del proyecto, y por última sección tenemos las referencias bibliográficas, las que permite reconocer el trabajo citado a sus respectivos autores.

Este proyecto se realizó utilizando herramientas informáticas, principalmente el lenguaje de programación Python (MSSQL para la base de datos com motor).

Explicando el desarrollo del proyecto, para su creación se utilizó Python y ciertas librerías como Tkinter y las que dispone Python para hacer conexiones a distintos motores de base de datos, se comienza con la lectura de los requisitos del proyecto para su posterior desarrollo. Después de establecer los requisitos del proyecto, se comenzó con la codificación de la aplicación, al final de la codificación se realiza una revisión y corrección de los errores presentes en el programa, para que el proyecto sea fluido y no tenga complicaciones para la hora de usarlo.

Un proyecto como este es muy necesario para aprender a administrar sus datos y/o aplicaciones optimizar el tiempo y administrar mejor sus actividades, además, ver los privilegios en las bases de datos, qué se puede hacer y qué no. También, ver cómo se ejecutan las consultas en las bases de datos; esto se simboliza en la sección del plan de ejecución.

## Análisis del problema

Preliminarmente, se conoce que el proyecto consiste en elaborar una aplicación de escritorio, la cual debe cumplir una serie de requerimientos y que esta, debe hacer uso de la base de datos desarrollada en modo local o remoto.

El siguiente apartado tratará punto por punto los requerimientos solicitados para la realización de este proyecto, consiguiendo así, una explicación más detallada y amplia de lo que el proyecto consiste.

### Requerimiento #1:

Como primer requerimiento , el sistema debe permitir el ingreso de parámetros para conectarse a cualquier base de datos remota (Servidor, puerto, base de datos, usuario y clave). Al conectarse de deben controlar las excepciones y de tener una conexión favorable se facilita el acceso a los dos módulos a desarrollar: monitor de privilegios y despliegue de planes de ejecución.

### Requerimiento #2:

Como segundo requerimiento, se desplegarán las tablas y atributos para cada una de estas con su respectivo nivel de privilegios.

### Requerimiento #3:

Como tercer requerimiento, el usuario puede seleccionar la tabla y ver sus privilegios, o bien expandir los atributos de una tabla determinada y ver su nivel de privilegios para cada atributo de forma individual.

### Requerimiento #4:

Como cuarto requerimiento, el usuario podrá ingresar un query para la base de datos a la que se encuentre conectado de modo que el sistema retorne el plan de ejecución definido por el motor de base de datos. El plan de ejecución deberá ser retornado en JSON o XML para que pueda ser desplegado en un árbol de propiedades gráfico para que sea simple su interpretación y visualización.

### Requerimiento #5:

Como quinto requerimiento, el usuario puede definir mostrar el plan simple o el plan con detalles, en el caso de postgres basta con agregar “verbose” al comando y en el caso de SQL server deberán filtrar los datos más relevantes. El usuario puede determinar si ve el plan de ejecución estimado o el plan de ejecución real (postgreSQL - analize, SQL Server - statistics).

### Requerimiento #6:

Como sexto requerimiento, se debe mostrar un resumen de uso de índices en la consulta. Este resumen se calcula según el plan de ejecución del script, y debe mostrar el uso o no de los índices existentes y asociados a cada tabla o vista del script evaluado.

