



INFORME DE PRÁCTICA O TALLER N° 1 y N° 2

DATOS DE LA ASIGNATURA

Nombre del estudiante / Integrantes:

Michael Paucar

Jefferson Álvarez

José Luis Frías

Apauki Herrería

Carrera: Desarrollo de Software

Docente: Luis Yulan

Asignatura: Integración de Sistemas Informáticos

Nivel: Quinto

Fecha: 11/07/2023

Calificación:

Periodo Académico:

Abril – Septiembre 2023

1. TÍTULO:

Taller 1: Creación de base de datos y consulta.

Taller 2: Generación de informe en PDF y envío por correo, creación de repositorio colaborativo.

2. OBJETIVO ALCANZADO:

Taller 1: Se ha solicitado crear un proyecto con Python que permita realizar consultas en la base de datos creada previamente (Blog), al realizar esta consulta se debe generar un reporte de la información que se haya obtenido, esto en formato PDF y que este reporte se envíe por correo electrónico a las direcciones especificadas.

Taller 2: Para completar esta actividad se solicita crear un ejecutable del programa y automatizar las acciones correspondientes, es decir, que al realizar la consulta se genere automáticamente el reporte y, así mismo, se envíe de forma automática este reporte por correo electrónico en un tiempo determinado.

Y por último crear un repositorio en github para trabajar de forma colaborativa en el presente proyecto.



3. DESCRIPCIÓN DE LAS ACTIVIDADES DESARROLLADAS DURANTE LA PRÁCTICA

ACTIVIDAD	ENTREGABLES DEL ESTUDIANTE
1. Se ha creado una base de datos llamada “Blog” en el administrador de MYSQL, phpMyAdmin de la cual se realizará posteriormente las consultas solicitadas.	Base de datos
2. Creación de proyecto Python para realizar la consulta de la base de datos	Programa Python para consulta en BD
3. Creación de función para crear reporte de la consulta previa en formato PDF	Archivo .py para creación de reporte PDF
4. Creación de función para enviar el reporte PDF resultante de la consulta	Archivo .py para envío de reporte por email
5. Automatización de envío de reporte por email.	Tarea programada para envío de email

4. MARCO TEÓRICO:

Taller 1:

Django:

Django es un framework de desarrollo para Python que se emplea para la creación de páginas web. Se trata de una herramienta de código abierto y gratuita que cuenta con una comunidad amplia y que comparte recursos constantemente. Además, Django también cuenta con funciones de pago que pueden facilitar más el trabajo de los desarrolladores.

Django es una herramienta que se puede usar para el desarrollo *full-stack* de aplicaciones y páginas web, así como para el desarrollo de servidores. Está considerado como el mejor framework para el desarrollo de aplicaciones web con Python y es uno de los marcos de desarrollo más demandados por los programadores que trabajan con este lenguaje en el desarrollo web.

Ngrok:

Ngrok es un servicio que nos permite crear nuestro servidor local en un subdominio para poder visualizarlo fuera de la LAN, a través de internet; por ejemplo, para realizar pruebas de intrusión necesitamos de un túnel donde recibiremos las conexiones, un método que puede sustituir el uso de Ngrok es el “PortForwarding”, sin embargo, este método es más laborioso, a comparación de Ngrok. Esto nos ayudará mucho principalmente si nuestro objetivo se encuentra fuera de la red



local, también cabe recalcar que no solo sirve para realizar pruebas de intrusión, ustedes podrán subir sus aplicaciones o plataformas de forma online.

API:

Una API o interfaz de programación de aplicaciones es un conjunto de definiciones y protocolos que se usan para diseñar e integrar el software de las aplicaciones. Las API permiten que sus productos y servicios se comuniquen con otros, sin necesidad de saber cómo están implementados. Esto simplifica el desarrollo de las aplicaciones y permite ahorrar tiempo y dinero. Las API le otorgan flexibilidad; simplifican el diseño, la administración y el uso de las aplicaciones; y ofrecen oportunidades de innovación, lo cual es ideal al momento de diseñar herramientas y productos nuevos (o de gestionar los actuales).

Taller 2:

Entorno Programable:

El entorno programable a través de Python sirve como estructura para programar código base del proyecto al igual del uso de herramientas para su funcionamiento a través de librerías, plugins, funciones, clases, Apis y otros.

Para el presente proyecto la decisión de utilizar Python es su flexibilidad que provee y el uso acelerado de herramientas externas como librerías que permiten su integración de código.

Python:

Según Python (2018) menciona que “es un lenguaje de programación poderoso y fácil de aprender. Tiene estructuras de datos de alto nivel eficientes y un sistema orientado a objetos simple pero eficiente. La sintaxis elegante y el tipo dinámico de Python, junto con su expresividad, lo convierten en el lenguaje ideal para escribir y desarrollar aplicaciones rápidamente en muchos entornos en muchas plataformas”. De cierta manera el uso de este lenguaje permite integrar una codificación de scripts y adicional a este permite generar archivos ejecutables con los cuales se generarán los documentos.

ReportLab:

Es una librería que permite la construcción de documentos PDF a través de la codificación de scripts con Python, esta librería según su plataforma permite recopilar el código escrito por el usuario y compilarlo de manera funcional. (ReportLab, 2014)

PyInstaller

Esta es una librería que permite convertir el script de un código de Python en un instalador de tipo exe, esto es gracias a la librería que permite interpretar el código y compilarlo como un ejecutable,



adicional la librería permite instalar herramientas que gestionen la creación del archivo ejecutable.
(PyInstaller, 2015)

5. MATERIALES / EQUIPOS

NRO	Nombre de Equipo / Material	Uso que se le dio en la práctica
1	Python	Lenguaje de Programación
2	ReportLab	Librería para generar reportes PDF
3	Computador	Proporcionó los programas anteriormente mencionados para realizar cada actividad.
4	Visual Studio Code	Entorno de desarrollo, creación del código fuente y ejecutable.
5	PyInstaller	Librería para creación del ejecutable (.exe)
6	mysql_connector	Librería para conexión con base de datos
7	Xampp	Ayuda a levantar los servicios y los puertos que necesitamos para conectar el proyecto con la base de datos.
8	Git/Github	Trabajo colaborativo y creación de versiones

6. CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES	RECOMENDACIONES
Al completar los talleres 1 y 2, se logra desarrollar un proyecto completo que permite consultar una base de datos en Python, generar un informe en formato PDF basado en los resultados de la consulta y enviar automáticamente este informe por correo electrónico en un horario específico. Estas tareas combinadas ofrecen una solución integral y automatizada para el procesamiento de datos y la entrega de informes.	Es importante generar un entorno de desarrollo con las herramientas correctas, sin embargo se debe poner mayor énfasis en las versiones de cada herramienta para que su integración sea funcional, puesto que en versiones anteriores suele provocar errores a nivel técnico del proyecto.
El uso de Python y las bibliotecas correspondientes permite automatizar tareas que de otro modo requerirían una intervención manual. En este caso, la consulta a la base de datos, la generación del informe y el envío por correo electrónico se realizan automáticamente.	Se recomienda probar librerías que se acomoden al proyecto con un breve análisis de por medio, ya que algunas librerías no están funcionales o por su lado están deprecadas, por el momento en el proyecto las librerías que mayor aceptación tuvieron fueron las de PyInstaller y ReportLab que se encuentran como unas de las principales



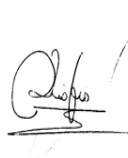



sin la necesidad de intervención humana.	actualmente.
Los talleres permiten integrar diferentes tecnologías para lograr una solución completa. Se utiliza una base de datos para almacenar la información, Python para la manipulación y extracción de datos, la generación de informes en formato PDF y la capacidad de enviar correos electrónicos. Esta integración demuestra la flexibilidad y potencia de las herramientas disponibles.	La integración de los módulos generados por python permiten funcionar como un solo sistema gracias al código de tipo script que se codificó, sin embargo se puede equilibrar su funcionamiento mediante el uso de una base de datos, para esto se requiere un énfasis en el enlace del proyecto desarrollado en Python con la integración de SQL a través del mysql.conector

8. EVALUACIÓN REFLEXIVA DE LA PRÁCTICA (Estudiante)

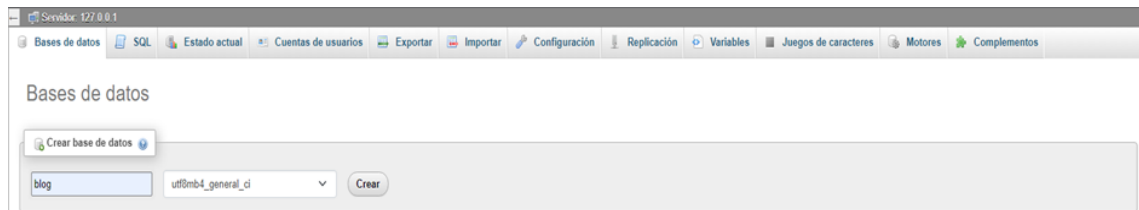
ESCALA DE EVALUACIÓN							
1	2	3	4	5			
Nunca	Pocas veces	Algunas veces	Casi siempre	Siempre			
DESCRIPCIÓN DE CRITERIOS A EVALUAR			Escala de evaluación				
			1	2	3	4	5
¿Considera que la práctica le ayudó en su formación integral (actitudes, conocimientos y habilidades)?							X
¿Lo aprendido en su proceso de formación profesional le sirvió para el desempeño de su Práctica?						X	
¿Considera que las actividades que realizó fueron las pertinentes para el cumplimiento del objetivo de su práctica?							X
¿La realización de la práctica cumplió con sus expectativas personales y profesionales?						X	
DE MANERA BREVE, CLARA Y CONCISA, PRESENTA SUS CONSIDERACIONES SOBRE ⁹ :							
Dificultades observadas en el desarrollo de su práctica		El uso de Python permite integrar varios recursos en el desarrollo de módulos. Las descargas de los reportes generados pueden referirse con rutas específicas en el proyecto.					
Logros obtenidos gracias a su práctica, aprovechables para su futura vida profesional		Generar reportes en PDF para consultas de bases de datos y automatización de envío de los mismos por email..					
Recomendaciones para el desarrollo de módulos		Énfasis en el direccionamiento de rutas del proyecto. El uso variado de herramientas en el diseño de un PDF con librerías.					
OBSERVACIONES FINALES: Podemos concluir que se han aplicado los conocimientos y técnicas impartidas por el profesor durante el presente módulo, pero igualmente se ha podido observar que se necesitaría más tiempo e información para una aplicación completa de los temas que se han tratado, ya que, en el ámbito laboral, al aplicar ciertas técnicas y procedimientos es algo mucho más extenso y complejo que los ejercicios prácticos que se realizaron aquí.							



 Michael Paucar	 Jefferson Álvarez	 José Luis Frías	 Apauki Herrería
Firma de los estudiantes			

9. EVIDENCIAS FOTOGRÁFICAS / ANEXOS:

1. Se ha creado una base de datos llamada “Blog” en el administrador de MYSQL, phpMyAdmin de la cual se realizará posteriormente las consultas solicitadas.



2. Aquí se crea la tabla correspondiente de nombre “blog” con los campos especificados: id (por default, autoincremental), BlogId, Title, Author_Name y las fechas Start_Date, End_Date.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	Acción
<input type="checkbox"/>	1 id	bigint(20)			No	Ninguna		AUTO_INCREMENT	Cambiar Eliminar Más
<input type="checkbox"/>	2 BlogId	varchar(30)	utf8mb4_general_ci		No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/>	3 Title	varchar(200)	utf8mb4_general_ci		No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/>	4 Author_Name	varchar(300)	utf8mb4_general_ci		No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/>	5 Start_Date	date			No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/>	6 End_Date	date			No	Ninguna			Cambiar Eliminar Más

3. Esta base de datos fue creada previamente en una app web con Django, por lo que lanzamos dicha aplicación para agregar registros a la base de datos “Blog”.





