

RENware Software Systems

xl2roefact

Cuprins:

- [xl2roefact](#)
 - [Facilitati](#)
 - [Instalarea](#)
 - [Instalarea aplicatiei xl2roefact](#)
 - [Instalarea bibliotecii Python xl2roefact PyPi](#)
 - [Configurarea aplicatiei xl2roefact](#)
 - [Comenzile aplicatiei](#)
 - [about](#)
 - [settings](#)
 - [xl2json](#)
 - [Practici si reguli referitoare la continutul facturilor din Excel](#)
 - [Tutorial utilizare aplicatie](#)
 - [Organizarea informatiei](#)
 - [Exemplu de procesare a unei facturi](#)
 - [Documentatia tehnica](#)
 - [Formatul fisierului JSON](#)
 - [Descarcare \(download\) aplicatie xl2roefact CLI](#)
 - [Arhiva versiuni publicate disponibile](#)
 - [Date identificare](#)
 - [License](#)
 - [Note](#)

versiune MSI installer 0.1.19

versiune script Python 0.1.20

versiune pachet Python PyPi 0.1.20

Facilitati

Aceasta componenta este "totul despre crearea de facturi electronice" din formatul Excel office (xlsx). Aplicatia poate genera factura in format JSON, XML, PDF si o poate incarca in sistemul *RO E-Fact*¹.

Aceasta componenta ofera urmatoarele facilitati (acestea fiind obiectivele fundamentale ale componentei):

- **transformarea facturilor din Excel in formatul XML** cerut de catre sistemul ANAF RO E-Fact pentru incarcare
- **incarcarea acestora** in sistemul ANAF RO E-Fact¹
- **transformarea facturilor din Excel intr-un format JSON** intermediar, independent de platforma si care permite integrarea acestora cu alte sisteme (standard *REST*)
- **generarea facturii in format PDF** pentru transmiterea acesteia catre client, semnarea electronica, tiparirea si arhivarea acesteia in format fizic (in general manipularea facturii in format "*human readable*")

Componenta ofera doua instrumente pentru realizarea si indeplinirea acestor obiective:

- `xl2roefact` o **aplicatie de tip linie de comanda** (disponibila pentru sistemele de operare Windows, Linux si MacOS)
- `xl2roefact` PyPi o **bioteca standard Python** utilizabila pentru dezvoltari proprii in scopul extinderii altor sisteme existente (*custom development*)

Instalarea

Instalarea este diferita pentru cele doua componente:

- aplicatia de tip *linie de comanda / consola*
- biblioteca "*standard Python package wheel*"

Instalarea aplicatiei xl2roefact

Pachetele de instalare se gasesc in directorul `dist/` ca arhive `ZIP`. Pachetele disponibile contin in numele lor versiunea de aplicatie utilizata si sistemul de operare pentru care sunt disponibile:

- `MSI` pachet instalare pentru *Windows
- `DEB` pachet instalare pentru *Linux Debian* (verificati disponibilitatea pentru varianta sistemului de operare folosit de dvs)
- `EXE` executabil *Windows in format "portabil" (un singur fisier)*
- **NOTA:** pentru echivalent utilizare *portabila pentru Linux* se va instala biblioteca Python (vezi sectiunea urmatoare) duoa care devine utilizabil scriptul Python "ca orice alta comanada Linux"

Instalarea bliotecii Python xl2roefact PyPi

Instalarea acesteia se face cu instrumentele standard Python. Recomandarea este pentru instalarea simpla cu: `pip install xl2roefact`, biblioteca fiind disponibila in repository-ul standard *PyPy*. Pentru instalarea din surse, biblioteca poate fi descarcata din [GitHub](#).

Configurarea aplicatiei xl2roefact

Parametrii de configurare a plicatiei se gasesc in fisierul `config_settings.py`. Acestia sunt sub elaborati in limbaj Python prin utilizarea conventiilor de constante conform recomandarilor PEP (numele capitatlizat) si sunt acompaniti de linii de explicatii privind aplicabilitatea lor.

Configurare aplicatiei se poate face interactiv si din aplicatie. Pentru a obtine help referitor la detaliile comenzi se va folosi

```
xl2roefact settings --help
```

Comenzile aplicatiei

Interfata aplicatie este realizata utilizind conventiile si practicile uzuale pentru aplicatii tip linie de comanda consola. Pentru informatii privind comenzile se poate folosi optiunea de **help**, dispobilia atit la nivelul general:

```
xl2roefact --help
```

cit si la nivel detaliat pentru fiecare comanda

```
xl2roefact [COMMAND] --help
```

about

Afiseaza informatii despre aceasta aplicatie (copyright, scop, etc).

settings

Afiseaza parametrii de configurare a aplicatiei. [Vezi sectiunea de configurare a aplicatiei.](#)

xl2json

Transforma fisierul (fisierele) Excel in forma JSON pentru utilizare ulterioara ca forma de date standardizat pentru schimbul de informatii cu alte sisteme electronice. Formatul JSON utilizat contine:

- informatiile aferente facturii (cheie: `Invoice`)
- o harta de ajutor in conversia formatului JSON in formatul XML acceptat de sistemul RO E-Fact (cheie `meta_info.map_JSONkeys_XMLtags`) si definitiile XML aferente (cheie `meta_info.invoice_XML_schemes`)
- alte informatii despre fisierul Excel prelucrat (alte chei din `meta_info`)
- datele preluate din formatul original Excel (cheie `excel_original_data`) - acestea sunt utile pentru depanare in caz ca aceasta este necesara in cazul specific al fisierului Excel folosit de dvs

Practici si reguli referitoare la continutul facturilor din Excel

Acest capitol se refera la modul in care este "tratata" continutul fisierului Excel cu factura, mai exact la modalitatea in care informatia facturii este cautata, identificata si gasita in scopul de a fi salvata in oricare din formatele de "factura electronica / E-Fact".