## Solución del problema

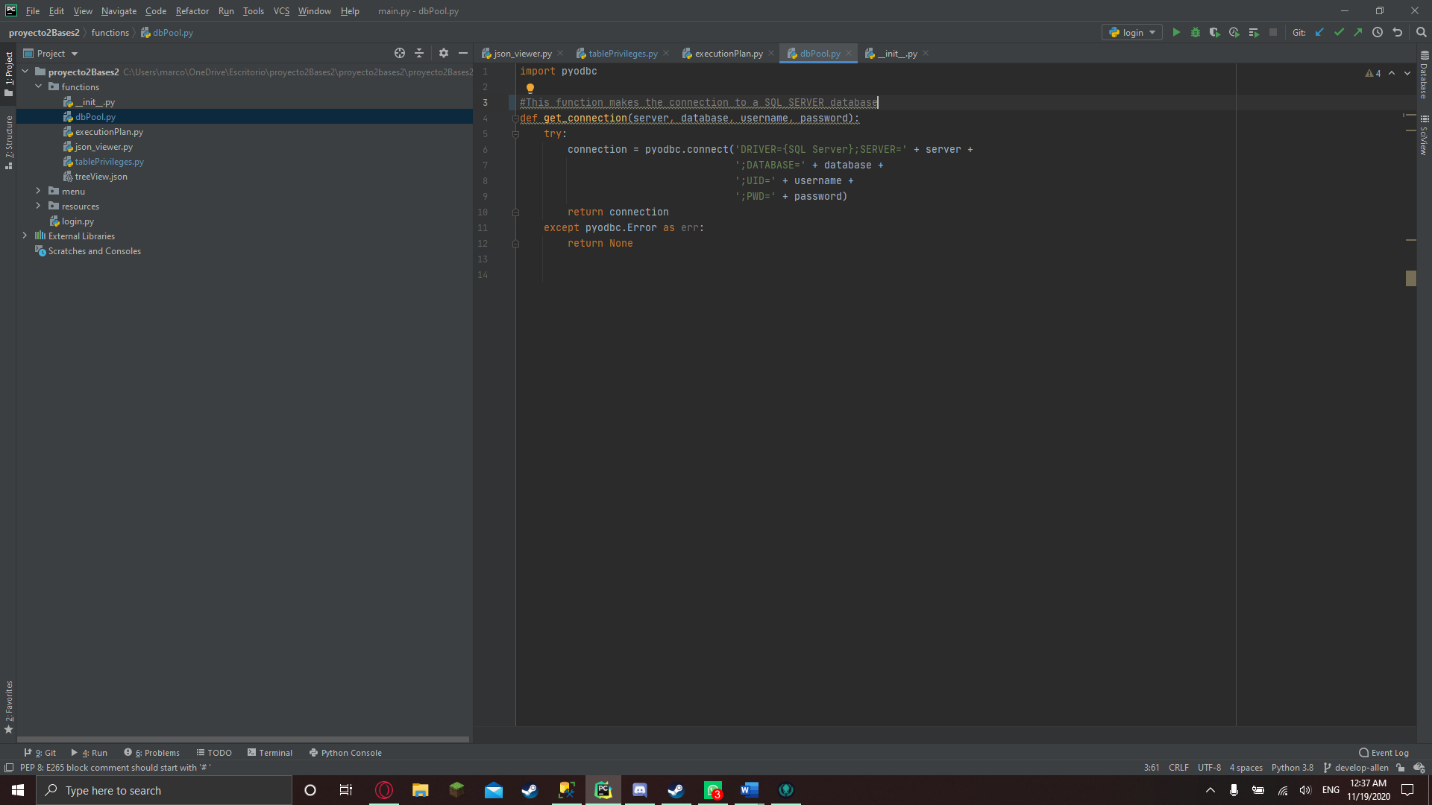
Para mantener una consistencia en el formato del presente informe, en esta sección se explicará la solución a cada requerimiento ejemplificado en el apartado anterior.

### Solución al requerimiento #1:



Imagen#1: Librería Tkinter. Fuente: <https://iot4beginners.com/data-and-classes-of-python-gui/>

Basándonos en facilidad de uso y con experiencia previa, se decidió utilizar la librería Tkinter[[1]](#footnote-1) para hacer la parte de la GUI de la aplicación.

Para lograr esto, con el uso de un editor de texto para Python; Atom y PyCharm [[2]](#footnote-2)en nuestro caso, entonces, con el entorno de programación que brindan estos editores (instalando IDES), se realizó la creación de la aplicación que representa la comunicación a base de datos, con la finalidad de poder ejecutar los procedimientos almacenados en la base de datos por medio de la aplicación de escritorio, ya sea para ver los privilegios de las diferentes tablas y/o atributos, así como el plan de ejecución.

Imagen#2: Ejemplo de conexión a base datos. Fuente: Elaboración propia

Para la conexión a la base de datos, se utilizó la librería Pyodbc [[3]](#footnote-3)de Python para hacer conexiones a distintas bases de datos y a distintos motores de base de datos, en este caso nos centramos más en el motor de base de datos MSSQL.