A continuación, se muestra de forma gráfica en una tabla que igualmente forma parte del proyecto anterior con Django:

Blog Information

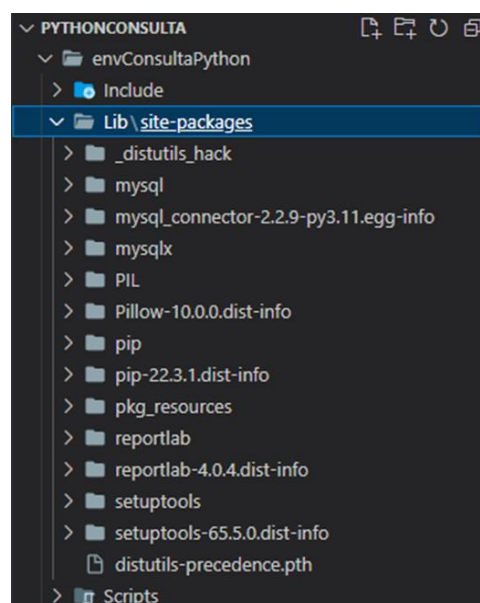
Blog ID	Title	Author	Start Date	End Date	Edit Blog	Remove Blog
100	Introduction to Django	Mary Smith	March 4, 2022	March 6, 2022	Update	Delete
2	Prueba Consulta Python	Apauki	July 5, 2023	July 6, 2023	Update	Delete

-Ahora se muestra la base de datos con el registro creado en este paso:

<div>← T →</div>						id	BlogId	Title	Author_Name	Start_Date	End_Date
<input type="checkbox"/>		Editar		Copiar		Borrar	1 100	Introduction to Django	Mary Smith	2022-03-04	2022-03-06
<input type="checkbox"/>		Editar		Copiar		Borrar	2 2	Prueba Consulta Python	Apauki	2023-07-05	2023-07-06

Nota: el registro que se observa primero fue uno de prueba que se creó al definir el proyecto Django.

4. Ahora, con este registro agregado a la base de datos con la que se trabajará esta actividad, se crea una carpeta para el proyecto y dentro de la misma se inicia un ambiente virtual, para posteriormente realizar la instalación de las bibliotecas y funciones necesarias. En este caso se necesita instalar mysql-connector, que como se puede deducir por su nombre, sirve para realizar la conexión con la base de datos de MYSQL y permitir realizar las consultas correspondientes:



5. Para realizar la consulta, se crea un archivo con la extensión de Python (.py) llamado “consulta”, en la que se importa el controlador mysql-connector previamente instalado, se especifica la conexión a la base de datos (host, usuario, contraseña y el nombre de la base de datos). Después se define un “cursor” que es el que realizará la consulta y dentro se especifican las filas (campos de la tabla) que se van a extraer, después se utiliza un bucle for para imprimir lo recuperado, y por último se cierra la conexión con la base de datos para evitar errores.

```
consulta.py X
consulta.py > ...
1  import mysql.connector
2
3  conexion1=mysql.connector.connect(host="localhost",
4                                   user="root",
5                                   passwd="",
6                                   database="blog")
7  cursor1=conexion1.cursor()
8  cursor1.execute("select * from blog")
9  for fila in cursor1:
10     print(fila)
11  conexion1.close()
```

6. Entonces, para crear un reporte en formato PDF como se ha solicitado, se ha instalado la librería de herramientas reportlab que sirve justamente para el propósito. Se crea un nuevo archivo .py llamado “reporte” en el que se importan las herramientas necesarias que vienen incluidas en reportlab: letter para el tamaño del documento, SimpleDocTemplate, Table que creará la tabla con los mismos campos de la tabla en la que se realiza la consulta, TableStyle, colors; de igual forma se importa el controlador mysql-connector por obvias razones.

```
reporte.py > ...
1  from reportlab.lib.pagesizes import letter
2  from reportlab.platypus import SimpleDocTemplate, Table, TableStyle
3  from reportlab.lib import colors
4  import mysql.connector
```

7. Se procede con la creación de la conexión a la base de datos.

```
# Conecta a la base de datos
conn = mysql.connector.connect(
    host="localhost",
    user="root",
    passwd="",
    database="blog"
)
cursor = conn.cursor()
```

8. Se realiza la consulta, se almacena el resultado y se cierra la conexión.



```
# Ejecuta la consulta SQL
cursor.execute('SELECT * FROM blog')

# Obtén los resultados de la consulta
resultados = cursor.fetchall()

# Cierra la conexión a la base de datos
conn.close()
```

9. Se crea el reporte en una tabla de la que se especifican los campos, se agrega estilos y agrega el contenido resultante de la consulta realizada previamente.

```
reporte.py > ...
26 # Crea el documento PDF
27 pdf_filename = 'informe.pdf'
28 document = SimpleDocTemplate(pdf_filename, pagesize=letter)
29
30 # Crea una lista para almacenar los datos de la tabla
31 data = []
32
33 # Agrega los encabezados de las columnas
34 column_names = cursor.column_names
35 data.append(column_names)
36
37 # Agrega los datos de las filas
38 for fila in resultados:
39     data.append(fila)
40
41 # Crea la tabla con los datos
42 table = Table(data)
43
44 # Estilo de la tabla
45 style = TableStyle([
46     # Encabezados de columna con fondo gris
47     ('BACKGROUND', (0, 0), (-1, 0), colors.grey),
48     # Texto blanco en los encabezados de columna
49     ('TEXTCOLOR', (0, 0), (-1, 0), colors.whitesmoke),
50     # Alineación centrada en todas las celdas
51     ('ALIGN', (0, 0), (-1, -1), 'CENTER'),
52     # Fuente en negrita para los encabezados de columna
53     ('FONTNAME', (0, 0), (-1, 0), 'Helvetica-Bold'),
54     # Tamaño de fuente 12 para los encabezados de columna
55     ('FONTSIZE', (0, 0), (-1, 0), 12),
56     # Espacio inferior de 12 puntos para los encabezados de columna
57     ('BOTTOMPADDING', (0, 0), (-1, 0), 12),
58     # Fondo beige para las filas de datos
59     ('BACKGROUND', (0, 1), (-1, -1), colors.beige),
60     ('GRID', (0, 0), (-1, -1), 1, colors.black), # Líneas de cuadrícula negras
61 ])
62
63 # Aplica el estilo a la tabla
64 table.setStyle(style)
65
66 # Crea el contenido y lo agrega al documento
67 content = [table]
68 document.build(content)
69 print(f'Informe generado: {pdf_filename}')
```

10. Ahora se crea un nuevo archivo que tendrá como propósito enviar este reporte generado a través de correo electrónico a las direcciones que se especifiquen dentro del mismo.



