

	<b>IRIDE Analytics</b> Il primo sistema integrato per l'analisi di dati in farmacia	IRIDE ANALYTICS MANUAL	
	<b>Invio giornaliero scontrini</b>	Revisione 02	Data inizio <b>2020          25 giugno</b>

# Specifiche Dialoghi IRIDE Analytics

Rev.	Motivo ultima revisione	Redatto	Verificato	Approvato	Data
01	Prima emissione	A. Lucchini			2021/06/25
02	Modifica formato data e endpoint	A. Lucchini			2021/07/01
			Rapp. Direzione	Rapp. Direzione	
			<i>Firma</i>	<i>Firma</i>	

## INDICE

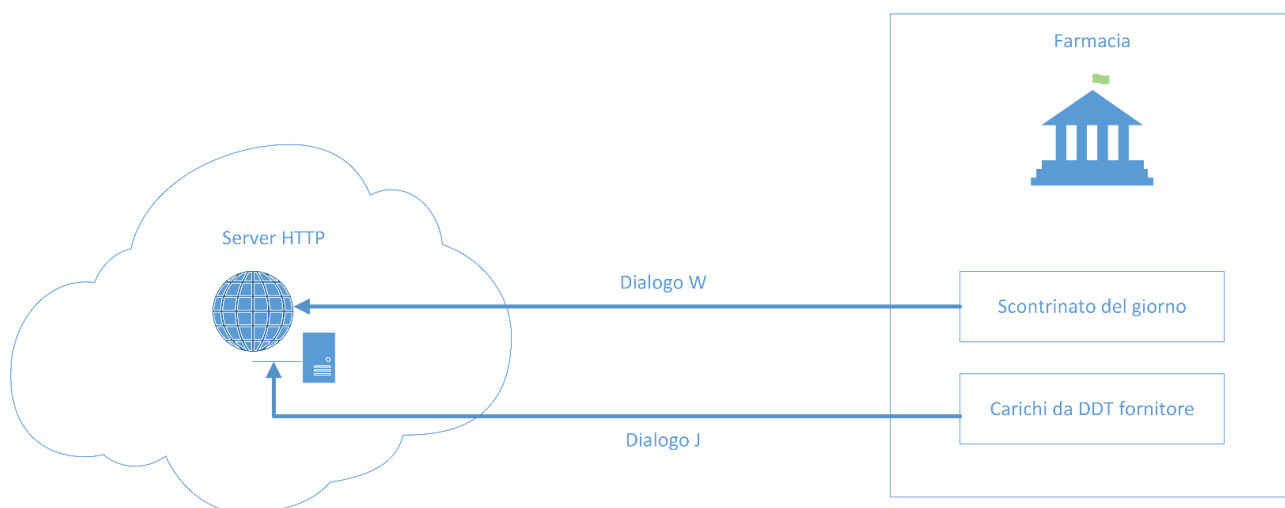
<b><u>I</u></b>	<b><u>STRUTTURA GENERALE DIALOGHI.....</u></b>	<b><u>3</u></b>
<b><u>1</u></b>	<b><u>DIALOGO W.....</u></b>	<b><u>4</u></b>
1.1	STEP 1 – LOGIN .....	4
1.2	STEP 2 – SALVATAGGIO DEL VENDUTO .....	5
1.3	ESEMPI JSON RISPOSTA .....	5
1.4	JSON DATI.....	6
1.4.1	SPECIFICHE STANDARD .....	6
1.4.2	DEFINIZIONE STRUTTURA .....	7
1.4.3	NOTE AGGIUNTIVE.....	7
1.4.4	ESEMPIO CODICE JSON VALIDATO - ESEMPIO VALIDATO RFC-8259 .....	8
<b><u>2</u></b>	<b><u>DIALOGO J.....</u></b>	<b><u>10</u></b>
2.1	STEP 1 – LOGIN .....	10
2.2	STEP 2 - SALVATAGGIO DEL DEPOSITO .....	11
2.3	ESEMPI JSON RISPOSTA .....	11
2.4	JSON DATI.....	12
2.4.1	DEFINIZIONE STRUTTURA.....	12
2.4.2	NOTE AGGIUNTIVE.....	12
2.4.3	ESEMPIO CODICE JSON VALIDO – ESEMPIO VALIDATO RFC-8259.....	13

## I STRUTTURA GENERALE DIALOGHI

Alcuni sistemi Iride Analytics richiedono l'acquisizione di dati relativi al venduto ed al caricato a magazzino giornalieri per poter operare. L'acquisizione di questi dati è prevista attraverso l'utilizzo di una chiamata POST ad un server posizionato nel cloud.

I dialoghi utilizzati per il trasferimento dati sono i seguenti:

- Dialogo W: per il trasferimento (tramite file JSON) dei dati di fatturato giornaliero
- Dialogo J: per il trasferimento (tramite file JSON) dei dati relativi al carico prodotti nel magazzino della farmacia.



## 1 DIALOGO W

Il dialogo W viene utilizzato per trasferire i dati relativi a tutti gli scontrini di vendita emessi dalla farmacia durante l'intera giornata.

Si propone di ricevere i dati all'interno di chiamate POST verso il box contenenti un tracciato JSON.

Ogni messaggio prevede un suo URL univoco (**endpoint**), e una risposta, sempre in formato JSON, sia in caso di successo che in caso di errore.

È necessario che la chiamata sia autenticata, ottenendo un token di autenticazione, da utilizzare nelle chiamate successive.

Negli esempi l'indirizzo IP dell'endpoint è identificato come **IP\_server**

### 1.1 STEP 1 – LOGIN

Si effettua una POST a `/api/v1/login` con parametri `username` e `password` (URL encoded), esempio con cURL:

```
curl \
  --data-urlencode "username=User" \
  --data-urlencode "password>Password" \
  -X POST \
  'http://IP_server /api/v1/login' -i
```

Il server risponde con content type `application/json` contenente un JSON ed imposta nel client il cookie di sessione e quello per CSRF:

```
HTTP/1.0 200 OK
Date: Tue, 02 Oct 2018 16:51:44 GMT
Vary: Cookie
X-Frame-Options: SAMEORIGIN
Content-Type: application/json
Content-Language: en
Set-Cookie: csrftoken=vL8byz5JxgdM7qqUoNgAs86HKyOQ8XiC; expires=Tue, 01-Oct-2019 16:51:44 GMT; Max-Age=31449600; Path=/
Set-Cookie: sessionid=hr5cjzobv6y0vgn3tn7bng34vhljccug; expires=Tue, 16-Oct-2018 16:51:44 GMT; httponly; Max-Age=1209600; Path=/

{"csrftoken": "vL8byz5JxgdM7qqUoNgAs86HKyOQ8XiC", "message": "Logged in",
"result": "OK"}
```

Il CSRF token è riportato per comodità anche nel JSON di risposta, benché di fatto inutile perché già impostato come cookie.

## 1.2 STEP 2 – SALVATAGGIO DEL VENDUTO

Si effettua una nuova POST a `/api/v1/venduto`, con inserito nel corpo della post i dati del messaggio, usando il cookie di sessione ed impostando l'header X-CSRFToken. Esempio con cURL:

```
CSRF_TOKEN=iZwVZXsO3aS6TIn9fDI0lpNF4ogCNgfI  
SESSION_ID=16sb0owhph9cqjn9thjfj36ccyvnl62
```

```
curl \  
  -X POST \  
  -H "X-CSRFToken: ${CSRF_TOKEN}" \  
  -b "csrftoken=${CSRF_TOKEN}; sessionid=${SESSION_ID}" \  
  
  --data "{ <QUI CI VA IL MESSAGGIO, VEDI ESEMPIO>}" \  
  'http://IP_server/api/v1/venduto' -i
```

Il server risponde con un JSON contenente un messaggio di OK o errore.

## 1.3 ESEMPI JSON RISPOSTA

Esempio di risposte:

```
{  
  
  "result": "OK",  
  
  "msg": "xDati registrati"  
}
```

oppure

```
{  
  
  "result": "KO",  
  
  "msg": "Errore durante il salvataggio"  
}
```

