**Modelo de Diseño**

**Comparador de PDF**

Versión 3.1

**Bitácora de Cambios**

|  |  |  |
| --- | --- | --- |
| **Fecha** | **Descripción del Cambio** | **Autor del Cambio** |
| 02/04/2024. | Inicio del documento | Leonardo Miranda Rico |
| 09/04/2024. | Llenado de tablas y revisión del código | Leonardo Miranda Rico  Jorge Emilio Gutiérrez Cortes |
| 12/04/2024. | Corrección de errores | Leonardo Miranda Rico |

**Tabla de Contenido**

[**I. Introducción 4**](#_heading=h.qsh70q)

[**II. Criterios 4**](#_heading=h.3as4poj)

[**Arquitectura 4**](#_heading=h.1pxezwc)

[**Herramientas 4**](#_heading=h.49x2ik5)

[**III. Inventario de Artefactos 4**](#_heading=h.2p2csry)

[**IV. Restricciones de la Arquitectura 5**](#_heading=h.147n2zr)

[**V. Criterios de Orden de Construcción 5**](#_heading=h.3o7alnk)

[**Definición de Criterios Para el Orden de Construcción 5**](#_heading=h.23ckvvd)

[**Orden de Construcción en Base a la Prioridad 5**](#_heading=h.ihv636)

[**VI. Ejecución de Casos de Uso 6**](#_heading=h.32hioqz)

[**<Nombre de la ejecución de caso de uso> 6**](#_heading=h.1hmsyys)

[**Artefactos involucrados** 6](#_heading=h.41mghml)

[**Descripción** 6](#_heading=h.2grqrue)

[**VII. Definición de las Pantallas del Sistema 7**](#_heading=h.vx1227)

[**Diagrama de Navegación de Pantallas 7**](#_heading=h.3fwokq0)

[**Definición de Pantallas 8**](#_heading=h.1v1yuxt)

[**Pantalla: Inicio de Sesión** 8](#_heading=h.4f1mdlm)

[**Objetivo de la Pantalla** 8](#_heading=h.2u6wntf)

[**Descripción de la Pantalla** 8](#_heading=h.19c6y18)

**I. Introducción**

El análisis de textos es una tarea fundamental en diversos campos, desde la investigación académica hasta el procesamiento de datos en empresas. El objetivo del software es identificar y extraer palabras o frases que coincidan con un diccionario dado en otro documento PDF; generando como resultado un archivo con las coincidencias encontradas siendo una herramienta diseñada para facilitar el análisis de textos en formato PDF.

**II. Criterios**

Arquitectura

El software se diseñó bajo la primicia de crear un comparador de archivos PDF, sencillo y fácil de usar. Con la finalidad de que este sea un programa rápido y eficaz y que solo realice el procedimiento que el usuario indique.

Herramientas

El software fue diseñado en el lenguaje de programación Python con la ayuda de librerías externas para la realización de ciertos procesos como: MinerPDF, una herramienta utilizada en la extracción de texto para documentos PDF,Customkinter, herramienta de interfaz de usuario de Python personalizable basada en Tkinter y CTkMessagebox, herramienta utilizada para cuadro de mensajes moderno y totalmente personalizables.

**III. Inventario de Artefactos**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre** | **Tipo** | **Archivo** | **Descripción** |
| Comparador de PDF versión 1.1 | Software | .PY | Se desarrollo la primera versión de código |
| Archivo de documentación 1 | Documento de texto | De texto | Se inicio un archivo con la documentación respectiva a la primera versión del software |
| Archivo de documentación 2 | Documento de texto | De texto | Se completaron los puntos faltantes del archivo de documentación en base a la primera versión del código |
| Repositorio en GitHub 1 | Diagrama y archivo de texto | Git | Se alojo la primera versión del código en GitHub y se inicio el diagrama para llevar control de las versiones desarrolladas. |
| Comparador de PDF versión 2.1 | Software | .PY | Se corrigieron funciones en base a la primera versión del software. |
| Archivo de documentación 3 | Documento de texto | De texto | Se corrigieron errores en la documentación en base a la segunda versión del software. |
| Repositorio en GitHub 2 | Diagrama y archivo de texto | Git | Se alojo la segunda versión del código y se subió la documentación, así como actualización del diagrama. |

**IV. Restricciones de la Arquitectura**

El software debía proporcionar una interfaz de usuario intuitiva y fácil de usar que permitiera a los usuarios cargar documentos PDF, así como ver los resultados de análisis, esto se realizo con la ayuda de las librerías Customkinter, herramienta de interfaz de usuario de Python personalizable basada en Tkinter y CTkMessagebox, herramienta utilizada para cuadro de mensajes moderno y totalmente personalizables.