### Solución al requerimiento #2:

Para la solución a este requerimiento, se procede con la creación de una serie de combox, los cuales extraen los esquemas de base de datos, una vez elegido el esquema, se llena otro con las tablas, y una vez que se elige la tabla, se llena otro con los atributos, automáticamente se ejecuta esta funcionalidad. Luego con una serie de botones se puede visualizar el nivel de privilegios de las tablas y también, los privilegios de los atributos de la tabla elegida.

### Solución al requerimiento #3:

Para la solución a este requerimiento, se complementa con el requisito número 2, y una vez que se despliega todas las tablas y los atributos de las tablas, se hace uso de un botón para extraer la información y mostrarla en un campo de texto en la aplicación.

### Solución al requerimiento #4:

Para solucionar este requerimiento, se crea un campo de texto en el cual el usuario puede escribir la consulta que desee a la base de datos, después se hará uso de un botón el cual muestra el plan de ejecución para dicha consulta.

### Solución al requerimiento #5:

Se toma en cuenta que el usuario puede elegir entre ver el plan simple de ejecución o el plan con detalles de la ejecución de un query específico. Y si ve el plan de ejecución estimado o el plan de ejecución real

### Solución al requerimiento #6:

Se muestra el resumen de uso de índice en un campo de texto como en la sección de los privilegios de la tabla y de los atributos, y se calcula con respecto a los lineamientos solicitados en el requerimiento número 6.

## Análisis de resultados

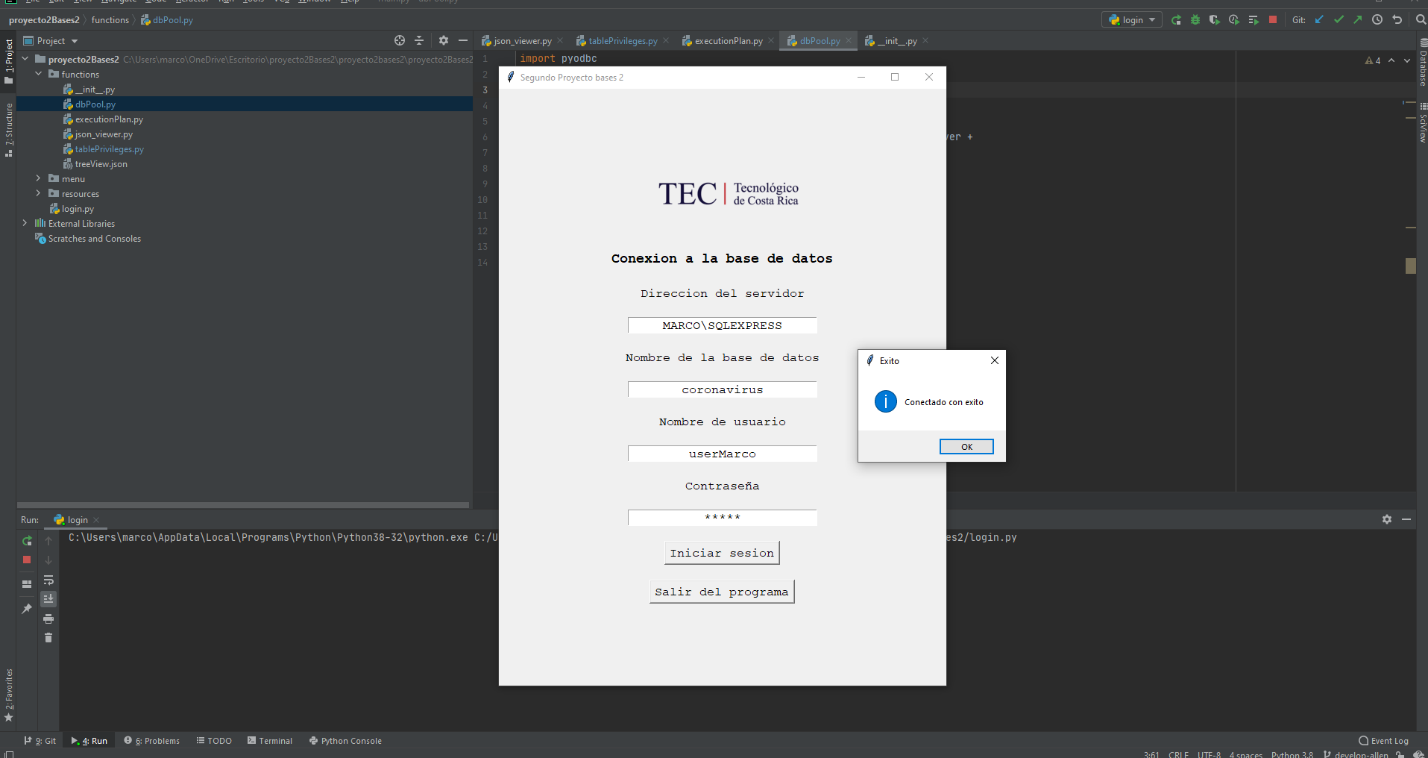
En esta sección se llevará a cabo un análisis de los resultados obtenidos a lo largo de creación del proyecto.

