

TECNOLÓGICO NACIONAL DE MÉXICO

INSTITUTO TECNOLÓGICO DE CULIACÁN



Fecha:

29 noviembre de 2021

PROYECTO:

convertidor de CSV a KML

INTEGRANTES:

NOMBRES DE LOS INTEGRANTES DEL EQUIPO (iniciando con los apellidos en orden alfabético y en mayúsculas)	PROGRAMACION AVANZADA Lun-Jue de 10:00 a 11:00 hrs / Vie 09:00 a 11:00
De La Cruz Nuñez Brandon Martin 17171089	ING. MECATRONICA
Medina Burgos julio Eduardo 17171167	ING. MECATRONICA
Sánchez Reyes José de Jesús 17171251	ING. MECATRONICA
Chahin Alapizco Jesus Haydar 16171093	ING. MECATRONICA
	ING. MECATRONICA
DOCENTE: Carlos Alberto Martínez Félix	

ÍNDICE

1	INTRODUCCIÓN.....	1
1.1	Objetivo General.....	2
1.2	Objetivo Específico CSV al formato KML.....	2
1.3	Objetivo específico Interfaz grafica	2
2	Materiales Y Métodos	3
2.1	Características técnicas del equipo	3
2.1.1	Información del sistema.....	3
2.1.2	Dispositivos de visualización	4
2.2	Metodología (Diagrama de Flujo).....	6
3	RESULTADOS	7
3.1	Código.....	7
4	CONCLUSIONES PERSONALES	17
4.1	Conclusión De La Cruz Núñez Brandon Martin.....	17
4.2	Conclusión de Medina Burgos julio Eduardo	17
4.3	Conclusión de Sánchez Reyes José de Jesús.....	17
4.4	Conclusión de Chahin Alapizco Jesus Haydar.....	17
4.5	Conclusión de <aquí se pone el nombre completo del alumno iniciando con los apellidos>	18
5	CITAS BIBLIOGRÁFICAS Y REFERENCIAS	19
6	Anexos	21

ÍNDICE DE GRÁFICAS, FIGURAS Y TABLAS

<i>Imagen 3—1: Código de librerías utilizadas.....</i>	<i>7</i>
<i>Imagen 3—2: Código de la interfaz grafica.....</i>	<i>8</i>
<i>Imagen 3—3: Código de abrir archivo.....</i>	<i>9</i>
<i>Imagen 3—4: Código de convertir archivo.....</i>	<i>9</i>
<i>Imagen 3—5: Código de Guardar archivo.....</i>	<i>10</i>
<i>imagen 3—6: Código de botones e ingreso de nombre del archivo.....</i>	<i>10</i>
<i>Imagen 3—7: Imagen de la interfaz grafica.....</i>	<i>11</i>
<i>Imagen 3—8: Imagen de la interfaz con archivo elegido.....</i>	<i>12</i>
<i>Imagen 3—9: Imagen de conversión exitosa.....</i>	<i>13</i>
<i>Imagen 3—10: Eleccion de nombre.....</i>	<i>14</i>
<i>Imagen 3—11: Guardado de archivo convertido.....</i>	<i>15</i>
<i>Imagen 3—12: Errores que pueden generarse por el operador.....</i>	<i>16</i>

1 INTRODUCCIÓN

El programa en la que se desarrolla este proyecto, documentado sobre sus funciones, que es la conversión de los archivos en su formato base CSV (sistema de control de versiones) es un sistema para gestionar código fuente. Normalmente se refiere al uso de software para manejar versiones de ficheros fuente.

Y transformarlos a archivos del formato KML (conocido antiguamente como Keyhole Markup Language) es un formato de archivo basado en XML que se utiliza para visualizar información en un contexto geográfico.

La información KML se puede dibujar en muchos navegadores basados en la tierra, como ArcGIS Earth y ArcGIS Pro. Esta clase de conversiones son importantes para tomar códigos fuentes de CSV que suelen ser modificados por diferentes programadores para sus propios propósitos, y mostrarlo con el formato KML para su demostración grafica en una gran variedad de programas capaces de leer el formato como ArcGIS Earth y ArcGIS Pro Y a su vez entender y comprender el proceso de transformación de un formato a través de una interfaz visual.

Diseñada y elaborada con los diferentes tipos de componentes de interfaz aprendidos en clase tomando en consideración los que consideramos los mas necesarios y fundamentales para el proyecto.

1.1 Objetivo General

El Programa fue desarrollado con el propósito de la conversión de Archivos base de CSV a archivos base KML con el propósito del desarrollo del programa para ver y saber los comandos necesarios para la conversión de uno a otro a través de la gran variedad de librerías que posee Python y visualizar y entender si es necesario agregar más de manera externa a las librerías bases que ya posee el programa de programación para el funcionamiento del mismo. Y a su vez aplicarlos de una interfaz gráfica desarrollada por nosotros que cumpla con lo establecido.

1.2 Objetivo Específico CSV al formato KML

El de entender y comprende el desarrollo del programa en la cual esta enfocado en la conversión de los archivos del formato CSV al formato KML, asegurándose que la transformación de un formato a otro sea la correcta observando las propiedades del archivo desde las propiedades del mismo con clic derecho en el archivo.

