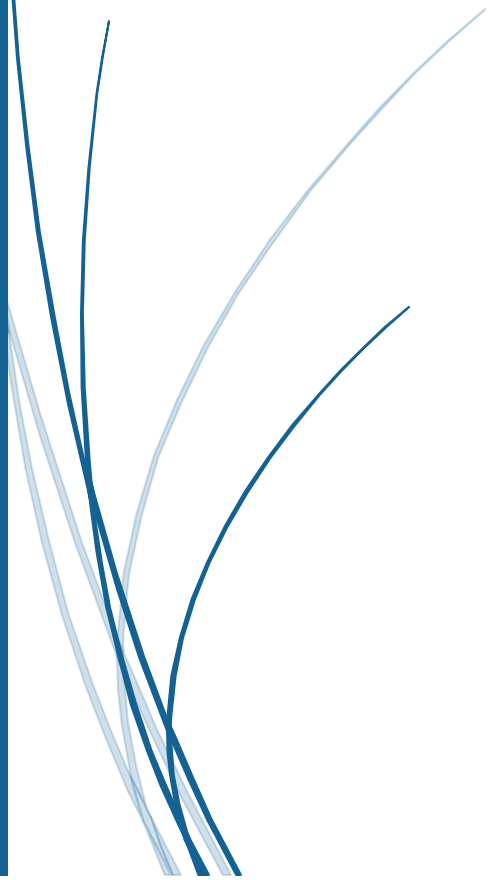
25-9-2016

Documentación Externa

[Proyecto Programado 1]



Estudiante Danny Xie Li - Carnet 2016086098
Profesor William Mata
INSTITUTO TECNOLÓGICO DE COSTA RICA
ESCUELA DE COMPUTACIÓN
CURSO: TALLER DE PROGRAMACIÓN

Tabla de contenidos

1. Enunciado del proyecto

Se va a desarrollar un programa que se implemente un libro de contacto para ser usado en el pseudo-teléfono. Que realizara opciones como modificar, agregar, eliminar y consultar contactos. Se explicará en el manual de usuario, cómo usar esta aplicación.

A) **ACCESO AL PROGRAMA**

Aparcera una ventana en donde usted toca el botón de contactos y se dirigirá a la aplicación.

B) **FUNCIONALIDAD DEL LIBRO DE CONTACTOS**

1. **Ver libro de contactos**

Es la entrada a esta aplicación: va a desplegar el libro de contactos en orden alfabético según el nombre. Inicialmente está vacía. La lista debe permitir un desplazamiento vertical. Por cada contacto se va desplegar una línea con esta información: Imagen y nombre del contacto.

En esta ventana se presenta ventanillas con íconos para realizar las operaciones de mantenimiento o actualización de datos: consultar, modificar y eliminar. Además el botón de agregar se encuentra en la parte superior de la aplicación.

2. **Agregar contactos**

Cuando se quiere agregar un contacto, se toca el botón de agregar y aparecerá una ventana que pedirá los datos del contacto. No se pueden repetir contactos.

3. **Consultar un contacto específico**

Se usa para visualizar todos los datos registrados de un contacto. Se escribe el contacto que desea consultar.

4. Modificar contactos

Esta operación se usa para modificar datos de contactos que ya hayan sido agregados. Se debe escribir el contacto a modificar.

5. Eliminar contacto

Este elimina el contacto del libro de contacto.

C) **Ayuda**

Esta opción despegara el manual de usuario.

D) **Acerca de**

Esta opción se usa para desplegar información "Acerca del programa".

E) **Salir**

Esta opción se usa para salir del libro de contactos, regresa a la ventana inicial donde está el pseudo-teléfono. Para salir del programa también se usa una opción de salir o cerrar.

2. Temas Investigados

Importancia de la documentación de software

Es importante la documentación de los programas para su futuro mantenimiento. Además porque se puede reutilizar una gran parte del código para futuras aplicaciones sin conocer el programa y su funcionamiento. Al cliente se le debe entregar 3 tipos de documentos: el manual de usuario, la documentación técnica y el documento de instalación. Existen dos tipos de documentación: la documentación externa; en esta se escribe en documentos de textos. Y la documentación interna; es la que se escribe dentro del programa o el código, pueden ser en comentarios.