|  |  |  |
| --- | --- | --- |
| Tarea/Requerimiento | Estado | Observaciones |
| Conexión a base datos. | Completo | Ninguna |
| Se desplegarán las tablas y atributos para cada una de estas con su respectivo nivel de privilegios. | Completo | Ninguna |
| El usuario puede seleccionar la tabla y ver sus privilegios, o bien expandir los atributos de una tabla determinada. | Completo | Ninguna |
| El usuario podrá ingresar un query para la base de datos a la que se encuentre conectado de modo que el sistema retorne el plan de ejecución definido por el motor de base de datos. | Completo | Ninguna |
| El usuario puede definir mostrar el plan simple o el plan con detalles, también si ve el plan de ejecución estimado o el plan de ejecución real. | Completo | Ninguna |
| Se debe mostrar un resumen de uso de índices en la consulta. | Completo | Ninguna |

Tabla#1: Cotejo de los requerimientos, sus estados y observaciones. Fuente: Elaboración propia

En relación con la “Tabla#1” se destaca punto por punto los requerimientos del proyecto. En el siguiente apartado, se mostrará los resultados obtenidos de estos puntos para así, justificar el estado de estos requerimientos.

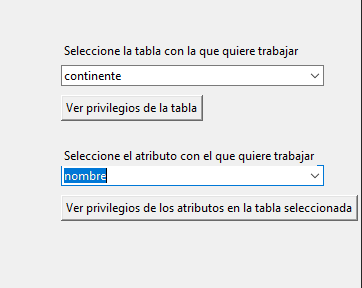
### Análisis de resultados del primer requerimiento:

De este requerimiento se obtuvieron resultados alentadores, ya que, se realizó una conexión exitosa a las bases de datos.

Imagen#3: Mensaje de conexión exitosa. Fuente: Elaboración propia.