Utilizarea sablonului de factura Excel ce este livrat impreuna cu aplicatia **ESTE O VARIANTA DE LUCRU RECOMANDATA**, dar nu obligatorie. Chiar si in cazul utilizarii acestuia, prin modificarea "structurii" acestuia, informatia poate ajunge *nerecognoscibila / neidentificabila* total sau partial daca nu sunt urmate regulile expuse.

In general trebuie facuta diferenta intre datele facturii si modul in care aceasta va fi tiparita (va apare la tiparire / previzualizare).

Mai exact **continutul informational** al facturii nu trebuie nici confundat si nici mixat cu **formatul de afisare al acesteia** (layout). Pentru acesta din urma se recomanda a fi folosite cu precadere *regulile de formatare* din Excel si nu cele stocare a datelor. Un exemplu este un numar zecimal oarecare unde:

- una este valoarea introdusa intr-o celula (de ex cu 3 zecimale) si
- alta este valoarea afisata (cu 2 zecimale) - aceasta din urma trebuie obtinuta prin formatarea celulei respective de a afisa 2 zecimale prin rotunjire insa valoarea efectiva trebuie sa fie cea originala cu 3 zecimale, lucru (diferenta) care se poate vedea la editarea continutului celulei.

Tutorial utilizare aplicatie

Organizarea informatiei

Aplicatia `xl2roefact` "promoveaza" structurarea informatiei procesate astfel incit sa fie evitata situatia "*de aglomerare*" a directorului curent cu fisiere ce trebuiesc identificate si izolate in situatia in care se fac *procesari in masa* (pe mai multe fisiere / facturi sursa).

Astfel, aplicatia se asteapta ca fisierele Excel sursa (*adica facturile de procesat*) sa fie copiate in directorul `invoice_files/` de unde vor fi citite si tot aici vor fi create fisierele rezultate (JSON, XML, etc). Acest director este relativ la directorul curent de unde este lansata aplicatia si considerat "*implicit*" cu acest nume dar daca se doreste un alt director acest lucru poate fi facut folosind parametrul `--files-directory` (sau prescurtat `-d`) la lansarea aplicatiei astfel:

```
xl2roefact -d "calea si numele directorului dorit"
```



Nota

Ghilimelele sunt necesare numai daca numele si calea (path) contin caracterul spatiu.

Exemple:

- pentru stabilirea directorului curent ca sursa pentru fisierele factura Excel: `xl2roefact -d ./`
- procesarea tuturor facturilor din luna *iunie*, copiate intr-un director dedicat sub directorul curent:
`xl2roefact -d ./facturi_iunie/`

Exemplu de procesare a unei facturi

- se creaza directorul `invoice_files`
- se copiaza factura `factura_A.xlsx` in acest director apoi se revine in directorul anterior daca acesta a fost schimbat pentru efectuarea copierii
- se lanseaza aplicatia: `xl2roefact xl2json`

In urma acestor operatii, in directorul `invoice_files` vor rezulta:

```
invoice_files/  
├─ factura_A.xlsx # fisierul Excel original  
└─ factura_A.json # fisierul JSON rezultat in urma procesarii
```

- `factura_A.xlsx` ca fiind fisierul Excel original cu factura
- `factura_A.json` acesta fiind fisierul format JSON rezultat in urma procesarii si ce poate fi folosit pentru interschimbarea electronica a informatiei intre sisteme

Documentatia tehnica

Documentatia tehnica de design si specificatiile detaliate pot fi accesate [aici](#)

Formatul fisierului JSON

Structura de baza a fisierului JSON aferent unei facturi este:

```
{
  "Invoice": {...},
  "meta_info": {...},
  "excel_original_data": {...}
}
```

Cheile de la primul nivel reprezinta:

- **Invoice** - #TODO tbd...
- **meta_info** - #TODO tbd...
- **excel_original_data** - #TODO tbd...

Descarcare (download) aplicatie xl2roefact CLI

- 0.1.20.dev invoice customer address [MSI installer win64](#)

Arhiva versiuni publicate disponibile

- 0.1.19.dev invoice customer and partial invoice total values calculations [MSI installer win64](#)
- 0.1.18.dev invoice customer CUI partial invoice total values calculations [MSI installer win64](#)

NOTA: Pentru descarcarea bibliotecii, pachetului Python `xl2roefact` [procesul este descris aici](#)

Date identificare

- part number (p/n): `0000-0095-xl2roefact`
- producator si copyright: RENWare Software Systems
- author: Petre Iordanescu (petre.iordanescu@gmail.com)

License

Note

1. Toate interactiunile cu sistemul *ANAF RO E-Fact* necesita o *conexiune la internet* si un set de *credentiale ANAF RO E-Fact ale companiei* pentru care se incarca factura. In lipsa acestora, fisierul `XML` generat de aplicatie poate fi incarcat ulterior (de ex de catre departmentul contabilitate) [↩↩](#)
-

Last update: January 30, 2024