Primero se importan las herramientas necesarias de creación de email y después se especifican el remitente, los destinatarios, un asunto, el cuerpo del mensaje, se ejecuta este archivo para que se proceda con el envío del correo con un tiempo determinado en la misma función. También se especifica la ruta en la que se encuentra el archivo de reporte que se adjuntará a este correo y el nombre del mismo.

```
correoReporte.py > ...
1 import smtplib
2 import os
3 import time
4
5 from email.mime.multipart import MIMEMultipart
6 from email.mime.text import MIMEText
7 from email.mime.base import MIMEBase
8 from email import encoders
9
10
11 # Iniciamos los parámetros del script
12 remitente = 'hapauki@gmail.com'
13 destinatarios = ['aherrera@itsqmet.edu.ec', 'jeffalvarez1996@gmail.com', 'jjossue_lolkilso@hotmail.com', 'lyulan@itsqmet.edu.ec']
14 asunto = '[RPI] Correo de envío de reporte PDF (consulta en BD con Python) Grupo 2'
15 cuerpo = 'Se ha adjuntado a este correo el archivo reporte en PDF generado como resultado de la consulta en la base de datos "Blog" con Python'
16
17 def func1():
18     os.system('python correoReporte.py')
19     print("Se ejecuta")
20     time.sleep(10)
21     func1()
22
23 ruta_adjunto = "F:\Proyectos Integracion\PythonConsulta\reporte.pdf"
24 nombre_adjunto = 'PruebaAdjunto.pdf'
25
```

11. El correo se envía gracias a la librería importada previamente llamada smtplib utilizando la función .sendmail y especificando remitente, destinatarios y el cuerpo del mensaje.

12. Para generar el ejecutable de este programa que se ha creado se instala la librería pyinstaller, y se utilizan los comandos de la misma.