1.3 Objetivo específico Interfaz grafica

- Al desarrollar el programa de la conversión de archivos CSV al Formato KML, estos se tienen que visualizar, seleccionar, convertir y guardar a través de una interfaz gráfica que cumpla con lo esencial del programa mientras conserva un apartado técnico específico y funcional, cumpliendo las características fundamentales del programa.

2 Materiales Y Métodos

El lenguaje de programación que se efectuó para la elaboración de este proyecto de conversión de CSV a KML es Python a través del programa Spyder Ejecutado a través de Anaconda que es una distribución de los lenguajes de programación Python y R para computación científica, que tiene como objetivo simplificar la administración y la implementación de paquetes. La distribución incluye paquetes de ciencia de datos adecuados para Windows, Linux y macOS.

2.1 Características técnicas del equipo

La PC utilizada para la realización de este programa de convención CSV a KML a través del programa Spyder de Python, posee las siguientes características:

2.1.1 Información del sistema

- Machine name: LAPTOP-5STT8IQO
- Machine Id: {95380AE6-228E-43EE-B729-E81880FA4C6C}
- Operating System: Windows 10 Home Single Language 64-bit (10.0, Build 19042) (19041.vb_release.191206-1406)
- Language: Spanish (Regional Setting: Spanish)
- System Manufacturer: HP
- System Model: OMEN Laptop 15-en0xxx
- BIOS: F.13 (type: UEFI)
- Processor: AMD Ryzen 7 4800H with Radeon Graphics (16 CPUs), ~2.9GHz
- Memory: 16384MB RAM
- Available OS Memory: 15732MB RAM
- Page File: 21626MB used, 5880MB available

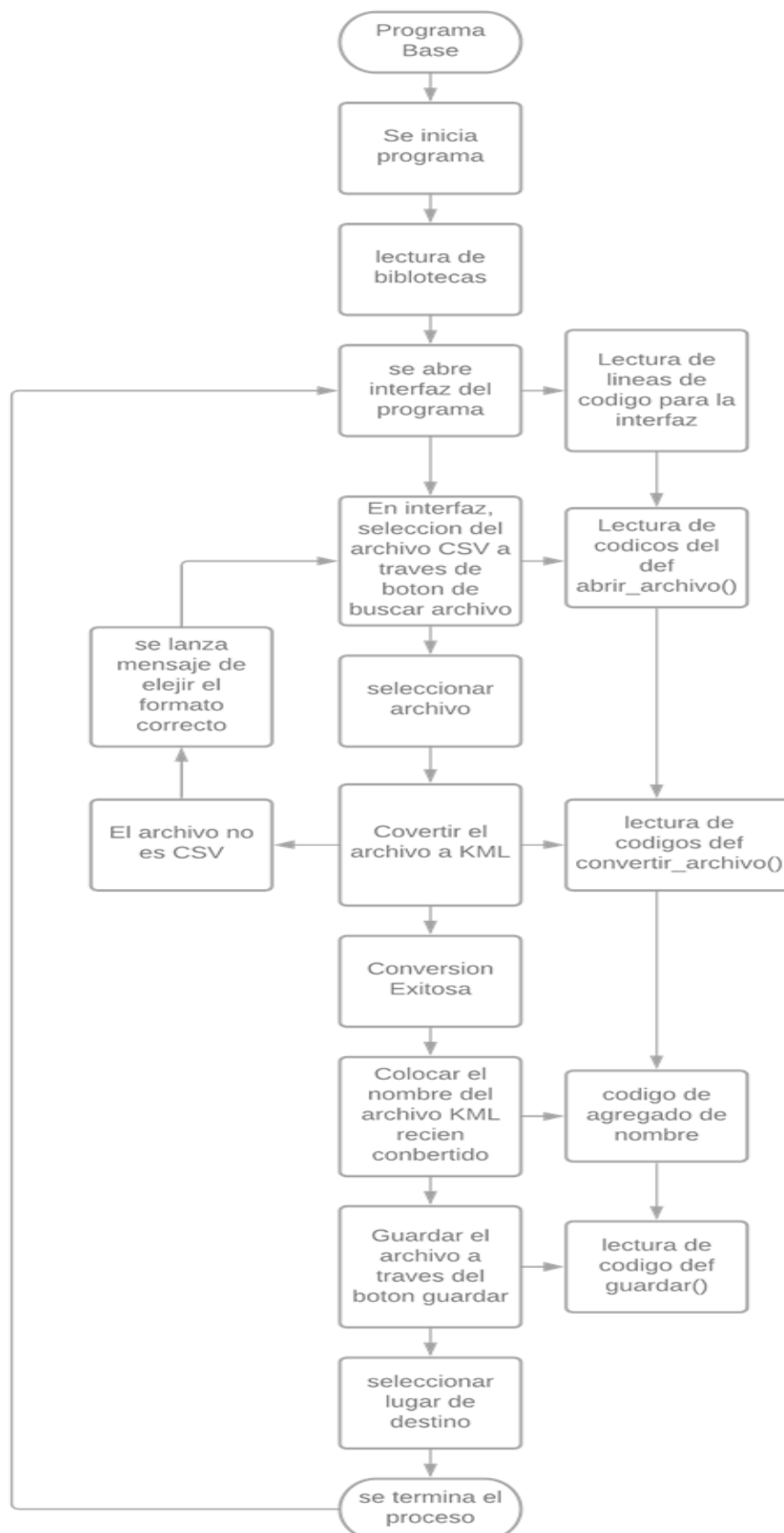
- Windows Dir: C:\windows
- DirectX Version: DirectX 12
- DX Setup Parameters: Not found
- User DPI Setting: 120 DPI (125 percent)
- System DPI Setting: 120 DPI (125 percent)
- DWM DPI Scaling: UnKnown
- Miracast: Available, with HDCP
- Microsoft Graphics Hybrid: Supported
- DirectX Database Version: 1.3.1
- DxDiag Version: 10.00.19041.0928 64bit Unicode