### Análisis de resultados del segundo requerimiento:

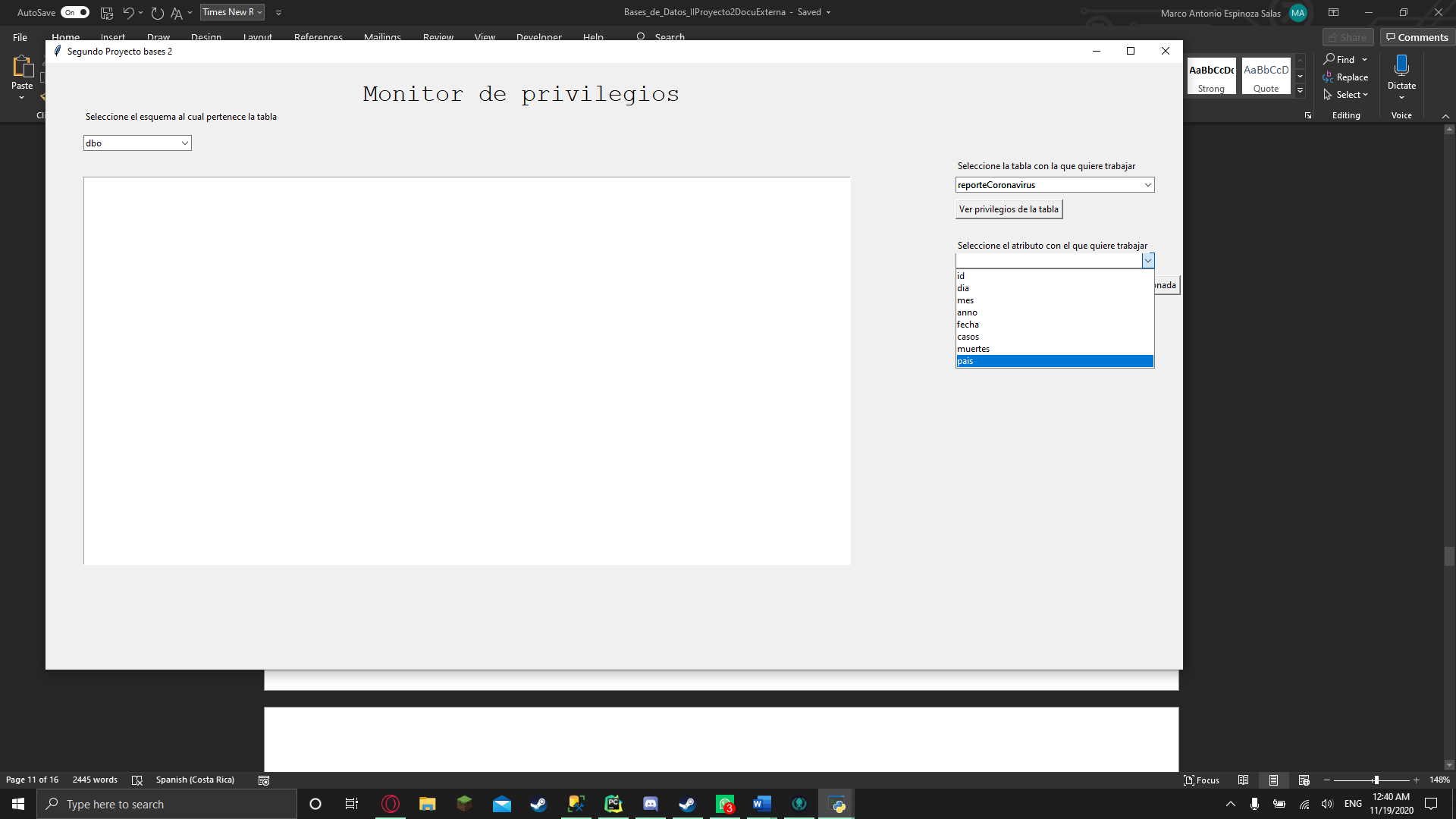
Como parte del despliegue de los esquemas, de las tablas y de los atributos se obtuvieron resultados positivos, ya que tanto los esquemas, las tablas y los atributos se despliegan de la manera óptima, demostrando la efectividad y eficacia de las funciones creadas.



Imagen#4: Ejemplo de despliegue de tablas y atributos. Fuente: Elaboración propia.

### Análisis de resultados del tercer requerimiento:

Para hacer la selección de tablas y de atributos, se tomó en cuenta la creación de dos “comboBox”, uno para cada valor el cual, si se elige una tabla específica el “comboBox” de los atributos se actualiza con los valores (atributos) correspondientes a esa tabla.



Imagen#5: Ejemplo de despliegue de atributos. Fuente: Elaboración propia.

### Análisis de resultados del cuarto requerimiento:

La aplicación permite el ingreso de un texto tipo “query” para hacer la consulta a la base de datos.