-Primero se crea el ejecutable de consulta.py

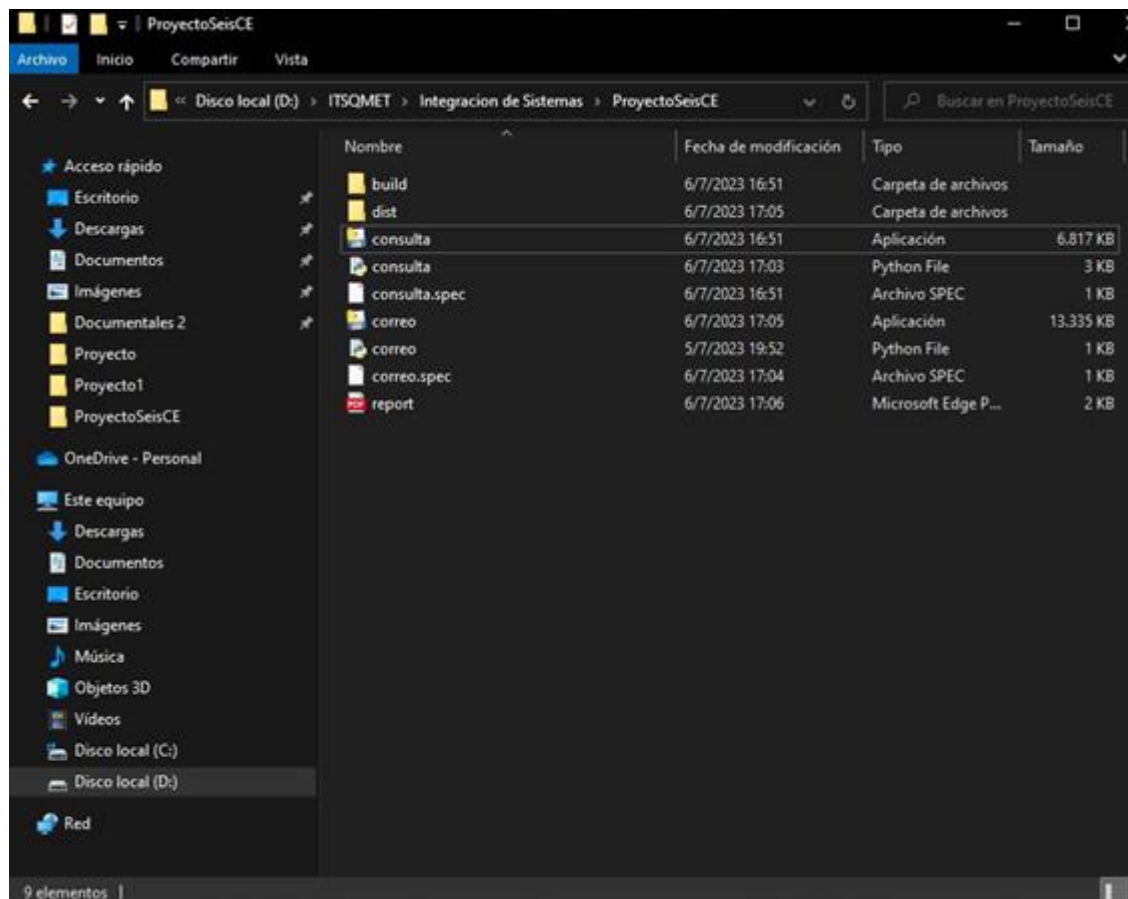
```
Seleccionar Símbolo del sistema
D:\ITSQMET\Integracion de Sistemas\ProyectoSeisCE\pyinstaller --clean --onefile --windowed consulta.py
979 INFO: PyInstaller: 5.13.0
980 INFO: Python: 3.11.2
981 INFO: Platform: Windows-10-10.0.19045-SP0
982 INFO: wrote D:\ITSQMET\Integracion de Sistemas\ProyectoSeisCE\consulta.spec
983 INFO: Removing temporary files and cleaning cache in C:\Users\jeffa\AppData\Local\pyinstaller
984 INFO: Extending PYTHONPATH with paths
985 INFO: ['D:\ITSQMET\Integracion de Sistemas\ProyectoSeisCE']
986 INFO: checking Analysis
987 INFO: Building Analysis because Analysis-00.toc is non-existent
988 INFO: Initializing module dependency graph...
989 INFO: Caching module graph hooks...
990 INFO: Analyzing base library.zip...
991 INFO: Loading module hook 'hook-heapq.py' from 'C:\Users\jeffa\AppData\Local\Programs\Python\Python311\Lib\site-packages\PyInstaller\hooks'...
992 INFO: Loading module hook 'hook-encodings.py' from 'C:\Users\jeffa\AppData\Local\Programs\Python\Python311\Lib\site-packages\PyInstaller\hooks'...
993 INFO: Loading module hook 'hook-pickle.py' from 'C:\Users\jeffa\AppData\Local\Programs\Python\Python311\Lib\site-packages\PyInstaller\hooks'...
994 INFO: Caching module dependency graph...
995 INFO: running Analysis Analysis-00.toc
996 INFO: Adding Microsoft.Windows.Common-Controls to dependent assemblies of final executable
997 INFO: required by C:\Users\jeffa\AppData\Local\Programs\Python\Python311\python.exe
998 INFO: Analyzing D:\ITSQMET\Integracion de Sistemas\ProyectoSeisCE\consulta.py
999 INFO: Processing module hooks...
1000 INFO: Looking for ctypes DLLs
1001 INFO: Analyzing run-time hooks...
1002 INFO: Including run-time hook 'C:\Users\jeffa\AppData\Local\Programs\Python\Python311\Lib\site-packages\PyInstaller\hooks\rthooks\pyi_rth_inspect.py'
1003 INFO: Looking for dynamic libraries
1004 INFO: Extra DDL search directories (AddDllDirectory): []
1005 INFO: Extra DDL search directories (PATH): ['C:\Program Files (x86)\VMware\VMware Workstation\bin', 'C:\Windows\system32', 'C:\Windows', 'C:\Windows\System32\Wbem', 'C:\Windows\System32\WindowsPowerShell\v1.0\bin', 'C:\Windows\System32\OpenSSH\bin', 'C:\Program Files\Git\cmd', 'C:\Program Files (x86)\AQMEI\AQMEI Backupper\7.2.0', 'C:\Program Files\nodejs\bin', 'C:\Users\jeffa\AppData\Local\Programs\Python\Python311\Scripts', 'C:\Users\jeffa\AppData\Local\Programs\Python\Python311\bin', 'C:\Users\jeffa\AppData\Local\Microsoft\WindowsApps', 'C:\Users\jeffa\AppData\Local\Programs\Microsoft VS Code\bin', 'C:\Users\jeffa\AppData\Local\GitHubDesktop\bin', 'C:\Users\jeffa\AppData\Roaming\npm']
1006 INFO: Looking for eggs
1007 INFO: Using Python library C:\Users\jeffa\AppData\Local\Programs\Python\Python311\python311.dll
1008 INFO: Found binding redirects:
1009 []
1010 INFO: Warnings written to D:\ITSQMET\Integracion de Sistemas\ProyectoSeisCE\build\consulta\warn-consulta.txt
1011 INFO: Graph cross-reference written to D:\ITSQMET\Integracion de Sistemas\ProyectoSeisCE\build\consulta\xref-consulta.html
1012 INFO: checking PVZ
1013 INFO: Building PVZ because PVZ-00.toc is non-existent
1014 INFO: Building PVZ (2libArchive) D:\ITSQMET\Integracion de Sistemas\ProyectoSeisCE\build\consulta\PVZ-00.pyz
1015 INFO: Building PVZ (2libArchive) D:\ITSQMET\Integracion de Sistemas\ProyectoSeisCE\build\consulta\PVZ-00.pyz completed successfully.
```

-Después el ejecutable correo.py (enviar el correo con el reporte creado en consulta.py)