2.1.2 Dispositivos de visualización

- Card name: AMD Radeon (TM) Graphics
- Manufacturer: Advanced Micro Devices, Inc.
- Chip type: AMD Radeon Graphics Processor (0x1636)
- DAC type: Internal DAC (400MHz)
- Device Type: Full Device (POST)
- Device Key:
Enum\PCI\VEN_1002&DEV_1636&SUBSYS_8787103C&REV_C6
- Device Status: 0180200A
[DN_DRIVER_LOADED|DN_STARTED|DN_DISABLEABLE|DN_NT_ENUMERATOR|DN_NT_DRIVER]
- Device Problem Code: No Problem
- Driver Problem Code: Unknown
- Display Memory: 8361 MB
- Dedicated Memory: 496 MB

- Shared Memory: 7865 MB
- Current Mode: 1920 x 1080 (32 bit) (60Hz)
- Card name: NVIDIA GeForce RTX 2060
- Manufacturer: NVIDIA
- Chip type: GeForce RTX 2060
- DAC type: Integrated RAMDAC
- Device Type: Full Device
- Device Key:
Enum\PCI\VEN_10DE&DEV_1F15&SUBSYS_8787103C&REV_A1
- Device Status: 0180200A
[DN_DRIVER_LOADED|DN_STARTED|DN_DISABLEABLE|DN_NT_ENUMERATOR|DN_NT_DRIVER]
- Device Problem Code: No Problem
- Driver Problem Code: Unknown
- Display Memory: 13845 MB
- Dedicated Memory: 5980 MB
- Shared Memory: 7865 MB

2.2 Metodología (Diagrama de Flujo)



3 RESULTADOS

Toda la línea de código de programa del convertidor de formatos CSV a formato KML realizado en spyder de Python. Con evidencia del programa funcionando a la perfección cumpliendo el propósito con el cual de desarrollado.

3.1 Código

Códigos de librerías necesarias de Python para la interfaz, la convención del archivo CSV y el guardado del archivo KML. Con una librería agregada de manera externa simplekml para cumplir el propósito del programa. Se importa también ImageTk,Image, filedialog, pandas, os y messagebox.

```
7
8 import sys
9
10 from tkinter import *
11 from PIL import ImageTk,Image
12 from tkinter import filedialog
13 import simplekml
14 import pandas
15 from tkinter import messagebox
16 import os
17
```

Figura 3—1:Código de Librerías utilizadas.

Líneas de código para la gestión y creación de la interfaz grafica del programa de conversión de CSV a KML. Tiene incluida una imagen como logotipo característico del programa. Se usa tittle, geometry, resizable, Label, place, image open y resize para el aspecto general de la interfaz gráfica.

```

18 #creacion de la interfaz
19
20 ventana_principal = Tk()
21 ventana_principal.title("convertidor de CSV a KML")
22 ventana_principal.geometry("500x500")
23 ventana_principal.resizable(width=False, height=False)
24 ventana_principal.config(bg="grey")
25 inicio=Label(ventana_principal, text="convertidor de Archivo CSV a Archivo KML",font=("Arial",13))
26 nombre=Label(ventana_principal,text="Earthcode")
27 nombre.place(x=0,y=50)
28 inicio.place(x=100,y=10)
29
30 image=Image.open("C:\\Users\\hayda\\Dropbox\\Mi PC (LAPTOP-5STT8IQ0)\\Downloads\\tierra1.jpeg")
31 image=image.resize((50,50),Image.ANTIALIAS)
32
33 img=ImageTk.PhotoImage(image)
34 lbl_img=Label(ventana_principal,image=img)
35 lbl_img.place(x=0,y=0)
36

```

Figura 3—2: Código de la interfaz gráfica.

Líneas de código con el nombre def abrir_archivo(): para la carga del archivo CSV al programa para realizar la conversión a KML. Se usa la keyword global de archivo y inicio_1 para el abrir archivos que se necesitan para la transformación.

```

37 def abrir_archivo():
38     global archivo
39     archivo = filedialog.askopenfilename(title="Documentos", initialdir="C:/", filetypes=[("Archivos .csv", "*.csv")])
40     print(archivo)
41     global inicio_1
42     inicio_1=Label(ventana_principal, text=archivo)
43     inicio_1.place(x=50,y=120)
44

```

Figura 3—3: Código de abrir archivo.