Programación por eventos

Es un paradigma de programación en el que se basa en eventos o sucesos que ocurren en el sistema que son provocados por el usuario. Puede ser secuencial cuando el programador define cual va a ser el flujo del programa y el otro es que el usuario sea el que dirija los eventos. Por ejemplo: Programa Orientado a eventos.

```
While (true){
      Switch (event){
          case mousse_button_down:
          case mouse_click:
          case keypressed:
          case Else:
      }
  }
```

La programación orientada a eventos es la base de la interfaz de usuario, también se usa para desarrollar interfaces en los componentes de software.

Interfaces graficas en Python

De alguna u otra manera ocupamos que el programa pueda tener una forma para recibir entradas y retornar una salida, en las clases nos enseñaron la forma de input (como entrada) y print o return (como salida).

Existen diversas librerías de interfaces graficas como Tkinter, WxPython, PyQT, PyGTK que permite interactuar con el usuario. A continuación se explicara más detalladamente la librería Tkinter, debido a que fue la que se usó en la tarea programada.

Interfaz Gráfica usada: Librería Tkinter de Python.

Elementos Usados:

- **Tk()** : Esta opción le permite al programador crear una ventana.
  ¿Cómo se usa?
  Cuando se desea crear una ventana llama está función y se lo asigna una variable (es recomendado) para futuros usos. Por ejemplo:
  Ventana = Tk ()

- **Variables globales:** Es un tipo de variable libre en la cual se orienta en el ámbito del programa principal.
  ¿Cómo se usa?
  En el programa principal se define la variable global, que debe contener estos 3 elementos: el nombre de la variable, la asignación y el valor. Por ejemplo:
  Lista = [ ]
  Si se quiere usar esta variable en una función se llama esta variable por medio de la palabra global. Por ejemplo:
  global Lista

- **Toplevel (Ventana)**: Esta opción permite al programador crear una pantalla secundaria a la pantalla principal.
  ¿Cómo se usa?

Cuando se desea crear una pantalla secundaria al principal se escribe esa función y entre paréntesis la pantalla principal o la pantalla que desea relacionarlo. Por ejemplo:
VentanaSecundaria = Toplevel (Ventana)

- **Button (opciones)**: Esta opción le permite al programador crear botones y agregarlos a la pantalla, en donde se le puede agregar diferentes opciones como el tamaño de la letra, color del botón, la acción que desea realizar y en que pantalla se debe contener.
  ¿Cómo se usa?
  A la opción Button se le da las opciones entre paréntesisy separados por comas, en donde lo desea ubicar, en que pantalla, color, letra, entre otros, además se le debe empacar el botón para que aparezca en la pantalla de lo contrario no existe. Por ejemplo:
  Botón = Button (Ventana, text ="Haga click").pack ()

- **Label (opciones):** Al igual que el botón, esta opción le permite crear etiquetas de textos o imágenes, posee las mismas opciones que el botón.
  ¿Cómo se usa?
  A la opción de Label se le da unos conjuntos de opciones separados por coma y entre paréntesis todas las opciones que se desea asignar. Además se le debe empacar la etiqueta para que aparezca en la pantalla de lo contrario no existe. Por ejemplo:
  Etiqueta = Label (Ventana, text ="Hola").pack ()

- **Canvas (opciones) y Frame (opciones):** Estas dos opciones le permite al programador crear espacios dentro de una pantalla principal, en la opción de Canvas () le permite crear objetos, agregar botones, etiquetas entre otros en cambio el Frame todo lo mencionado anteriormente con excepción con crear objetos.
  ¿Cómo se usa?
  MiniEspacio = Canvas (Ventana, bg = "blue").pack ()
  MiniEspacio = Frame (Ventana, text = "Hola", fg = "red").pack ()
  En las opciones se deben poner la ventana en la que se quiere insertar y empacarlo o ubicarlo en la pantalla según pixeles o filas y columnas.

- **PhotoImage (file = NombreDelArchivo):** En esta opción le permite al programador agregar imágenes a la pantalla por medio de esta función.

  ¿Cómo se usa?