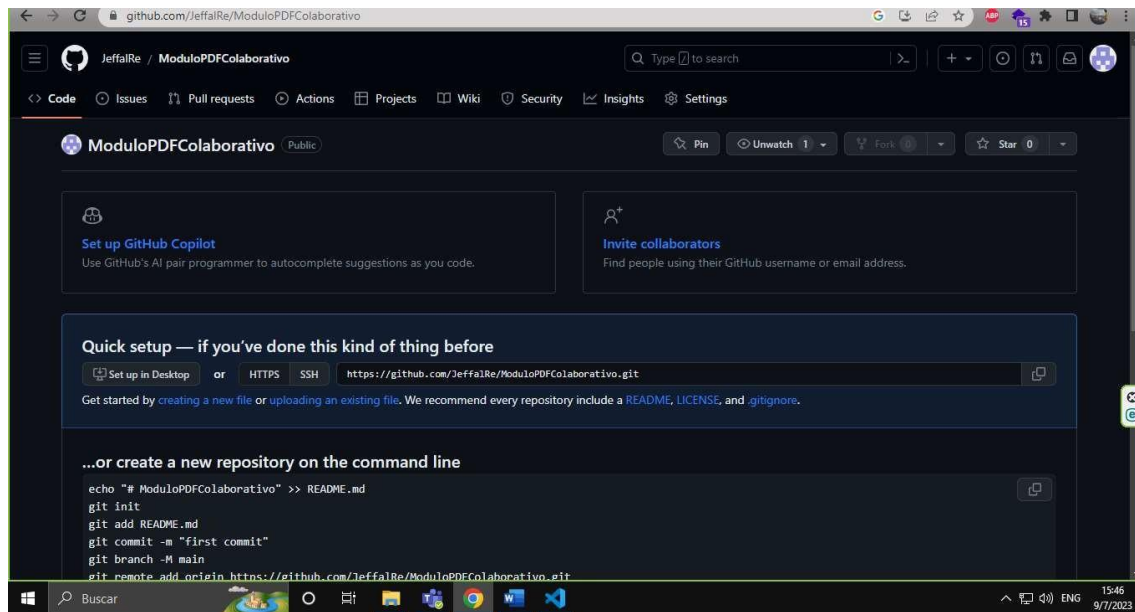
```
Seleccionar Símbolo del sistema
D:\ITSQMET\Integracion de Sistemas\ProyectoSeisCE>pyinstaller --clean --onefile --windowed correo.py
2464 INFO: PyInstaller: 5.13.0
2464 INFO: Python: 3.11.2
2495 INFO: Platform: Windows-10-10.0.19045-SP0
2495 INFO: wrote D:\ITSQMET\Integracion de Sistemas\ProyectoSeisCE\correo.spec
2511 INFO: Removing temporary files and cleaning cache in C:\Users\jeffa\AppData\Local\pyinstaller
2527 INFO: Extending PYTHONPATH with paths
['D:\ITSQMET\Integracion de Sistemas\ProyectoSeisCE']
3242 INFO: checking Analysis
3242 INFO: Building Analysis because Analysis-00.toc is non existent
3242 INFO: Initializing module dependency graph...
3242 INFO: Caching module graph hooks...
3273 INFO: Analyzing base_library.zip ...
7777 INFO: Loading module hook 'hook-heapq.py' from 'C:\Users\jeffa\AppData\Local\Programs\Python\Python311\Lib\site-packages\PyInstaller\hooks'...
7964 INFO: Loading module hook 'hook-encodings.py' from 'C:\Users\jeffa\AppData\Local\Programs\Python\Python311\Lib\site-packages\PyInstaller\hooks'...
11442 INFO: Loading module hook 'hook-pickle.py' from 'C:\Users\jeffa\AppData\Local\Programs\Python\Python311\Lib\site-packages\PyInstaller\hooks'...
15426 INFO: Caching module dependency graph...
15598 INFO: running Analysis Analysis-00.toc
15645 INFO: Adding Microsoft.Windows.Common-Controls to dependent assemblies of final executable
required by C:\Users\jeffa\AppData\Local\Programs\Python\Python311\python.exe
15801 INFO: Analyzing D:\ITSQMET\Integracion de Sistemas\ProyectoSeisCE\correo.py
C:\Users\jeffa\AppData\Local\Programs\Python\Python311\Lib\site-packages\mysql\connector\abstracts.py:130: SyntaxWarning: "is" with a literal. Did you mean "--"?
if group is 'connector_python':
C:\Users\jeffa\AppData\Local\Programs\Python\Python311\Lib\site-packages\mysql\connector\optionfiles.py:98: SyntaxWarning: "is" with a literal. Did you mean "--"?
if group is 'connector_python':
16796 INFO: Loading module hook 'hook-platform.py' from 'C:\Users\jeffa\AppData\Local\Programs\Python\Python311\Lib\site-packages\PyInstaller\hooks'...
17213 INFO: Loading module hook 'hook-xml.py' from 'C:\Users\jeffa\AppData\Local\Programs\Python\Python311\Lib\site-packages\PyInstaller\hooks'...
17521 INFO: Loading module hook 'hook-reportlab.lib.utils.py' from 'C:\Users\jeffa\AppData\Local\Programs\Python\Python311\Lib\site-packages\PyInstaller\hooks'...
17724 INFO: Loading module hook 'hook-PIL.py' from 'C:\Users\jeffa\AppData\Local\Programs\Python\Python311\Lib\site-packages\PyInstaller\hooks'...
17943 INFO: Loading module hook 'hook-PIL.Image.py' from 'C:\Users\jeffa\AppData\Local\Programs\Python\Python311\Lib\site-packages\PyInstaller\hooks'...
19458 INFO: Loading module hook 'hook-pycparser.py' from 'C:\Users\jeffa\AppData\Local\Programs\Python\Python311\Lib\site-packages\PyInstaller\hooks'...
20349 INFO: Processing pre-safe import module hook distutils from 'C:\Users\jeffa\AppData\Local\Programs\Python\Python311\Lib\site-packages\PyInstaller\hooks'...
20927 INFO: Processing pre-find module path hook distutils from 'C:\Users\jeffa\AppData\Local\Programs\Python\Python311\Lib\site-packages\PyInstaller\hooks'...
21145 INFO: Loading module hook 'hook-distutils.py' from 'C:\Users\jeffa\AppData\Local\Programs\Python\Python311\Lib\site-packages\PyInstaller\hooks'...
21145 INFO: Loading module hook 'hook-distutils.util.py' from 'C:\Users\jeffa\AppData\Local\Programs\Python\Python311\Lib\site-packages\PyInstaller\hooks'...
21317 INFO: Loading module hook 'hook-sysconfig.py' from 'C:\Users\jeffa\AppData\Local\Programs\Python\Python311\Lib\site-packages\PyInstaller\hooks'...
22099 INFO: Loading module hook 'hook-setuptools.py' from 'C:\Users\jeffa\AppData\Local\Programs\Python\Python311\Lib\site-packages\PyInstaller\hooks'...
28887 INFO: Loading module hook 'hook-pkg_resources.py' from 'C:\Users\jeffa\AppData\Local\Programs\Python\Python311\Lib\site-packages\PyInstaller\hooks'...
```

-Ahora se agregan estos ejecutables a la carpeta del proyecto y ya se pueden ejecutar.