Línea de código con el nombre `def convertir_archivo()`, estas líneas son las que se encargaran de convertir el archivo con formato CSV al formato KML, en la cual también se encarga de avisar si el archivo subido al programa es CSV o de lo contrario lanza un mensaje indicando que no es formato correcto, así como también si en el programa aún no se han subido archivos para la conversión. Se utilizan la keyword `kml` e `inicio_2` para la transformación del archivo y se tiene en consideración los errores de `UnicodeDecodeError` y `NameError` por si no se carga un archivo o un archivo incorrecto.

```
44
45 def convertir_archivo():
46     try:
47         global kml
48         kml = simplekml.Kml()
49         df = pandas.read_csv(archivo)
50         for lon, lat in zip(df["Longitude"], df["Latitude"]):
51             kml.newpoint(coords = [(lon,lat)])
52         global inicio_2
53         inicio_2=Label(ventana_principal,text="Conversión Lista")
54         inicio_2.place(x=50,y=220)
55
56     except UnicodeDecodeError:
57         messagebox.showwarning(message='Debes cargar un archivo .csv', title="Error")
58     except NameError:
59         messagebox.showwarning(message='Primero se debe cargar un archivo', title="Error")
60     del kml
61
```

Imagen 3—4: Código de convertir archivo

Líneas de códigos para el guardado del archivo KML ya convertido, en la cual lanza una ventana en el lugar en donde quiera guardar el documento KML ya otorgándole el nombre establecido antes de guardarlo. En la cual lanza 2 errores indicando de volver agregar nombre si colocas caracteres especiales o si no se a cargado ni convertido ningún archivo. Se usan las keyword `archivo_guardar` y `nn` para el guardado de archivos ya transformados y se tienen

en consideración posibles errores con OSError y NameError por si poseen caracteres especiales o no se ha cargado ningún archivo.

```
62 def guardar():
63     try:
64         global archivo_guardar
65         archivo_guardar = filedialog.askdirectory()
66         global nn
67         nn = text.get()
68         print(nn)
69         nuevo_nombre = archivo_guardar + "/" + nn + ".kml"
70         kml.save(nuevo_nombre)
71         messagebox.showinfo(message='Archivo Guardado', title="Listo")
72         inicio_1.destroy()
73         inicio_2.destroy()
74         text.delete(0, END)
75     except OSError:
76         messagebox.showwarning(message='Los nombres del archivo no pueden tener ninguno de los siguientes caracteres: /:*"?\\<>|', title="Error")
77     except NameError:
78         messagebox.showwarning(message='Primero se debe cargar un archivo', title="Error")
79
```

Imagen 3—5: Código de Guardar archivo

Líneas de código en la cual se agregan los botones que ejecutan las funciones de buscar archivo, convertir archivo, el agregado de nombre para el archivo y el guardar el archivo. Se usan las características especiales Button, place y Label para la colocación de botones y el agregado de nombre necesario para realizar las operaciones de transformación del formato CSV al KML

```
79
80 #Lo que hay en la ventana Principal
81 boton_archivo = Button(ventana_principal, text="buscar archivo", command=abrir_archivo, width=11, height=3)
82 boton_archivo.place(x=300, y=100)
83
84 boton_convertir = Button(ventana_principal, text="convertir archivo", command=convertir_archivo, width=12, height=3)
85 boton_convertir.place(x=300, y=200)
86
87 boton_guardar = Button(ventana_principal, height=1, width=10, text="Guardar", command=guardar)
88 boton_guardar.place(x=200, y=350)
89
90 nombre = Label(ventana_principal, text="Ingresa nombre del archivo a crear").place(x=30, y=300)
91
92 text = Entry(ventana_principal)
93 text.place(x=250, y=300)
94
95 ventana_principal.mainloop()
96
```

imagen 3—6: Código de botones e ingreso de nombre del archivo

Diseño de interfaz graficas sencilla y fácil de entender para cualquiera que quiera convertir un archivo CSV a KML en la cual cumple su función a la perfección. Sin la necesidad de un diseño sobresaliente y con características innecesarias.

