

**SEGUNDA ESPECIFICACION FUNCIONAL:** Que el programa que estamos diseñando use el diccionario para compararlo junto con el archivo de entrada.

**INTRODUCCION:** En este modelo de diseño que esta especialmente hecho para que el código que estamos realizando, use el diccionario para compararlo junto con el archivo de entrada.

**CRITERIOS:**

1. **ARQUITECTURA:** Elegimos un sistema Modular ya que este mismo, ya que divide las diferentes funciones del sistema de manera independiente, y mas centralizados.
2. **HERRAMIENTAS:** De Diseño empleamos un sistema PDF, ya que este es imposible de manipular y es más accesible en términos para que el sistema lo lea. Y para Documentar la información, decidimos hacerlo por "GITHUB", ya que esta misma posee ciertos elementos que requerimos y necesitamos para la documentación.

**INVENTARIO DE ARTEFACTOS:**

Nombre	Tipo	Archivo	Descripción
Col-TEST	Diagrama de Ejemplo funcional del código.	"F:\proyecto analizador de textos\Ejemplo proyecto final.pdf"	En este diagrama se describe la forma en la que debe de funcionar el proyecto de manera adecuada, y se encuentra la primera especificación funcional y su función principal.
Col-EF2	Diagrama de la segunda E. Funcional.	"F:\proyecto analizador de textos\F2Diagrama.pdf"	En este diagrama se describe la forma en como el diccionario sirve como comparador junto con el archivo de entrada, para así que un archivo de salida salga con los resultados esperados.

**RESTRICCIONES DE LA ARQUITECTURA:**

- I. **Administración de la seguridad del sistema:** El archivo PEDF numero 2 debe ser traído por el profesor, es bastante seguro ya que es en PDF, y este no permite que exteriores cambien su contenido.
- II. **Cuestiones de portabilidad:** La portabilidad es bastante bueno ya que se puede trasladar en la nube, USB o simplemente en un correo electrónico, o en la misma PC se encuentra el archivo.
- III. **Escalabilidad:** Hay mucha oportunidad de escalar ya que puedes meter en el código cualquier tipo de archivo PDF.

**IV. Alta disponibilidad del sistema:** Es bastante accesible ya que cualquier PC, lee PDF, y con eso el programa puede leerlo.

#### CRITERIOS DE ORDEN DE CONSTRUCCION:

Prioridad	Criterios
1	- Archivos PDF. - Especificación del código para que lea el segundo archivo (diccionario)...
2	- Archivos PDF tanto en entrada, diccionario y salida. - ...
3	- ...

Prioridad	Componente
1	Palabras, números. Símbolos claves en el texto.
1	Modulo de entrada, el modulo del diccionario, correr el analizador de textos, y un archivo de salida para el resultado.
2	Cifrado exacto para acceder a los archivos PDF.
2	...
3	...

#### EJECUCION DE CASOS DE USO:

##### Nombre de la ejecución de caso de uso>

En este escenario solo comprobamos que el código lea el archivo PDF traído por el profesor, y este funcione para que este mismo sea comparado con el archivo de entrada y haga un archivo de salida.

##### Artefactos involucrados:

"F:\proyecto analizador de textos\Ejemplo proyecto final.pdf"

"F:\proyecto analizador de textos\F2Diagrama.pdf"

##### Descripción:

En estos diagramas podemos ver la función del código, pero sobre todo en el segundo el escenario que decidimos probar en esta prueba.

#### DEFINICION DE INTERFACES DEL SISTEMA:

