**Documento de requerimientos de software**

***Comparador de texto***

***Fecha: [2/04/2024]***

***Piña Daniel***

***Cruz Jesus***

***Mallorga Esteban***

**Tabla de contenido**

Historial de Versiones 3

Información del Proyecto 3

Aprobaciones 3

1. Propósito 4

2. Alcance del producto / Software 4

3. Referencias 4

4. Funcionalidades del producto 5

5. Clases y características de usuarios 5

6. Entorno operativo 5

7. Requerimientos funcionales 6

9.1. (Nombre de la funcionalidad 1) 6

9.2. (Nombre de la funcionalidad 2) 7

9.3. (Nombre de la funcionalidad N) 7

8. Reglas de negocio 8

9. Requerimientos de interfaces externas 9

9.1. Interfaces de usuario 9

9.2. Interfaces de hardware 9

9.3. Interfaces de software 9

9.4. Interfaces de comunicación 9

10. Requerimientos no funcionales 10

11. Otros requerimientos 11

12. Glosario 12

**Historial de Versiones**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Fecha** | **Versión** | **Autor** | **Organización** | **Descripción** |
| 1/04/24 | 1 | Esteban | Codigo | Genero la primera version del codigo |
| 1/04/24 | 1 | Daniel | Documentos | Primer paquete de archivos |
| 1/04/24 | 1 | Jesus | Creacion del repositorio | Creacion del repositorio en github |
|  |  |  |  |  |

**Información del Proyecto**

|  |  |
| --- | --- |
| Empresa / Organización | Proyecto Final |
| Proyecto | Parceador de archivos |
| Fecha de preparación | 1/04/2024 |
| Cliente | Profesor Gerardo |
| Patrocinador principal | Equipo del proyecto |
| Gerente / Líder de Proyecto | Equipo Proyecto |
| Gerente / Líder de Análisis de negocio y requerimientos | Equipo Proyecto |

**Aprobaciones**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nombre y Apellido** | **Cargo** | **Departamento u Organización** | **Fecha** | **Firma** |
| Jesus Cruz | Creador del repositorio | Github |  |  |
| Esteban | Creador Codigo | Python |  |  |
| Daniel | Creador de documentos | Docs |  |  |
|  |  |  |  |  |

1. **Propósito**

El proposito de este proyecto es generar un repositorio que debe contener la documetacion requerida asi como el codigo que debe compar una pdf y generar de la misma forma un con las palabras o frases ya encontradas.

1. **Alcance del producto / Software**

* Generar un comparador de pdf funcional que cumpla con los requerimietos proporcionados por el profesor..

1. **Referencias**

Aquí se pueden incluir otros documentos impresos, documentos electrónicos o direcciones electrónicas que complementen la documentación de requerimientos de software, por ejemplo: Documentos de visión, definición de alcance, otros documentos de especificación de requerimientos de software, flujogramas, políticas, procedimientos de la organización, entre otros.

Para cada referencia es recomendable incluir el título, autor, versión, fecha y ubicación física o electrónica.

1. **Funcionalidades del producto**

1.- Recibir un pdf

2.- Leer y comparar las palabras o frases que se tiene en el pdf

3.- Crear otro pdf solo con las palabras que ya son iguales

1. **Clases y características de usuarios**

1.- Prueba de regresión: Se realizo una prueba de regresion ya que entregamos un codigo al profesor y solo podia comparar palabras y en la actualizacion del codigo ya podia leer frases.

1. **Entorno operativo**

Esperamos que nuestro codigo pueda funcionar en una maquina que tiene incluido windows.

1. **Requerimientos funcionales**

**7.1 (Recibimiento de PDF)**

Descripción: Recibimiento del archivo.

Prioridad: Nivel alto..

Acciones iniciadoras y comportamiento esperado: El archivo proprocionado por el profesor sea aceptado por codigo

REQ-1:Compatibilidad con windows.

REQ-2:El pdf sea recibido por el codigo

**7.2 (Parceo del pdf propocionado)**

Prioridad: Nivel alto..

Acciones iniciadoras y comportamiento esperado: El archivo sea recibido y se inicie el parceo de los que contenga el mismo archivo.

Requerimientos funcionales: El codigo reciba el pdf y el mismo codigo pueda parecear el documento .

Prioridad: Nivel alto..

Acciones iniciadoras y comportamiento esperado: El profesor proporciona el Pdf o el archivo y es recibido por el código.

Requerimientos funcionales: Que el sistema se compatible con winows.

REQ-1:Compatibilidad con windows.

REQ-2: Correcto recibimiento del archivo por el codigo

REQ-3: El parceo sea posible en el codigo

**7.3 (Salida de pdf ya parceado)**

Requerimientos funcionales: El codigo reciba el pdf y el mismo codigo pueda parecear el documento, salida del documento ya parceado

Prioridad: Nivel alto.

Acciones iniciadoras y comportamiento esperado: que el codigo ya allá parceado el archivo previamente asignado y logre mandar un pdf de salida con el texto ya parceado.

Requerimientos funcionales: Que el sistema sea compatible con Windows .

REQ-1:Compatibilidad con windows.

REQ-2: Correcto recibimiento del archivo por el codigo

REQ-3: El parceo sea posible en el codigo

REQ-4: El codigo logre parcear y logre generar el archivo de salidad ya parceado

1. **Reglas de negocio**

Cada uno de los participantes cumpla con lo asignado del proyecto

1. **Requerimientos de interfaces externas**
2. **Interfaces de usuario**

La intefaz usada sera el cmd

1. **Interfaces de hardware**

Sera compatible solo con sistema windows

1. **Interfaces de software**

Las librerias usadas fueron de python

1. **Interfaces de comunicación**

Usamos como interfaces de comunicacion github y whattsap

1. **Requerimientos no funcionales**

Que el sistema se funcional en cualquier ordenador.

1. **Otros requerimientos**

El unico requerimiento requerido es el archivo proporcionado por el profesor.

1. **Glosario**

.