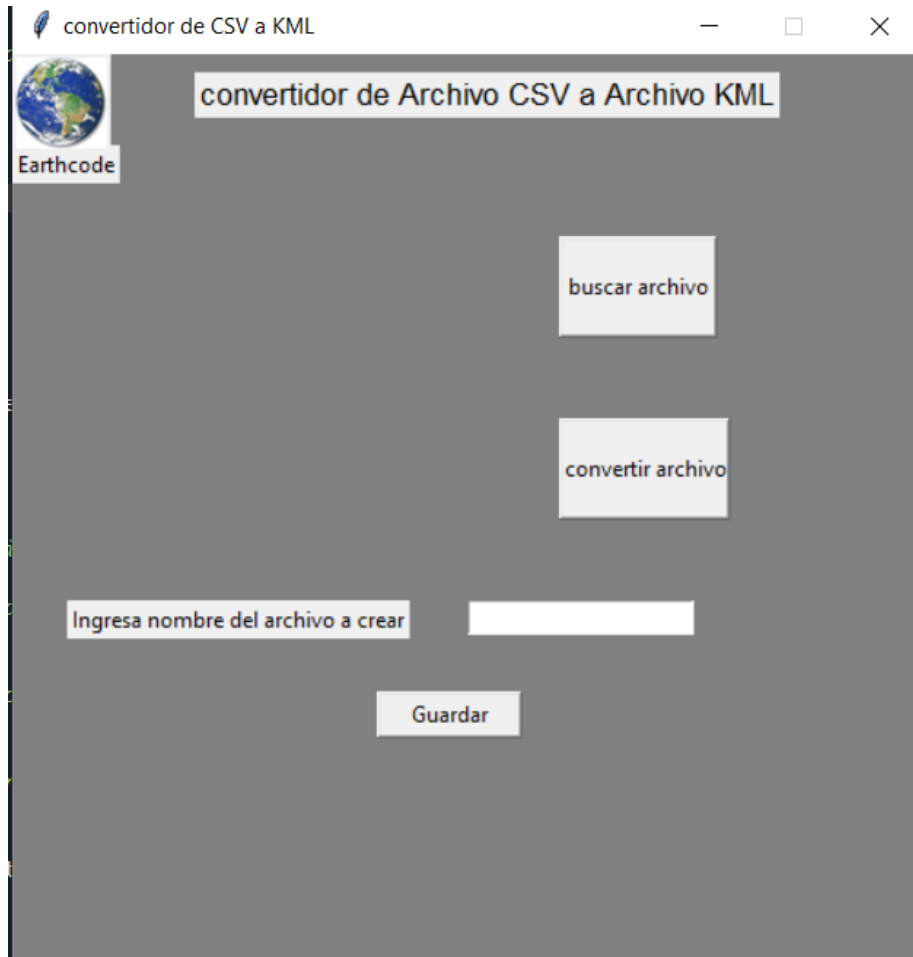


Imagen 3—7: Imagen de la interfaz grafica

Cuando se selecciona un archivo CSV para realizar la conversión al lado izquierdo de la interfaz grafica aparece la dirección de donde fue tomado el archivo.



Imagen 3—8: Imagen de la interfaz con archivo elegido

Al seleccionar el archivo, se da clic en el botón de convertir archivo y realizara la conversión de manera automática y lanzara un mensaje en la cual indicara que el formato CSV a formato KML.

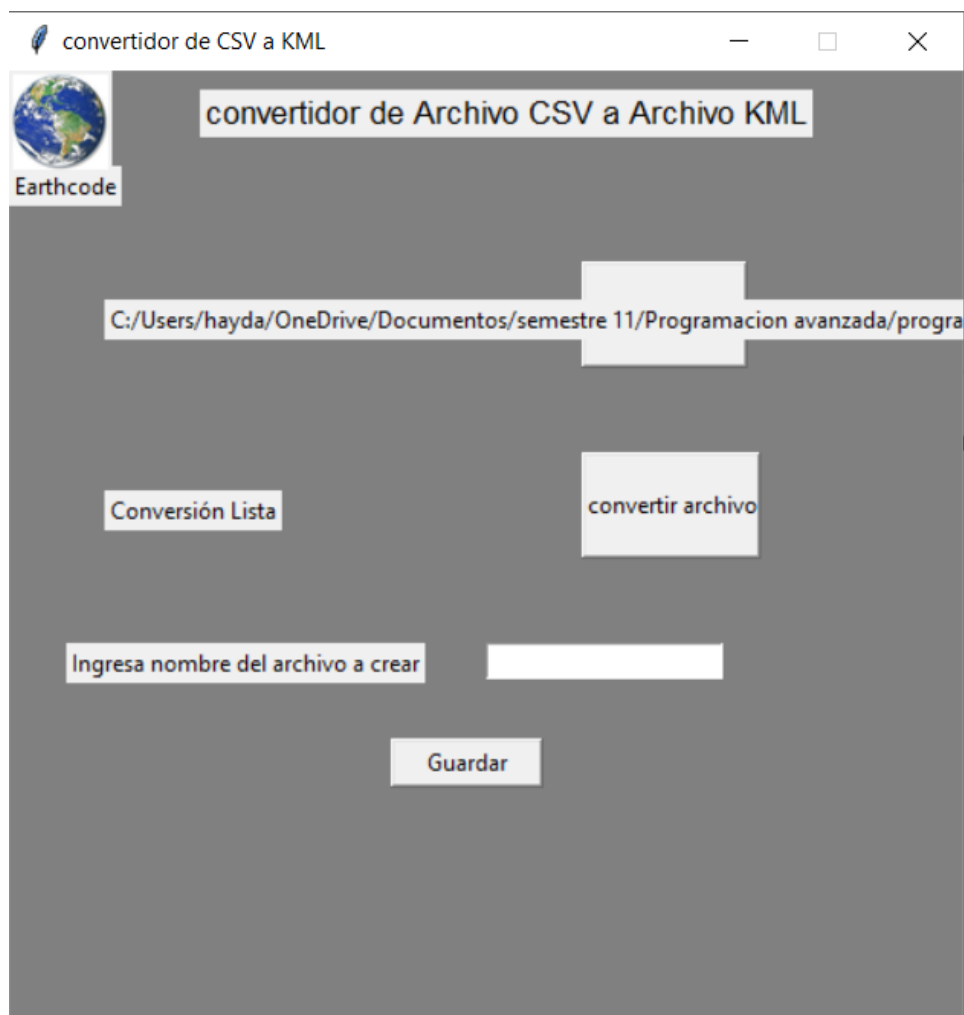


Imagen 3—9: Imagen de conversión exitosa

El archivo ya transformado, se le tiene que agregar un nombre en la cual la barra de escribir en la parte inferior de la interfaz se le puede añadir el nombre que se quiera tener el documento.