El software debía ser compatible con dispositivos de entrada estándar, como teclados y ratones, para permitir a los usuarios interactuar con la interfaz de usuario y realizar acciones dentro del software, y con monitores de pantalla estándar para garantizar una visualización adecuada de la interfaz de usuario y los resultados del análisis; así como interactuar con el sistema de archivos local del usuario para permitir la carga y guardado de documentos en formato PDF. Esto fue posible gracias a la compatibilidad de Python con los distintos sistemas operativos y el hardware con el cual trabajan.

Por último una documentación clara y soporte técnico para ayudar a los usuarios a comprender y utilizar eficazmente el software.

**V. Criterios de Orden de Construcción**

Definición de Criterios Para el Orden de Construcción

|  |  |
| --- | --- |
| **Prioridad** | **Criterios** |
| 1 | Análisis de texto, que nuestro software sea capaz de leer el texto dentro de los archivos PDF. |
| 2 | Carga de documentos tipo PDF, que nuestro software sea capaz de recibir los dos archivos PDF a comparar. |
| 3 | Generar archivo de coincidencias, que nuestro software sea capaz de redactar las coincidencias en un archivo tipo txt. |
| 4 | Interfaz, que el software cuente con una interfaz fácil de usar para el usuario permitiendo el uso satisfactorio del software. |

Orden de Construcción en Base a la Prioridad

|  |  |
| --- | --- |
| **Prioridad** | **Componente** |
| 1 | Interfaz intuitiva |
| 2 | Carga de documentos tipo PDF |
| 3 | Análisis de textos |
| 4 | Generar archivo de coincidencias |

**VI. Ejecución de Casos de Uso**

**Comparador de PDF**

En este caso este es un ejemplo de un caso de uso, en el cual se compara el archivo de entrada con el diccionario de palabras ambos otorgados por el usuario. Dándonos como resultado el archivo de salida con las coincidencias encontradas.

**Artefactos involucrados**

En este caso intervienen todas las partes del software como lo son:

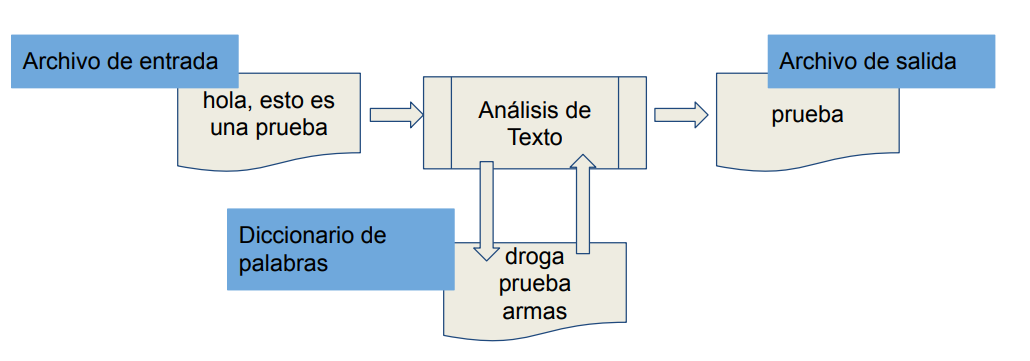
1. Interfaz de Usuario

2. Cargar Documentos PDF

3. Análisis de Texto

4. Generar Archivo de Coincidencias

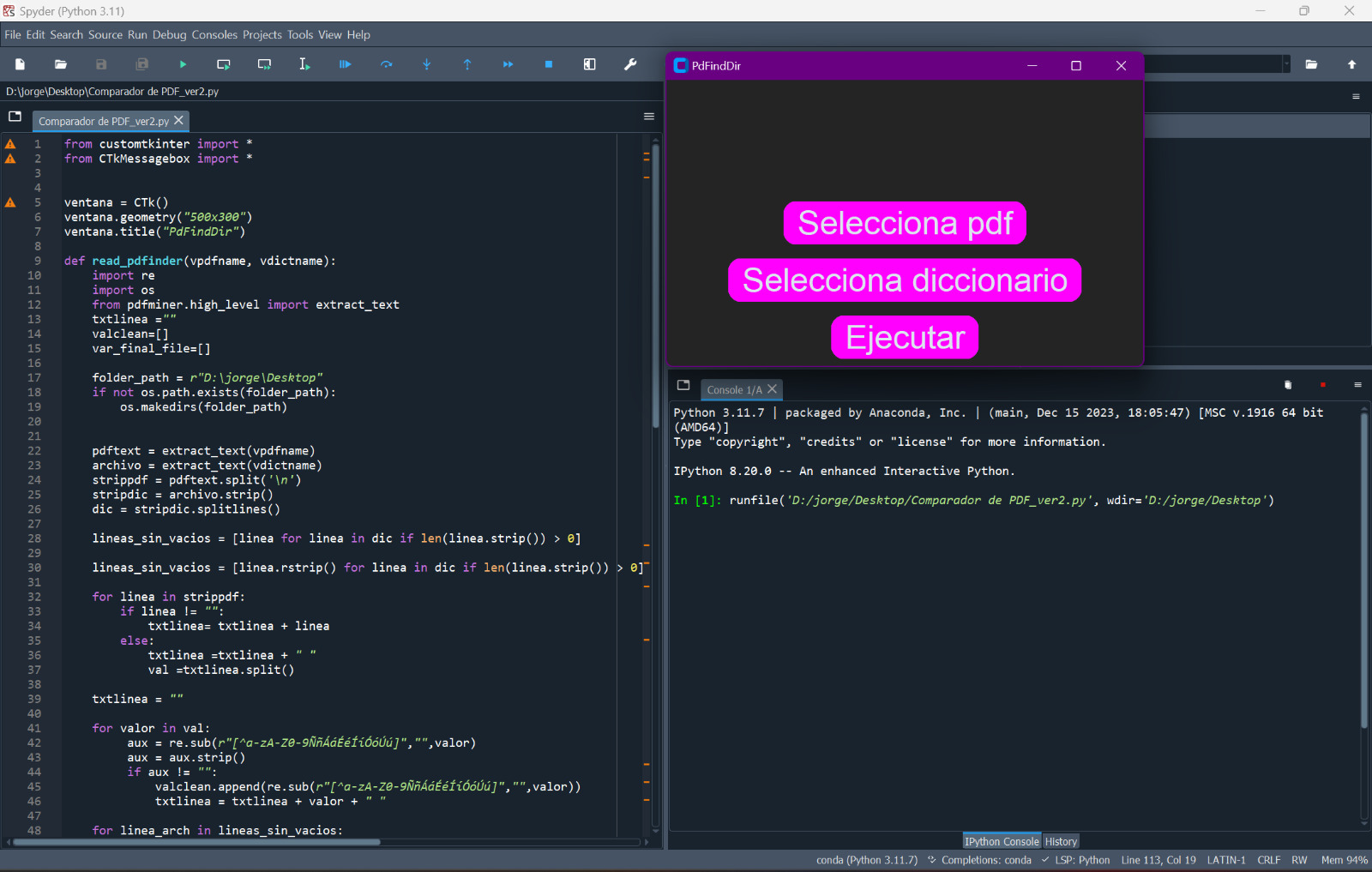
**Descripción**



**VII. Definición de las Pantallas del Sistema**

Definición de Pantallas

**Pantalla: Interfaz**



**Objetivo de la Pantalla**

Esta pantalla es el centro de mando del software aquí cargaremos nuestros archivos PDF a comparar y ejecutaremos el análisis de texto con fin de obtener nuestro archivo de coincidencias.

**Descripción de la Pantalla**

|  |  |  |
| --- | --- | --- |
| **Campo** | **Tipo** | **Comentario** |
| Seleccionar PDF | Botón | Al presionar este botón se abrirá el seleccionador de archivo de nuestro equipo para cargar nuestro archivo PDF. |
| Seleccionar Diccionario | Botón | Al presionar este botón se abrirá el seleccionador de archivo de nuestro equipo para cargar nuestro diccionario en archivo PDF. |
| Ejecutar | Botón | Al presionar este botón se iniciará el análisis de texto y se empezaran a escribir las coincidencias en nuestro archivo de resultados. |