  La función PhotoImage, entre paréntesis se escribe la palabra file, el símbolo de asignar y el nombre del archivo en formato string. Por ejemplo:

  Imagen = PhotoImage (file ="hola.gif")

- **Entry (opciones):** Esta función le permite al programador agregar campos de textos a la interfaz gráfica. Además para tomar el valor que está en el campo de texto se usa una variable de tipo string o tipo entero.

  ¿Cómo se usa?

  Para crear un campo de texto se le pone las opciones entre paréntesis, la variable y empacarlo.

  Variable = StringVar () #Se define la Variable como un valor string.
  Espacio = Entry (Ventana, Variable).pack () #Se le asigna el campo de texto a una variable.
  Variable.get () #Toma el valor que tiene la variable.
  Variable.set ("Adiós") #Pone un valor a la variable.

- **ttk.ComboBox (opciones):** Esta función crea algo similar a una list box pero no lo es, es un tipo de botón que despliega una tira de valores que le permite al usuario seleccionar un valor de todos los que tiene en esa caja.

  ¿Cómo se usa?

  Se llama la función ttk.ComboBox y entre paréntesis las opciones que desea agregar, Además para agregarle valores se le asigna una lista de valores a la variable values. Por ejemplo:

  Valor = StringVar () #Se le asigna el valor de string a la variable.
  Combo = ttk.ComboBox (Ventana, textvariable = valor, values = ["Casa", "Trabajo", "Otros"]).pack () #Se le asigna el combobox a la variable.
  Valor.get () #Para conocer el valor que se escogió.

- **messagebox.show Tipo (titulo,mensaje):** Esta función le permite al programador mandar un cuadro de texto dependiendo del tipo que se quiere mandar puede ser de error, información, advertencia, entre otros.

  ¿Cómo se usa?

  Se llama a la función escribiendo messagebox.show, el tipo de cuadro de texto que se quiere mandar y entre paréntesis el título del cuadro en formato string y el mensaje que se quiere transmitir también en formato string. Por ejemplo:

  messagebox.showerror ("Error", "Debe escribir un nombre")

- **Radiobutton (opciones):** En esta función le permite al programador crear botones en donde el usuario puede escoger entre las opciones que existe, su forma es de un círculo (botón) y el enunciado.

  ¿Cómo se usa?

  Se escribe la función radiobutton y entre paréntesis las opciones que desea que contenga y además se le debe poner un valor al radiobutton y una variable para poder conocer su valor. Por ejemplo:

  Radiobutton (ventana, text=texto, variable=IntString,value=1).pack()

- **os.system (ubicación, nombre del archivo):** Es un módulo que le permite al programador ejecutar archivos, en el programa de contactos fue usado para ejecutar sonidos o música en formato .wav.

  ¿Cómo se usa?

  Se debe llamar la función de os.system (), y en formato string la ubicación del archivo y el nombre del archivo. Por ejemplo:

  os.system ("start E:\BeethovenMoonlight.wav")

- **ttk.Notebook (opciones):** El propósito de este widget es permitir que el usuario pueda seleccionar ventanas de contenidos haciendo click en los diferentes pestañas.

  ¿Cómo se usa?

  Se llama el widget y entre paréntesis las opciones que desea agregarle.

  nbook = ttk.Notebook(ventana1).pack() #El widget se le asigna a la variable.

```
frame1 = Frame (ventana1) #Se debe crear una ventanilla

tab1=nbook.add (frame1, image = buscar,padding = 10)  #Se le agrega una pestaña al widget.
```

- **Startfile("Nombre del Archivo"):** Esta opción le permite al programador abrir archivos.
  ¿Cómo se usa?

  Se escribe la función y entre paréntesis y en formato string el nombre del archivo que desea abrir. Por ejemplo:

```
startfile("Manual-De-Usuario-DContact.pdf")
```

- **Scrollbar(opciones):** Esta función le permite crear un scrollbar a la ventana, se le debe poner en las opciones la ventana en donde la desea poner y otras opciones si lo desea.
  ¿Cómo se usa?
  Se pone la función, la ventana en donde lo desea ubicar, en que orientación (vertical o horizontal). Por ejemplo:

```
Barra1 = Scrollbar (frame2, orient = "vertical", command = canvas1.yview)
```

Librerías usadas

Antes de usar los elementos de la librería se deben importarlas, con la palabra import:

- ***import random***: Esta librería permite obtener datos aleatorios.
- ***Import sys o import os:*** Esta librería permite accede a funcionalidades del sistema operativo, que nos permite manipular la estructura de directorios. Import startfile: Permite abrir archivos.
- ***import tkinter.ttk as ttk:*** Esta librería contiene diferentes versiones de los elementos standares de Tkinter.
- ***from tkinter import *:*** Esta librería nos permite accede a todos los componentes graficas que posee Tkinter para desarrollar una interfaz gráfica.