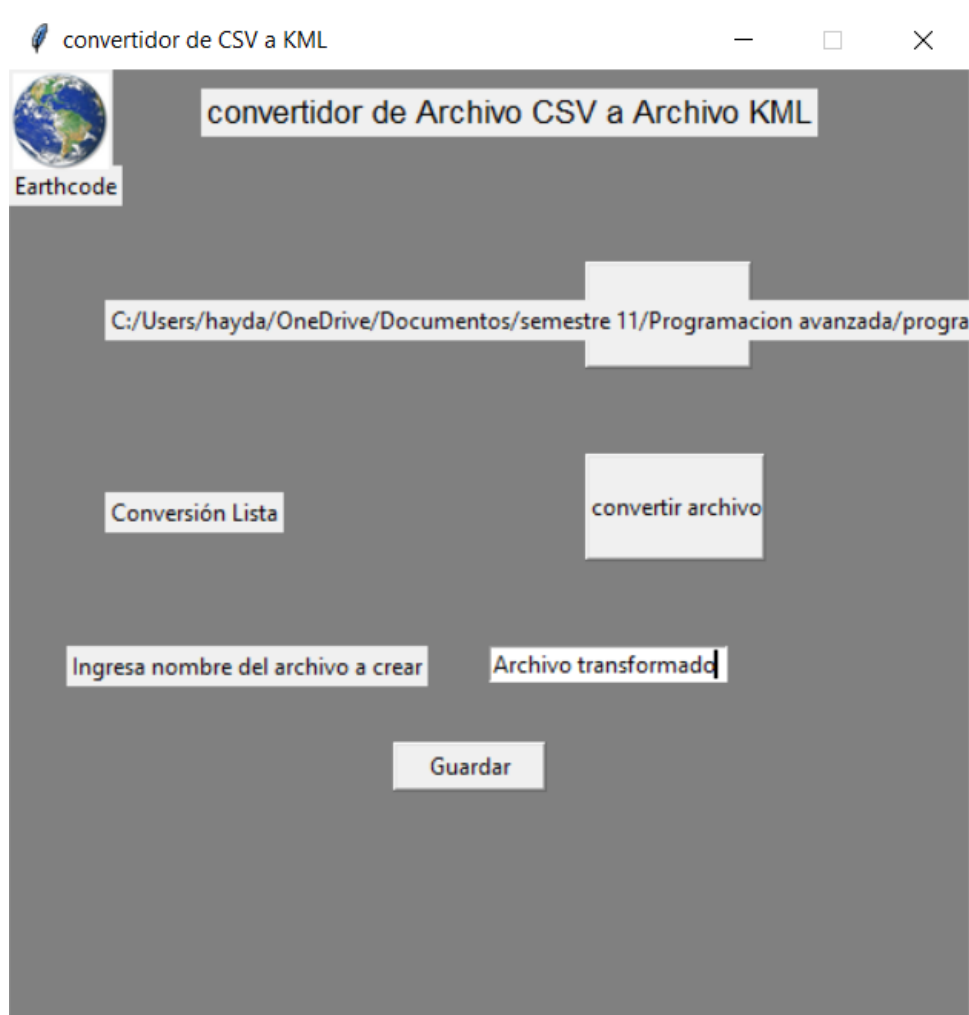


Imagen 3—10: Elección de nombre

Ahora se procede a guardar el archivo KML en un lugar de preferencia por el Usuario, dando clic en el botón guardar que se encuentra en la parte inferior de la interfaz gráfica.