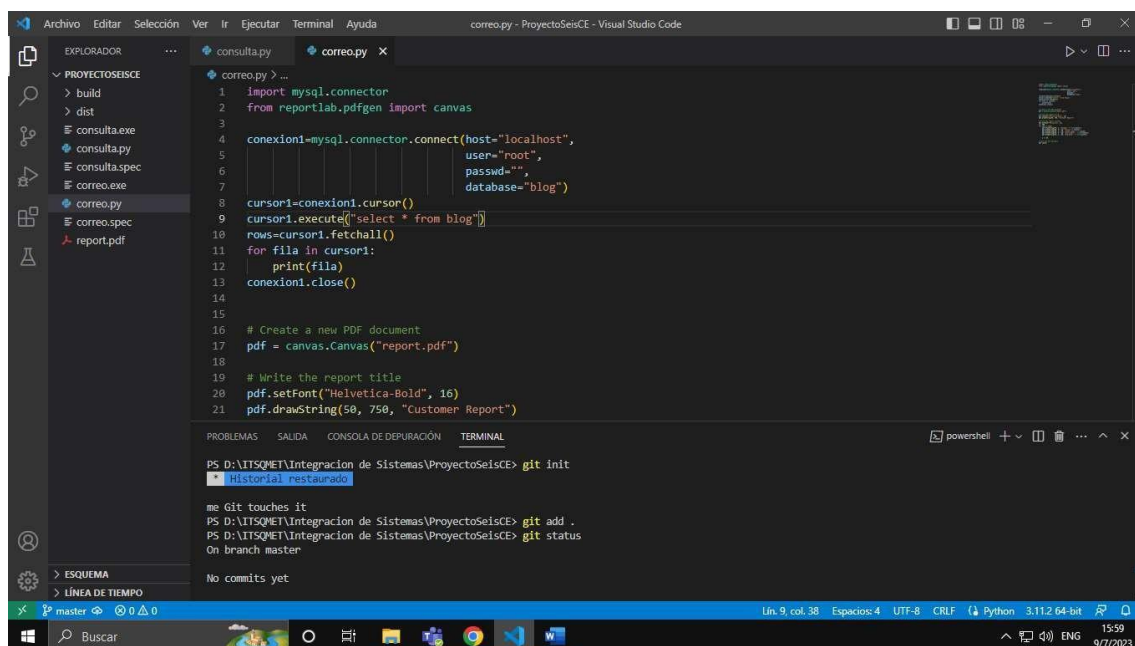




Generación de un repositorio colaborativo en GitHub.



13. Ejecución de GitHub, añadido de archivos desarrollados y validación de la rama principal de repositorio.



Conexión y subida de archivos a GitHub.

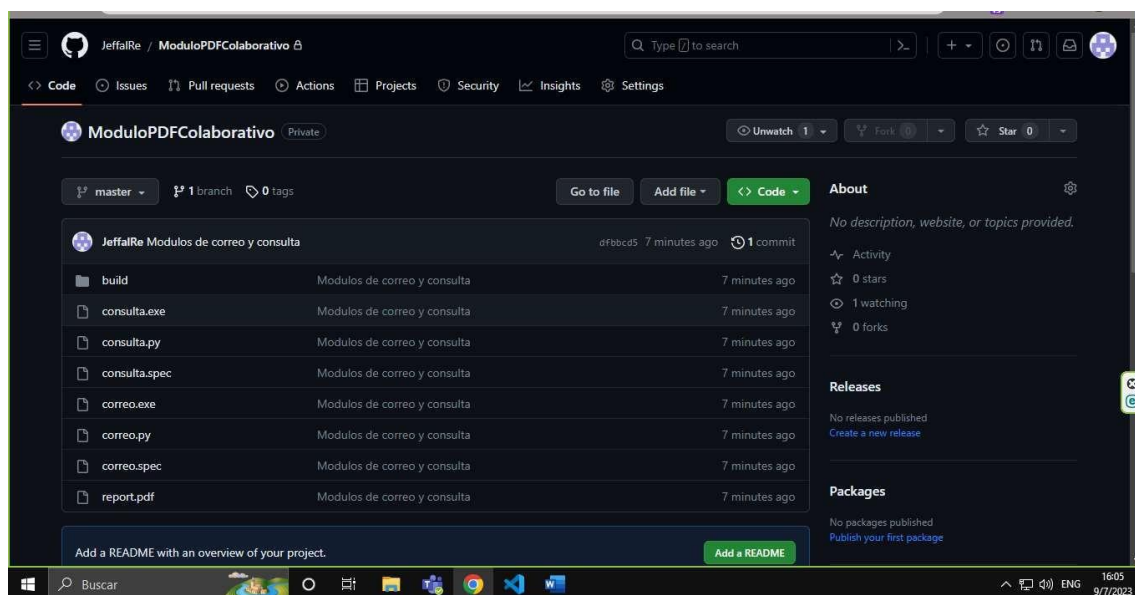


```
1 import mysql.connector
2 from reportlab.pdfgen import canvas
3
4 conexion=mysql.connector.connect(host="localhost",
5                                 user="root",
6                                 passwd="",
7                                 database="blog")
8 cursor=conexion.cursor()
9 cursor.execute("select * from blog")
10 rows=cursor.fetchall()
11 for fila in cursor:
12     print(fila)
13 conexion.close()
14
15 # Create a new PDF document
16 pdf = canvas.Canvas("report.pdf")
17
18 # Write the report title
19 pdf.setFont("Helvetica-Bold", 16)
20 pdf.drawString(50, 750, "Customer Report")
```

```
create mode 100644 correo.spec
create mode 100644 report.pdf
PS D:\ITSQMET\Integracion de Sistemas\ProyectoSeisCE> git remote add origin https://github.com/JeffalRe/ModuloPDFColaborativo.git
PS D:\ITSQMET\Integracion de Sistemas\ProyectoSeisCE> git push -u origin master
Enumerating objects: 37, done.
Counting objects: 100% (37/37), done.
Delta compression using up to 4 threads
Compressing objects: 100% (37/37), done.
Writing objects: 45% (17/37), 9.02 MiB | 2.91 MiB/s
```

14. Repositorio creado en GitHub completamente funcional.

Repositorio de GitHub: <https://github.com/JeffalRe/ModuloPDFColaborativo.git>



Adición de colaboradores a través de GitHub con el fin de desarrollar el proyecto de manera grupal.



github.com/jeffalRe/ModuloPDFColaborativo/settings/access

Code Issues Pull requests Actions Projects Security Insights Settings

General

Access

Collaborators

Code and automation

Branches

Tags

Rules Beta

Actions

Webhooks

Codespaces

Pages

Security

Code security and analysis

Deploy keys

Secrets and variables

Integrations

GitHub Apps

Who has access

PRIVATE REPOSITORY

Only those with access to this repository can view it. [Manage](#)

DIRECT ACCESS

0 collaborators have access to this repository. Only you can contribute to this repository.