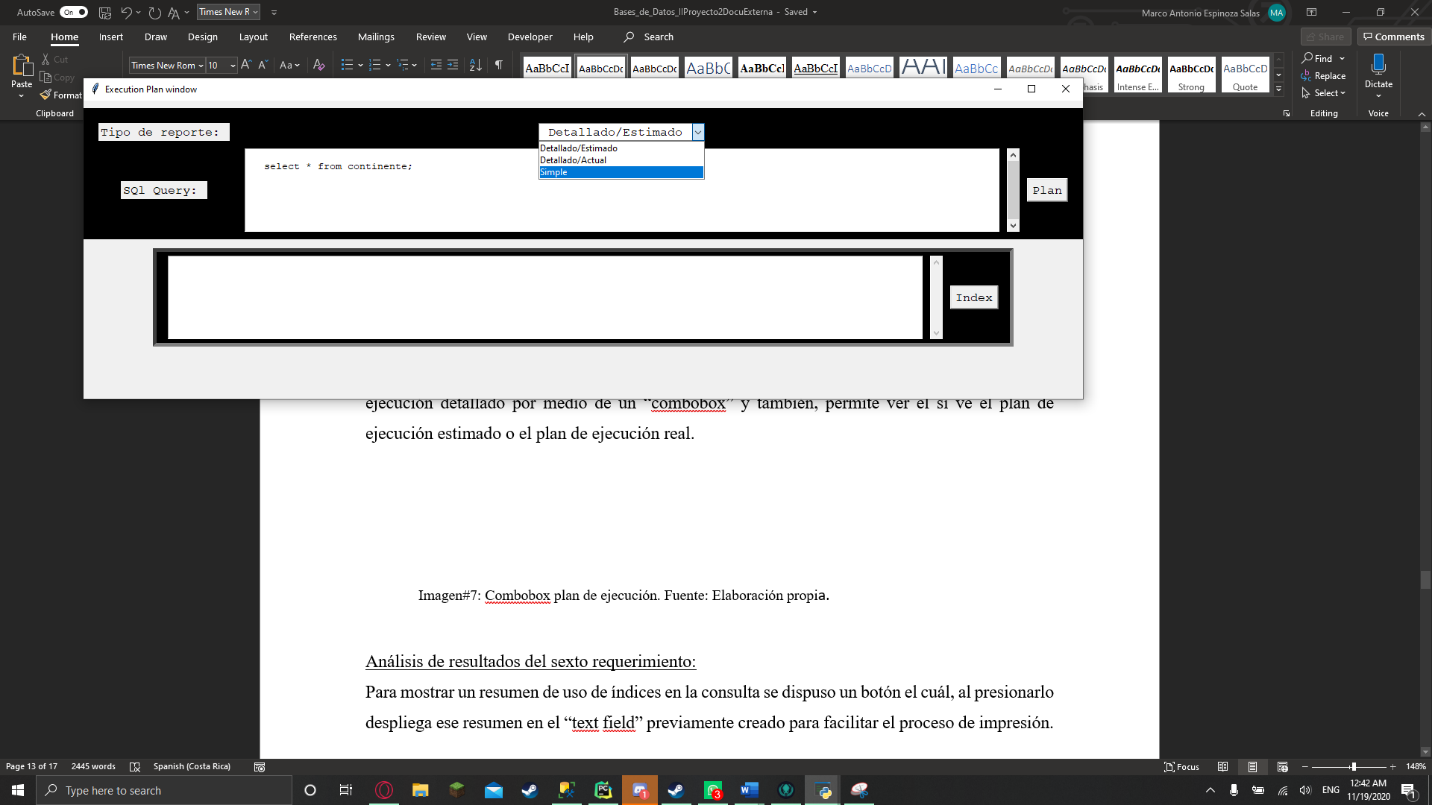
Para esto se creó un “text field” con Tkinter para obtener ese query desde ese componente específico y facilitar la consulta.

### 

Imagen#6: Ingreso de query. Fuente: Elaboración propia.

### Análisis de resultados del quinto requerimiento:

La aplicación permite elegir si se quiere visualizar el plan de ejecución simple o el plan de ejecución detallado por medio de un “combobox” y también, permite ver el si ve el plan de ejecución estimado o el plan de ejecución real.



Imagen#7: Combobox plan de ejecución. Fuente: Elaboración propia.

### Análisis de resultados del sexto requerimiento:

Para mostrar un resumen de uso de índices en la consulta se dispuso un botón el cuál, al presionarlo despliega ese resumen en el “text field” previamente creado para facilitar el proceso de impresión.



Imagen#8: Botón índices con el resumen. Fuente: Elaboración propia.