- ***from tkinter import messagebox***: Esta libreria le permite al usuario usar cuadros de textos de informacion,error,advertencia,entre otros de la libreria Tkinter.

Opciones de Labels, Buttons, Entrys, ComboBox, ventanas, entre otros.

.title (): Esta opción le permite ponerle un título a la ventana, debe ser en formato string y entre paréntesis. Por ejemplo: ventana.title ("Hola").

.geometry (): Está opción le permite definir el tamaÑo de la ventana ancho x altura en formato string. Por ejemplo: ventana.geometry ("500x100").

.maxsize (): Esta opción le permite definir el tamaño máximo que se puede expandir la ventana. Por ejemplo: ventana.maxsize (ancho,altura).

.iconbitmap (): Esta función le permite al programador agregarle un icono a la ventana, debe ser en formato string y el archivo en formato ico. Por ejemplo: ventana.iconbitmap ("Hola.ico").

Relief =: Esta opcion le permite agregarle relieve a los componentes de tkinter como botones, etiquetas, campos de texto, entre otros. El relieve se debe poner todo en mayúscula. Por ejemplo: relief=FLAT.

Width=: Está opcion le permite definir el tamaño del elemento en dimension a X. Por ejemplo: width = 20.

Height=: Está opcion le permite definir el tamaño del elemento en dimension a Y. Por ejemplo: height = 50.

Bg=: Esta opción le permite al programador definir el fondo de color de los componentes de Tkinter. Se debe poner el color en formato string y en minúscula. Por ejemplo: bg = "blue"

Fg=: Esta opción le permite al programador definir el color de letra ya sea en botones, campos de textos, etiquetas, etc. Por ejemplo: fg = "blue".

Font=: Esta opción le permite al programador definir el tipo de letra, familia de letra. Entre strings el tipo de letra y separado por coma el tamaño. Por ejemplo: Font= ("Helvetica",10)

Image=: Esta opción le permite al programador definir una imagen o agregar una imagen a un elemento, puede ser usados en botones, etiquetas, ventanas, etc. Se le asigna la variable que contiene la imagena la función image. Por ejemplo: Image = Salud.

Command=: Esta opción le permite al programador definir la función que desea realizar el botón. Se le debe asignar la palabra command a una función. Por ejemplo: command=saludar.

Row=: Esta opción le permite ubicar elemento en la posición de la fila que desea empezando desde 0 oeste a este. Esta función solo se usa con .grid().Por ejemplo: grid (row=1,column=0).

Column=: Esta opción le permite ubicar elemento en la posición de la fila que desea empezando desde 0 norte a sur. Esta función solo se usa con .grid().Por ejemplo: grid (row=1,column=0).

Text=: Esta opción le permite mostrar texto en un botón, etiqueta, etc. Debe asignarle a la variable text, el string que desea que se mueste. Por ejemplo: text= "Alo"

Textvariable=: Es un instancia de StringVar()que se asocia cn un texto en un botón, etiqueta, entre otros. Si la variable cambia, el nuevo valor será mostrado en el elemento. Por ejemplo: textvariable = cambio. Para obtener el valor de la variable se usa .get() y para poner un valor a la variable .set().

Herramienta de Debugger en python

La herramienta de depurar consiste en seguir el flujo de un programa a medida que se ejecuta, de manera que podemos observar que es lo que ocurre en cada momento. Es una gran ventaja para el programador para encontrar errores. O también se le conoce como Pdb.

Funciones del "debugger" que trae el IDLE de Python.

- pdb.Pdb().set_trace(): ejecuta paso a paso hasta encontrar el error.
- Go - Ejecuta el resto del código normalmente, o hasta que alcanza un punto de quiebre (break, que será descripto luego).
- Step - Ejecuta una línea de código. Si la línea es una llamada a una función, el depurador ingresará dentro de la función.
- Over - Ejecuta una línea de código. Si la línea es una llamada a una función, el depurador no ingresará dentro de la función.
- Out - Ejecuta líneas de código hasta que el depurador salga de la función en la que estaba cuando se presionó Out. Esto sale de la función.
- Quit - Termina el programa inmediatamente.

