

Firmex, prueba técnica: Análisis de aceptación de documento, en GO

Arturo Mantinetti, Fabián Miranda Muñoz

I. DESARROLLO

El objetivo de esta prueba, es la de diseñar y construir un microservicio en **Golang** que utilice una base de datos en **postgres**, encargado de analizar si un documento, **pdf**, puede o no ser aceptado según ciertos criterios que serán recibidos junto con el mismo por el uso de una API.

Se recibiría un objeto por el método POST /VALIDATE_DOCUMENT que traería:

```

1 {
2   "document": "base64 document",
3   "document_id": "uuid",
4   "required_strings": [
5     [
6       "12.874.354-2",
7       "12874354-2"
8     ],
9     [
10      "12.874.354-2",
11      "12874354-2"
12    ]
13  ],
14  "validate_strings": [
15    "Contrato de arriendo",
16    "Juan Benjamín Miranda"
17  ]
18 }
```

- document: Documento PDF en base64
- document_id: El identificador que va asociado exactamente a este documento en la BDD a utilizar
- required_string: Estos, son los strings que deben estar presentes en el documento. Se corresponde a un arreglo y cada elemento puede ser también un arreglo de strings que se debe evaluar en su totalidad.
- validate_strings: Este arreglo de strings, contiene aquellos que deban buscar no sólo su presencia, sino su grado de similitud en el documento mismo.

Por un lado, dado el input de ejemplo anterior, se debe tener una respuesta de salida similar a la siguiente propuesta:

```

1 {
2   "document_id": "uuid",
3   "required": false,
4   "required_strings": [
5     true,
6     false
7   ],
8   "validate_strings": [
9     80,
10    90
11  ],
12  "validated": 85
13 }
```

- document_id: Se corresponde al mismo identificador recibido anteriormente.
- required: Es el ponderado de booleanos de required_strings, teniendo como mayor peso el valor false. O sea, si existe un solo false, el resultado de required será false.
- required_strings: Dentro de cada elemento del arreglo principal, se ve si alguno dentro de cada uno (que puede ser lista) está o no presente. Por ejemplo, el primero elemento es ["12.874.354-2","12874354-2"], teniendo como resultado que como uno de ambos estaba presente, entonces el primer elemento de required_strings sería true.

- `validate_strings`: Aquí, se responde el porcentaje de similitud conseguido por cada string definido. En este caso, Contrato de arriendo tuvo un puntaje de 80.
- `validated`: Es la media de `validate_strings`.

Adicionalmente, se requerirá un método GET que posibilite obtener alguno de los resultados procesados anteriormente para un `document_id` específico.

II. ENTREGA

La prueba debe ser liberada en un repositorio de github.com, el cuál debe ser compartido a los usuarios **amantinetti** y **mclfabian**. Puede usar el branching strategy de su preferencia.

El tiempo límite para realizar la prueba es de una semana, desde el día de envío del correo hasta las 23:59Hrs del último día (Ej: Recepción, lunes 4 de octubre a las 12:00Hrs; Entrega, lunes 11 de octubre a las 23:59hrs).