Manage access

6 6 6

You haven't invited any collaborators yet

[Add people](#)

16:11 9/7/2023

Code Issues Pull requests Actions Projects Security Insights Settings

General

Collaborators

Code and automation

Branches

Tags

Rules Beta

Actions

Webhooks

Codespaces

Pages

Security

Code security and analysis

Deploy keys

Secrets and variables

Integrations

GitHub Apps

Add a collaborator to ModuloPDFColaborativo

[Select a collaborator above](#)

6 6 6

You haven't invited any collaborators yet

[Add people](#)

16:12 9/7/2023

Curso: INTEGRACION DE SISTEMAS x Sound of Freedom: cuándo se en... Manage access x WhatsApp x

github.com/jeffalRe/ModuloPDFColaborativo/settings/access?guidance_task=

Code and automation

Branches

Tags

Rules Beta

Actions

Webhooks

Codespaces

Pages

Security

Code security and analysis

Deploy keys

Secrets and variables

Integrations

GitHub Apps

Email notifications

Manage access

[Add people](#)

☐ Select all Type

<input type="checkbox"/>	Apauki Awaiting Apauki's response	Pending Invite	Remove
<input type="checkbox"/>	LuiggiMH Awaiting LuiggiMH's response	Pending Invite	Remove
<input type="checkbox"/>	MichaelPauca-m Awaiting MichaelPauca-m's response	Pending Invite	Remove

Get team access controls and discussions for your contributors in an organization. [Create an organization](#)

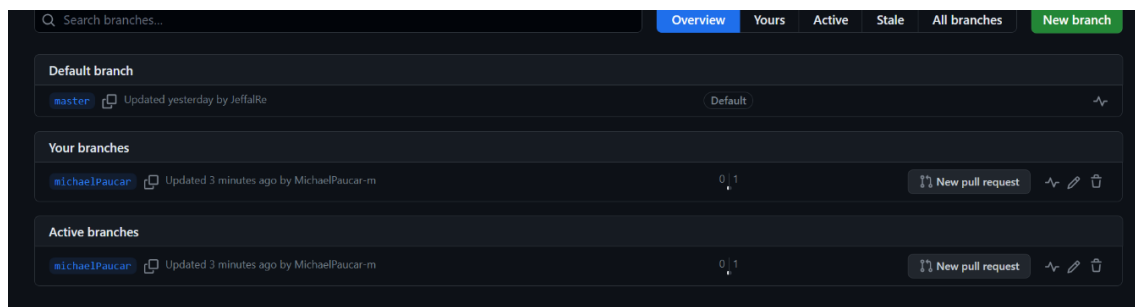
NEW Private repos and unlimited members are free.

[Previous](#) [Next](#)

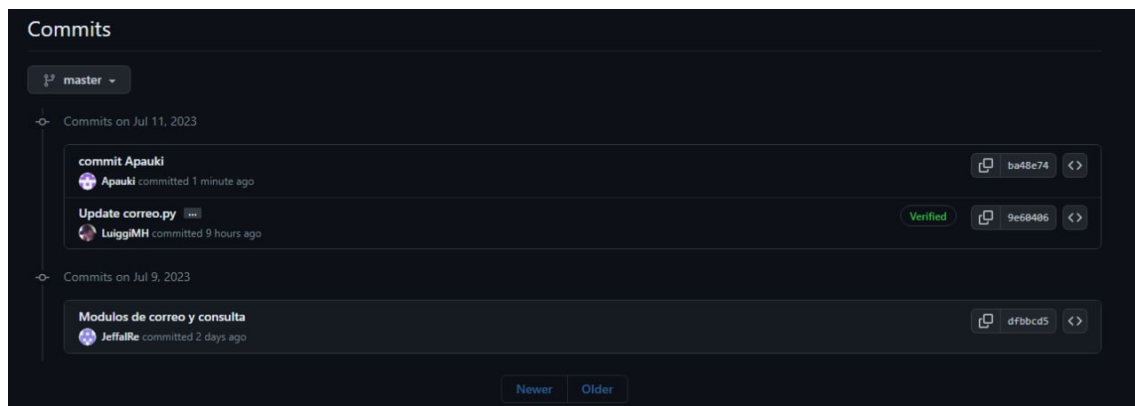
19:59 10/7/2023



15. Creamos las diferentes ramas o branches al repositorio



16. Cada colaborador ya puede realizar cambios en el proyecto



Bibliografía

- Flores, F. (03 de Julio de 2010). *Visual Studio Code*. Obtenido de ¿Que es VSC?: <https://openwebinars.net/blog/que-es-visual-studio-code-y-que-ventajas-ofrece/>
- PyInstaller. (10 de Septiembre de 2015). *Librería PyInstaller*. Obtenido de Como funciona PyInstaller: <https://pyinstaller.org/en/stable/operating-mode.html>
- Python. (15 de Enero de 2018). *¿Qué es Python?* Obtenido de Funcionamiento de Python: <https://docs.python.org/es/3/tutorial/>
- ReportLab. (15 de Julio de 2014). *ReportLab*. Obtenido de Como instalar reportLab?: <https://www.reportlab.com/>
- A.C.V., A. (s.f.). *El Curso del Hacker*. Obtenido de <https://www.elcursodelhacker.com/ngrok/>
- Red Hat. (20 de Enero de 2023). *Red Hat*. Obtenido de <https://www.redhat.com/es/topics/api/what-are-application-programming-interfaces>
- Tokio School. (06 de Junio de 2022). *Tokio New Technology School*. Obtenido de <https://www.tokioschool.com/noticias/que-es-django/>



REVISIÓN	APROBACIÓN
Firma:	Firma:
Fecha: 11/07/2023	Fecha: 12/07/2023
Docente:	Coordinadora:
MSc. Luis Yulan	Ing. Marcela Saavedra
Carrera:	
Desarrollo de Software	