PdfLogicalExtractor

Autor	Ángel Ibáñez Hernández
Versión	1.0.1
Fecha	20 de mayo de 2024

www.angelves.com

1 – Descripción General

PdfLogicExtractor es una pieza de sofware diseñada para extraer información de documentos pdf de manera lógica y ordenada, de tal forma que pueda después ser procesada por los sistemas en que se integre.

El sistema se basa en lógica adaptable implementada en un sistema de plantillas que puedan procesar todos documentos de un determinado tipo.

Una plantilla es capaz de adaptarse a las variaciones establecidas en su definición, como por ejemplo meses con diferente número de días, zonas del documento desplazadas o diferencias entre páginas dentro de un mismo documento. Y en general cualquier tipo de lógica de extracción que se necesite.

La lógica de extracción de plantilla puede ejecutar limpieza de resultados en base a reglas predefinidas, obteniendo tipos de dato puros o eliminando partes de textos poco significativas.

La lógica de extracción de plantilla puede efectuar cálculos en base a resultados obtenidos de diferentes extracciones o valores predefinidos, como por ejemplo calcular totales en una tabla en base a precio por unidades.

Así mismo, se pueden programar ex profeso todas las funcionalidades o excepciones que la lógica de plantilla de un tipo de documento requiera para resultar una herramienta más efectiva.

2 – Requisitos del Sistema

Esta pieza de software se ha programado en Microsoft .NET Standart 2.1, garantizando compatibilidad a largo plazo y entre diferentes plataformas de Microsoft. Siendo está opción la que ofrece máxima integración.

El sistema requiere de un software externo que realice las llamadas, seleccionado el documento y plantilla de procesados, y que reciba los datos tanto por integración directa, como en un formato estándar JSON, que es el formato que usan las API modernas en su interactuación con sistemas externos.

Por la propia idiosincrasia del software, capaz de integrarse en APIs, esta herramienta puede ser utilizada también por la práctica totalidad de plataformas modernas que se encuentran en el mercado, ya que las APIs establecen una capa de aislamiento y seguridad que no requiere integración.

Una API que use PdfLogicExtractor como una dll interna puede estar instalada en un único equipo, dentro de una I.I.S. local, en un servidor remoto o integrado en su intranet, o bien puede publicarse en Cloud compatible con .NET, que es la práctica totalidad del mercado (Azure, AWS, etc.).

3 – Integración

El sistema está contenido en una librería de enlace dinámico o dll, incluida en un nuget, que se puede incoporar a cualquier tipo de plataforma o proyecto de software del universo .NET.

La integración directa en un proyecto .NET puede hacerse desde el gestor de nuget de Visual Studio en el propio proyecto.

La integración define una única interfaz de software muy sencilla con unas pocas llamadas sobrecargas y un modelo de datos de respuesta accesible directamente, que puede procesarse fácilmente como respuestas JSON.

La integración mediante API requiere una API compatible con la arquitectura .NET donde poder integrar la dll/NuGet, que se utilizaría a nivel interno tal y como se ha descrito en el apartado anterior, devolviendo respuestas estándar JSON que se pueden integrar con la práctica totalidad de plataformas del mercado, escritas en .NET o no (Java Android, PHP, C++, etc.)

Puede integrar el software mediante el gestor de nuget de VS buscando:

Angelves.PdfLogicalExtractor

O usando el siguiente comando en la consola:

```
PM> NuGet\Install-Package Angelves.PdfLogicalExtractor -Version 1.0.2
```

Pueden ustedes encontrar más documentación en:

www.angelves.com/PdfLogicalExtractor

Pueden ustedes probar el software integrado en nuestra API en:

www.angelves.com/PdfLogicalExtractor/TestOnlineFree

4 – Ejemplos

Interfaz C#:

```
namespace Angelves.PdfLogicalExtractor.PublicInterface
{
    public interface ILetsGoToExtraction
    {
        ExtractionResult Start(Template template, string filePath, DocumentType type);

        ExtractionResult Start(string templatePath, string filePath, DocumentType type);

        ExtractionResult Start(string templatePath, Stream fileStream, DocumentType type);

        ExtractionResult Start(Template template, Stream fileStream, DocumentType type);

        string GetWordsInPdf(string filePath, string? filter1 = null, int round = 0);
    }
}
```