Bibliografía

Fuente: http://www.desarrolloweb.com/articulos/importancia-documentacion.html

https://inventwithpython.com/es/7.html

W. John, 2013, Tkinter 8.5 reference: a GUI for Python, Mexico, Computer Center

http://pybonacci.org/2013/06/14/como-depurar-un-programa-python-con-pdb/

https://es.wikipedia.org/wiki/Programaci%C3%B3n_dirigida_por_eventos

3. Conclusiones

Se presentaron problemas con el sonido, que no se podía agregarle sonido y también que al tocar el sonido abriera un programa para escuchar el sonido, se investigó y se dio la solución por medio de los elementos de os y unos tutoriales en internet, de la cual se descubrió que la sintaxis de la función era escribir start, la ubicación del archivo y el nombre del archivo en formato .wav todo entre paréntesis y en formato string.

Además apareció otro problema, era que el scrollbar de la pantalla principal no aparecía el botón o si aparecía pero el botón rebotaba (el botón del scrollbar se mantenía en la misma posición). Pero se solucionó creando un Frame, después un canvas y al final se le agrego un frame más al canvas.

En conclusión con este primer programa pude conocer muchos temas de los que en la clase no se daban, además la sensación de realizar un programa y el tiempo que duré para realizarlo. Además aprendí a agregarle sonidos y a usar la herramienta de Debbuger.

También he adquirido conocimientos acerca de la librería de tkinter, del sistema operativo, de cómo poner cuadros de textos, la función para abrir archivos. Estos conocimientos están aplicados en la tarea programa.

4. Estadísticas de tiempo

Actividad Realizada	Horas
Análisis de requerimientos	1: 30
Diseño de algoritmos	10
Investigación de ...	4:30
Programación	10
Documentación interna	2
Pruebas	6
Elaboración del manual de usuario	1
Elaboración de documentación del proyecto	1
Etc.	6
Total	42 Horas

5. Rúbrica de evaluación

Concepto	Puntos	Puntos obte-	% Avance	Análisis de resultados
Imagen del pseudo-teléfono e ícono para ingresar a las funcionalidades	2		100	
Ver libro de contactos			100	
Lista alfabética de contactos con scroll vertical	5		100	
Scroll vertical	5		100	
Opciones para ingresar a las operaciones de mantenimiento de información	1		100	
Agregar contactos			100	
Validar contactos no repetidos	1		100	
Registro de imagen de contacto	8		100	
Registro de teléfonos (ventana de selección de tipo 3, teléfono 1, varios teléfonos 2)	6		100	
Registro de correos electrónicos (ventana de selección de tipo 3, correo 2, varios correos 2)	7		100	
Registro de sonidos	6		100	
Registro de datos adicionales (ventana de selección de datos 4, notas 1, acerca de la familia 1, profesión1, fecha cumpleaños 2, aniversario matrimonio 2, aniversario trabajo 2, dirección física del trabajo 1, entretenimientos 1, deportes	15		100	
Registro de otros datos (dirección física de la casa 1, trabaja en 1, estudia en 1)	3		100	
Consultar un contacto específico				
Seleccionar contacto	1		80	En vez de seleccionar, puse un campo de texto para que escribiera el nombre
Operación de consulta	5		100	
Modificar contactos				
Seleccionar contacto	1		80	En vez de seleccionar, puse un campo de texto para que escribiera el nombre
Operación de modificación	20		80	Cuando hay datos registrados se cambia por los datos que se le agregó. Sólo puede hacer eso.

Eliminar contactos				
Seleccionar contacto	1		80	En vez de seleccionar, puse un campo de texto para que escribiera el nombre
Operación de eliminación	5		100	
Confirmar eliminación	1		100	
Ayuda (incluye manual de usuario)	5		100	
Acerca de	1		100	
Salir	1		100	
TOTAL	**100**			
Partes desarrolladas adicionalmente			0	

Partes Adicionales: No se desarrollaron partes adicionales.