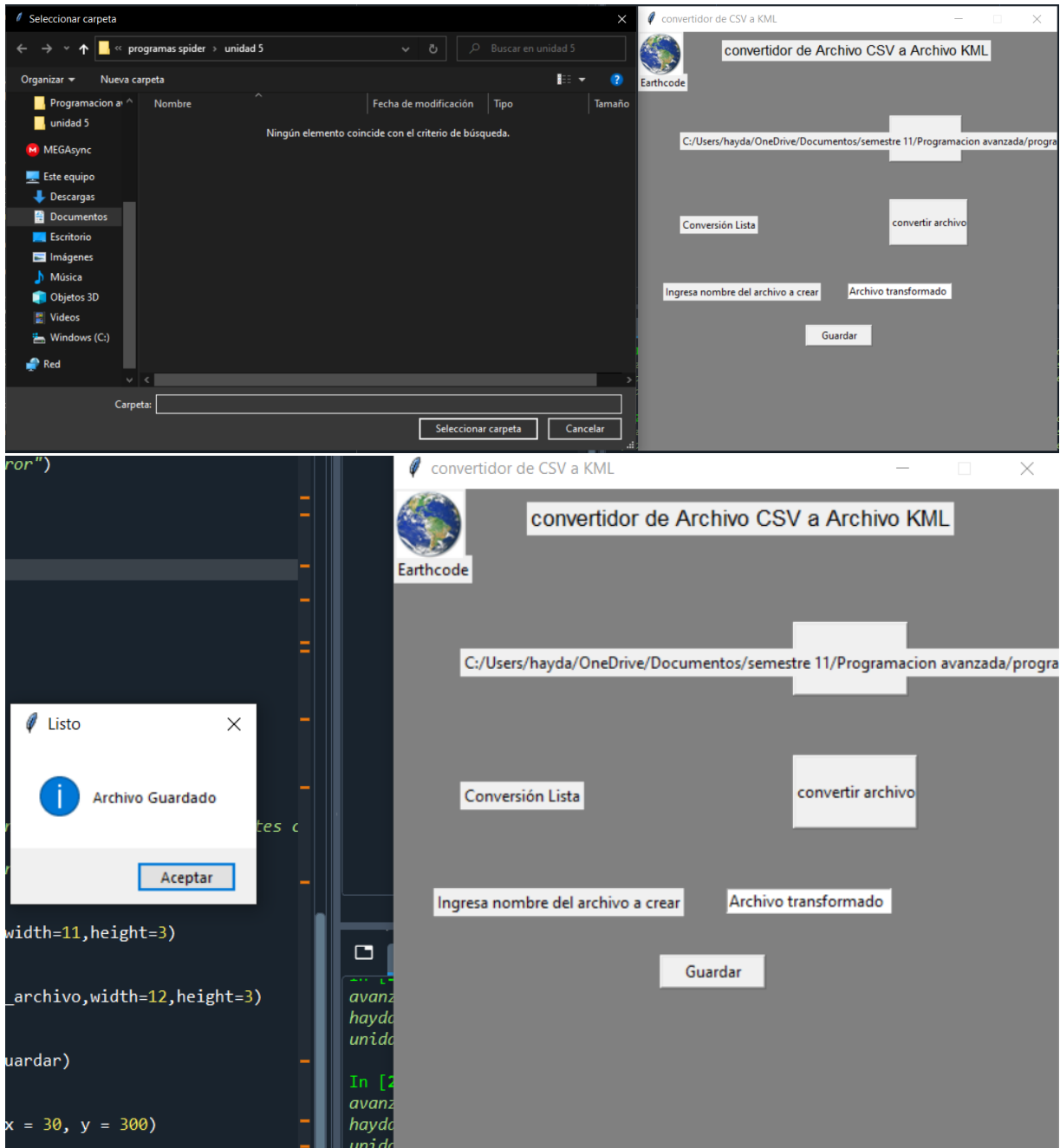


Imagen 3—11: Guardado de archivo convertido

Los errores que pueden generarse debido a errores del operador en la cual pueden aparecer si se coloca caracteres especiales a la hora de asignar un nombre, no cargar algún archivo o cargar un archivo erróneo.

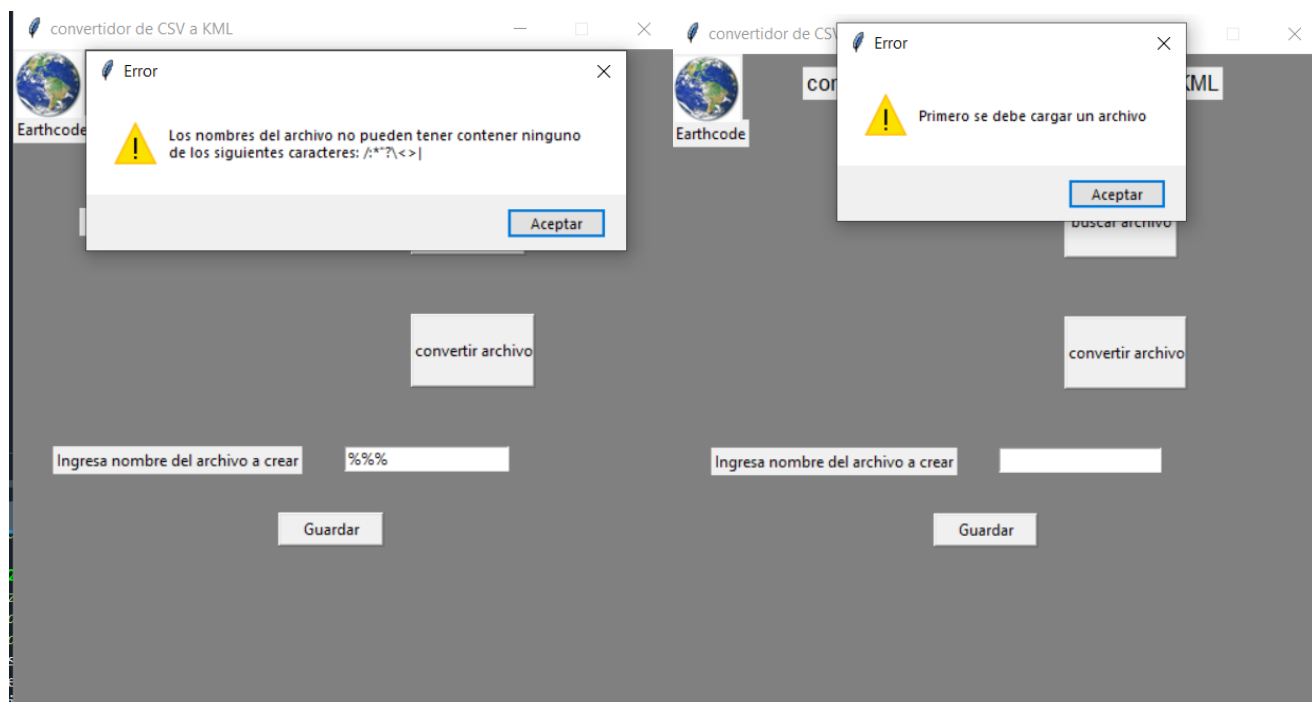


Imagen 3—12: Errores que pueden generarse por el operador

4 CONCLUSIONES PERSONALES

4.1 Conclusión De La Cruz Núñez Brandon Martin

Cad

4.2 Conclusión de Medina Burgos julio Eduardo

Contenido.

4.3 Conclusión de Sánchez Reyes José de Jesús

Contenido.

4.4 Conclusión de Chahin Alapizco Jesus Haydar

Después de analizar varios contenidos de CSV y KML llegamos a la conclusión que la conversión de CSV a KML es muy sencilla si se tiene las bibliotecas adecuadas para ayudar al código a realizar su función principal, y ha partir de ello se puede añadir todo lo demás fundamental para el cumplimiento del programa de la conversión de archivos de formato CSV a KML. En la cual añadimos primero los códigos fundamentales para la selección de archivos CSV a través de la interfaz grafica desarrollada para cumplir el propósito de la transformación

y luego agregar el comando para ejecutar la acción y después de realizarse guardar el archivo con un nombre establecido por el operador. Con este proyecto nos dimos cuenta que para la realización de programas que cumplan incluso funciones sencillas se tiene que tener todo bien estructurado para evitar cualquier tipo de falla posible ya sea por el operado o por el propio programa a la hora de ejecución del código.

4.5 Conclusión de <aquí se pone el nombre completo del alumno iniciando con los apellidos>

Contenido.

5 CITAS BIBLIOGRÁFICAS Y REFERENCIAS

- Fernandez, R. (2019, 30 diciembre). Cómo limpiar/borrar pantalla en Python. ▷ Cursos de Programación de 0 a Experto © Garantizados. Recuperado 27 de noviembre de 2021, de <https://unipython.com/como-borrar-pantalla-en-python/>
- Guzman, H. C. (s. f.). Widget Button (Botón) | Curso de Python | Hektor Profe. hektorprofe. Recuperado 27 de noviembre de 2021, de <https://docs.hektorprofe.net/python/interfaces-graficas-con-tkinter/widget-button-boton/>
- tkinter.messagebox — Tkinter message prompts — Python 3.10.0 documentation. (s. f.). Python. Recuperado 27 de noviembre de 2021, de <https://docs.python.org/3/library/tkinter.messagebox.html>
- Tkinter filedialog askopenfilename function. (2020, 12 mayo). Stack Overflow. Recuperado 27 de noviembre de 2021, de <https://stackoverflow.com/questions/61758460/tkinter-filedialog-askopenfilename-function>
- Ejemplos de asksaveasfile en Python, ejemplos de tkinterfiledialog.asksaveasfile en Python - HotExamples. (s. f.). hotexample. Recuperado 27 de noviembre de 2021, de <https://python.hotexamples.com/es/examples/tkinter.filedialog/-/asksaveasfile/python-asksaveasfile-function-examples.html>
- Tkinter Dialogs — Python 3.10.0 documentation. (s. f.). python. Recuperado 27 de noviembre de 2021, de <https://docs.python.org/3/library/dialog.html>

- Python, R. (2018, 15 agosto). Cuadros de diálogo (messagebox) en Tcl/Tk (tkinter). Recursos Python. <https://recursospython.com/guias-y-manuales/cuadros-de-dialogo-messagebox-en-tkinter/>
- Simplekml :: Anaconda.org. (s. f.). Anaconda. Recuperado 27 de noviembre de 2021, de <https://anaconda.org/conda-forge/simplekml>
- 22 Converting CSV data to a KML file in Python. (2020, 22 octubre). YouTube. Recuperado 27 de noviembre de 2021, de <https://www.youtube.com/watch?v=cEksZtdvg4>
- Instalar y Actualizar Librerías Python - Anaconda | #4 Machine Learning - Python. (2020, 16 abril). YouTube. Recuperado 27 de noviembre de 2021, de <https://www.youtube.com/watch?v=rRPNiWKn0Xg&t=57s>
- 20 Real world examples Generating KML files in Python. (s. f.). YouTube. Recuperado 27 de noviembre de 2021, de <https://www.youtube.com/watch?v=z2GRkuK-uQ>
- 21 Interacting with the user in Python. (2020, 22 octubre). YouTube. Recuperado 27 de noviembre de 2021, de <https://www.youtube.com/watch?v=7p17VbqVgoE>

6 Anexos

1. Para cargar el programa se debe tener la dirección de la imagen tierra1.jpeg que funciona como logotipo del programa en la interfaz gráfica colocada en una dirección específica y modificarlo en el programa para hacerlo funcionar correctamente. En la línea de código 23.

2. Si aparece un error “TclError: image "pyimage1" doesn't exist” después de cambiar la dirección de la imagen para hacer funcionar el programa de la conversión de CSV a KML, solo se tiene que cerrar el Spider y abrirlo nuevamente después de haber guardado la dirección de la imagen en el código fuente.