PdfLogicalExtractor

Author	Ángel Ibáñez Hernández
Version	1.0.1
Date	May 20, 2024

www.angelves.com

1 – General Descripcion

PdfLogicExtractor is a piece of software designed to extract information from PDF documents in a logical and orderly manner, so that can later be processed by the systems that integrates with.

The system is based on adaptable logic implemented in a template system that can process all documents of a certain type.

A template is capable of adapting to variations established in its definition, such as months with different numbers of days, displaced document areas, or differences between pages within the same document. And in general, any type of extraction logic needed.

Template extraction logic can perform result cleaning based on predefined rules, obtaining pure data types or removing parts of insignificant text.

Template extraction logic can perform calculations based on results obtained from different extractions or predefined values, such as calculating totals in a table based on price per unit.

Likewise, all functionalities or exceptions that the template logic of a document type requires to be a more effective tool can be specifically programmed.

2 – System Requirements

This software has been developed using Microsoft .NET Standard 2.1, ensuring long-term compatibility and interoperability across various Microsoft platforms. This option provides the highest level of integration.

The system requires external software to handle calls for sending document and template for be processed, and receiving data either through direct integration or with standard JSON format, which is the format used by modern APIs for interacting with external systems.

Due to the nature of the software, capable of integrating with APIs, this tool can also be used by almost all modern platforms on the market, as APIs establish a layer of isolation and security that does not require integration.

An API using PdfLogicExtractor as an internal DLL can be installed on a single machine, within a local IIS, on a remote server, or integrated into your intranet, or you can published in Cloud compatible with .NET (Azure, AWS, etc.).

3 – Integration

The system is contained in a dynamic link library DLL, included in a nuget package, which can be incorporated into any type of platform or software project in the .NET universe.

Direct integration into a .NET project can be done from the Visual Studio nuget manager in the project itself.

Integration defines a very simple software interface with a few overloaded calls and a response data model accessible directly, which can be easily processed as JSON responses.

Integration via API requires an API compatible with the .NET architecture where the dll/nuget can be integrated, which would be used internally as described in the previous section, returning standard JSON responses that can be integrated with almost all platforms on the market, whether written in .NET or not (Java Android, PHP, C++, etc.).

You can integrate the software using the VS nuget manager searching for:

Angelves.PdfLogicalExtractor

Or using the console with the next command:

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

You can find more documentation in:

www.angelves.com/PdfLogicalExtractor

You can try the software integrated at our API in:

www.angelves.com/PdfLogicalExtractor/TestOnlineFree

4 – Examples

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);
    }
}
```

Template example:

```
{
  "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
    }
  ]
}
```