Ejemplo de plantilla:

```
{
  "templatename": "Example Template",
  "config": {
    "externalsfieldsintables": false,
    "decimalseparator": ","
  },
  "offsets": [
    {
      "id": 1,
      "text": "PROGRAMA",
      "x": 82.92,
      "y": 153.95
    },
    {
      "id": 2,
      "text": "ANUNCIANTE:",
      "x": 82.92,
      "y": 101.99
    }
  ],
  "metaboxes": [
    {
      "name": "end_month_1",
      "type": "Number",
      "x1": 714,
      "y1": 146,
      "x2": 723,
      "y2": 153
    },
```

```
    {
      "name": "end_month_2",
      "type": "Number",
      "x1": 705,
      "y1": 146,
      "x2": 714,
      "y2": 143
    },
    {
      "name": "end_month_3",
      "type": "Number",
      "x1": 696,
      "y1": 146,
      "x2": 705,
      "y2": 153
    }
  ],
  "boxes": [
    {
      "name": "emisora",
      "required": true,
      "x1": 600,
      "y1": 60,
      "x2": 800,
      "y2": 80
    },
    {
      "name": "anunciante",
      "idoffset": 2,
      "required": true,
      "x1": 160,
      "y1": 90,
      "x2": 350,
      "y2": 110
    },
    {
      "name": "producto",
      "idoffset": 2,
      "required": true,
      "x1": 160,
      "y1": 106,
      "x2": 350,
      "y2": 116
    },
    {
      "name": "campaign",
      "type": "Empty"
    },
    {
      "name": "referencia",
      "idoffset": 2,
      "required": true,
      "x1": 600,
      "y1": 90,
      "x2": 800,
      "y2": 110,
      "extractionrules": [
        {
          "action": "QuitSpaces"
        },
        {
          "action": "Erase",
          "target": "N°ORDEN:"
```

```
      }
    ]
  },
  {
    "name": "fechafactura",
    "type": "DateTime",
    "idoffset": 2,
    "required": true,
    "format": "dd/MM/yyyy",
    "x1": 600,
    "y1": 106,
    "x2": 800,
    "y2": 125,
    "extractionrules": [
      {
        "action": "Erase",
        "target": "FECHA:"
      },
      {
        "action": "QuitSpaces"
      }
    ]
  },
  {
    "name": "table1",
    "type": "Table",
    "header": [
      {
        "name": "formato",
        "idoffset": 1,
        "required": true,
        "master": true,
        "x1": 235,
        "y1": 171.10,
        "x2": 273,
        "y2": 178.41,
        "extractionrules": [
          {
            "action": "QuitSpaces"
          },
          {
            "action": "Erase",
            "target": "20"
          },
          {
            "action": "Erase",
            "target": "\""
          }
        ]
      },
      {
        "name": "duracion",
        "idoffset": 1,
        "required": true,
        "x1": 235,
        "x2": 280,
        "extractionrules": [
          {
            "action": "QuitSpaces"
          },
          {
            "action": "Erase",
            "target": "CUÑA"
```

```
        },
        {
          "action": "Erase",
          "target": "\""
        }
      ]
    },
    {
      "name": "programa",
      "idoffset": 1,
      "required": true,
      "x1": 70,
      "x2": 180
    },
    {
      "name": "hora_iniciofin",
      "type": "TextSplit",
      "idoffset": 1,
      "parameters": [ "-" ],
      "x1": 180,
      "x2": 235,
      "extractionrules": [
        {
          "action": "QuitSpaces"
        },
        {
          "action": "Erase",
          "target": "("
        },
        {
          "action": "Erase",
          "target": "LV"
        },
        {
          "action": "Erase",
          "target": "S-D"
        },
        {
          "action": "Erase",
          "target": ")"
        }
      ]
    },
    {
      "name": "swap",
      "type": "ArraySwap",
      "idoffset": 1,
      "additionaldata": "Number",
      "deletenullsinarray": true,
      "x1": 444.21,
      "x2": 723.43,
      "w": 9.007,
      "arrayindex": {
        "y1": 146,
        "y2": 153
      },
      "alternativecoords": [
        {
          "condition": "[BOX:end_month_1] ! 31",
          "x2": 714.42
        },
        {
          "condition": "[BOX:end_month_2] ! 30",
```

```
            "x2": 705.41
          },
          {
            "condition": "[BOX:end_month_3] ! 29",
            "x2": 696.41
          }
        ]
      },
      {
        "name": "total_pases",
        "type": "Decimal",
        "idoffset": 1,
        "x1": 310,
        "x2": 350
      },
      {
        "name": "preciounitario",
        "type": "Decimal",
        "formula": "[BOXROW:preciototal] / [BOXROW:total_pases]"
      },
      {
        "name": "descuento",
        "type": "Decimal",
        "idoffset": 1,
        "x1": 380,
        "x2": 410,
        "extractionrules": [
          {
            "action": "Erase",
            "target": "%"
          },
          {
            "action": "QuitSpaces"
          }
        ]
      },
      {
        "name": "descuentoagencia",
        "type": "Empty"
      },
      {
        "name": "preciototal",
        "type": "Decimal",
        "idoffset": 1,
        "required": true,
        "x1": 410,
        "x2": 445,
        "extractionrules": [
          {
            "action": "Erase",
            "target": "€"
          },
          {
            "action": "QuitSpaces"
          }
        ]
      }
    ]
  },
  {
    "name": "observaciones",
    "idoffset": 2,
    "x1": 60,
```

```
      "y1": 230,
      "x2": 720,
      "y2": 260,
      "extractionrules": [
        {
          "action": "QuitSpaces"
        },
        {
          "action": "Erase",
          "target": "OBSERVACIONES:"
        }
      ]
    }
  ],
  "renames": [
    {
      "name": "hora_iniciofin.INDEX[0]",
      "rename": "hora_inicio",
      "exact": true,
      "casesensitive": false
    },
    {
      "name": "hora_iniciofin.INDEX[1]",
      "rename": "hora_fin",
      "exact": true,
      "casesensitive": false
    },
    {
      "name": "swap.SWAP[",
      "rename": "pases",
      "exact": false,
      "casesensitive": false
    }
  ]
}
```