## Conclusiones

* Con base en los resultados obtenidos por medio de la solución del problema, se concluye que la creación de una aplicación de escritorio facilita que el usuario final del sistema pueda ver los privilegios de las tablas y de los atributos de una base de datos, así como los planes de ejecución.
* Con base en las investigaciones realizadas para el desarrollo de este proyecto, se concluye que Tkinter es una herramienta óptima de cara a la realización de GUI en Python, ya que, cuenta con mucho soporte, y también, la información que existe en la red es muy basta para solucionar cualquier inconveniente que se presentó, y de la propia facilidad para utilizarla.
* Con base en las investigaciones realizadas para el desarrollo de este proyecto, se concluye que Python es una herramienta óptima de cara a las conexiones con las bases de datos, ya que, cuenta con mucho soporte, y también, una gran variedad de librerías para hacer estas conexiones. Cabe resaltar que, la información acerca de Python y sus librerías existentes en la red es muy extensa y útil para solucionar cualquier inconveniente que se presentó.

## Recomendaciones

* Dividir el trabajo en un proyecto informático es una buena práctica para su ejecución e implementación.
* Coordinar adecuadamente con los integrantes del equipo de trabajo para la adecuada distribución de tareas, estableciendo medios de retroalimentación por parte de los integrantes, para la corrección inmediata de posibles anomalías en el desarrollo del proyecto.
* Utilizar librerías estables de Python para el desarrollo de cualquier aplicación que las requiera, ya que, puede darse el caso que la librería que se esté utilizando ya no reciba soporte, y esto es un problema de cara a futuro.
* Gestione su proyecto con un controlador de versiones (GitKraken[[4]](#footnote-4)), ya que compartir el código es más fácil, mejora la corrección de errores y maneja las etapas de este.

## Referencias Bibliográficas

[1]"PyCharm: uno de los mejores IDE para Python — Escuela de Python", Escuela de Python, 2020. [Online]. Available: https://www.escuelapython.com/pycharm-uno-de-los-mejores-ide-para-python/. [Accesado: 15- Nov- 2020].

[2]"Tkinter - Introducción - ▷ Cursos de Programación de 0 a Experto © Garantizados", ▷ Cursos de Programación de 0 a Experto © Garantizados, 2020. [Online]. Available: https://unipython.com/tkinter-introduccion/. [Accesado: 15- Nov- 2020].

[3]"Acceso a Bases de Datos desde Python con pyodbc - ▷ Cursos de Programación de 0 a Experto © Garantizados", ▷ Cursos de Programación de 0 a Experto © Garantizados, 2020. [Online]. Available: https://unipython.com/acceso-a-bases-de-datos-con-pyodbc/. [Accesado: 15- Nov- 2020].

[4]"Free Git GUI Client - Windows, Mac, Linux | GitKraken", *GitKraken.com*, 2018. [Online]. Disponible en: https://www.gitkraken.com/. [Accesado: 15- Nov- 2020].

**Enlace del video en inglés:**

<https://drive.google.com/file/d/1CS2LTWQ5IzZnoKQJXUwwQh2Lo28Hz1Xz/view?usp=sharing>

1. Es una librería que proporciona a las aplicaciones de Python una interfaz de usuario fácil de programar. Además, es un conjunto de herramientas GUI de Tcl/Tk (Tcl: Tool Command Language), proporcionando una amplia gama de usos, incluyendo aplicaciones web, de escritorio, redes, administración, pruebas y muchos más.[2] [↑](#footnote-ref-1)
2. PyCharm es uno de los entornos de desarrollo más completos para Python. Es parte de la suite de herramientas de programación ofrecidas por JetBrains, que cuenta con entornos para construir código en distintos idiomas como PHP y Ruby. [1] [↑](#footnote-ref-2)
3. es un módulo de Python de código abierto que simplifica el acceso a las bases de datos ODBC desde Python, implementando el uso de la DB API 2.0 de una forma conveniente para Python. pyodbc también es considerado como un controlador SQL para Python. [3] [↑](#footnote-ref-3)
4. Permite llevar el completo seguimiento de nuestros repositorios, ver ramas, tags, crear nuevos, todo el historial de nuestro trabajo, commits. [4] [↑](#footnote-ref-4)