



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño



Nombre del alumno:

Luis Fernando Ochoa Angulo

Matricula:

372746

Semestre:

3ro- Computación

Grupo:

432

Materia:

Lenguaje de Programación Python

Actividad:

Actividad 14

Librería Tkinter

Nombre del Profesor:

Pedro Nuñez Yepiz

Fecha:Viernes 24 de Noviembre del 2023



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

Ingeniero en computación

Ingeniero en Software y tecnologías emergentes

Materia: Lenguaje de Programación Python / Clave 36276

Alumno: Luis Fernando Ochoa Angulo

Matrícula: 372746

Maestro: Pedro Núñez Yépiz

Actividad No. : 14

Tema - Unidad : Librería Tkinter

Ensenada Baja California a 24 de Noviembre del 2023



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

1. INTRODUCCIÓN

Este informe detalla nuestra experiencia en el desarrollo de interfaces gráficas mediante Tkinter en Python, destacando los aspectos clave desde la creación de ventanas hasta la implementación de interactividad en widgets. Como parte de este proyecto, hemos llevado a cabo la creación exitosa de una calculadora, lo que añade un elemento práctico y aplicado a nuestra exploración de los fundamentos para construir aplicaciones atractivas y funcionales. A lo largo de este documento, compartiremos los desafíos encontrados, las soluciones implementadas y las lecciones aprendidas en el dominio de las interfaces gráficas con Tkinter.

2. COMPETENCIA

Desarrollar habilidades avanzadas en la creación de interfaces gráficas con Tkinter, centrándose en la optimización y mejora de la usabilidad, a través de la implementación de proyectos prácticos, como la creación de una calculadora interactiva.

3. FUNDAMENTOS

En primer lugar, abordamos la creación y estructuración de ventanas y frames, elementos fundamentales que sientan las bases para construir interfaces visuales. A continuación, exploramos la diversidad de widgets básicos que Tkinter ofrece, desde etiquetas hasta botones y cuadros de texto, comprendiendo su implementación para integrar elementos esenciales en la interfaz de usuario. El diseño de la interfaz también se convierte en un aspecto clave, donde aprendemos principios para organizar y presentar widgets de manera intuitiva, mejorando la estética y usabilidad. La gestión de eventos se vuelve crucial para lograr interactividad, permitiendo que la interfaz responda dinámicamente a las acciones del usuario. Además, exploramos el uso de variables y su vinculación con widgets para lograr actualizaciones dinámicas en respuesta a cambios en los datos. La aplicación práctica de estos fundamentos se lleva a cabo a través de la creación de una calculadora funcional, integrando operaciones matemáticas y consideraciones de diseño. A medida que avanzamos, nos sumergimos en opciones de estilo y temas para personalizar la apariencia de la interfaz, y abordamos la optimización del código, la documentación efectiva y las prácticas de prueba y depuración específicas para interfaces gráficas. Estos pasos culminan en un enfoque integral para el desarrollo de aplicaciones atractivas y funcionales con Tkinter.



4. PROCEDIMIENTO

ACTIVIDAD 14

**REALIZA LOS EJERCICIOS EN PYTHON USANDO LIBRERIA
TKINTER**

CALCULADORA BÁSICA

REALIZA UNA APLICACIÓN DE ESCRITORIO, QUE SEA UNA CALCULADORA BÁSICA

 LFOA_Act14_PY_432(anexo).pdf

5. RESULTADOS Y CONCLUSIONES

En conclusión, nuestro recorrido por el desarrollo de interfaces gráficas con Tkinter ha sido integral y enriquecedor. Desde la creación de ventanas hasta la implementación práctica en una calculadora funcional, hemos abordado fundamentos esenciales para construir aplicaciones atractivas y usables. La exploración de widgets, el diseño de interfaz efectivo y la gestión de eventos han sido clave, mientras que la aplicación práctica ha consolidado estos conceptos. La atención a la estética con estilos y temas, junto con la optimización del código y prácticas de prueba, destaca la importancia de crear aplicaciones visualmente agradables y funcionalmente sólidas. Nos ha capacitado para enfrentar desafíos en el desarrollo de interfaces, asegurando aplicaciones eficientes y atractivas.

6. ANEXOS

<https://drive.google.com/file/d/1nZ6kkLyIH9dVb1lBKIJaajiAPxcK-9pc/view?usp=sharing>

ENLACE Github:

<https://github.com/LuisFernando8a/Codigos-Python-432/tree/24e002d34d2d202cd27b2f2cf92388dc05052de9/Practica%2014>



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

7. REFERENCIAS

Diseño de algoritmos y su codificación en lenguaje C

Corona, M.A. y Ancona, M.A. (2011)..

España: McGraw-Hill.

ISBN: 9786071505712

Programación estructurada a fondo: implementación de algoritmos en C

:Pearson Educación.Sznajdleder, P. A. (2017)..

Buenos Aires,Argentina: Alfaomega

Como programar en C/C++

H.M. Deitel/ P.J. Deitel

Segunda edición

Editorial: Prentice Hall.

ISBN:9688804711

Programación en C.Metodología, estructura de datos y objetos

Joyanes, L. y Zahonero, I. (2001)..

España:McGraw-Hill.

ISBN: 8448130138