## 1.4 JSON DATI

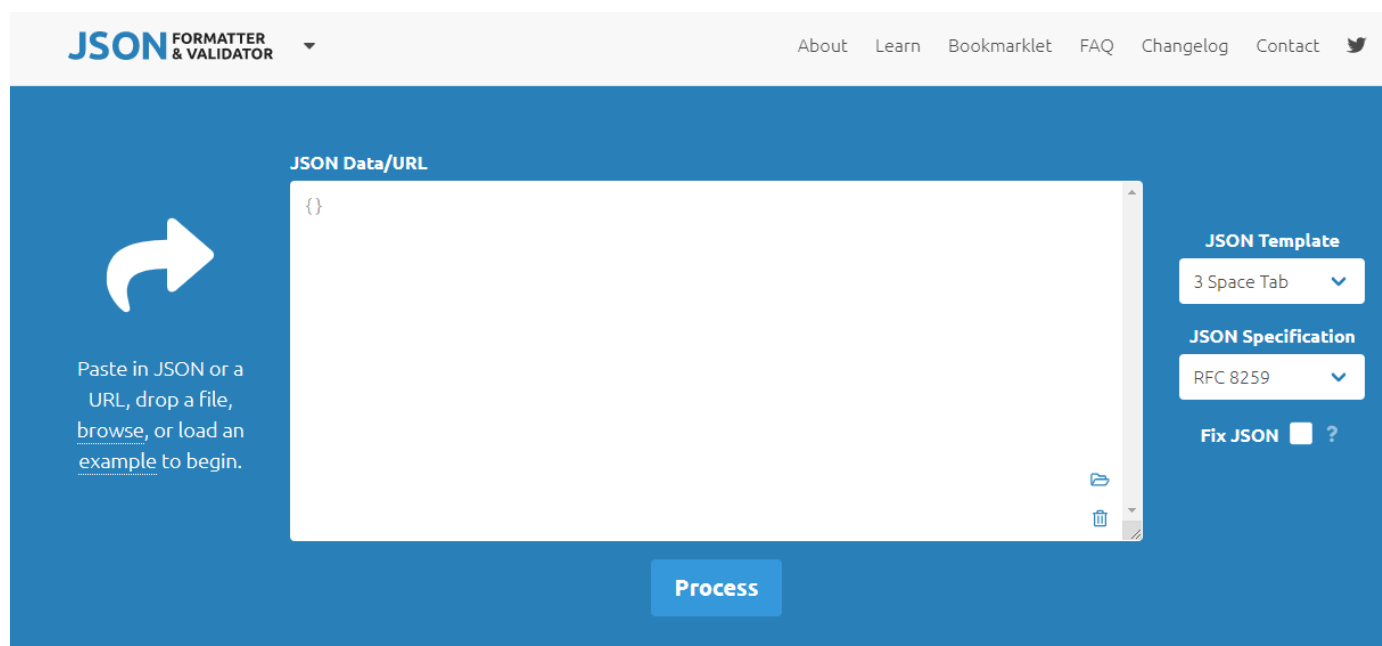
### 1.4.1 Specifiche standard

La comunicazione del venduto viene effettuata trasferendo un unico file JSON per ogni singola giornata di lavoro. Eventuali file non inviati nella giornata (ad es. per problemi di connettività) potranno essere inviati nei giorni successivi. Lo standard utilizzato per la codifica del file JSON è il RFC8259.

Per la verifica della validità del proprio JSON in output consigliamo un check online qui:

<https://jsonformatter.curiousconcept.com/#>

impostando lo standard RFC8259.



The screenshot shows the interface of the 'JSON FORMATTER & VALIDATOR' website. The header includes the site name and navigation links: 'About', 'Learn', 'Bookmarklet', 'FAQ', 'Changelog', and 'Contact'. The main content area has a blue background with a white arrow icon pointing right, accompanied by the text: 'Paste in JSON or a URL, drop a file, browse, or load an example to begin.' To the right of this text is a large white text area labeled 'JSON Data/URL' containing the JSON characters '{}'. Below this text area is a blue 'Process' button. On the right side of the interface, there are two dropdown menus: 'JSON Template' set to '3 Space Tab' and 'JSON Specification' set to 'RFC 8259'. Below these is a 'Fix JSON' checkbox which is currently unchecked, followed by a question mark icon.

## 1.4.2 Definizione struttura

La struttura del file JSON ed i relativi campi da compilare, è la seguente:

```

{codice_farmacia:      xxxxxxxxxxxxxxxxxxxx,
data:                  yyyy-mm-dd,
codice_gestionale:    zzzzzzzzzzzzzzzzzzzz,
versione_file:        wwwwwwwwwwwwwwwww,
vendite:               [{numero_scontrino:   numero_progressivo,
                        data_ora:           yyyy-mm-dd HH:MM:SS,
                        sconto_%.          xx.yy,
                        sconto_value:      vvvvvv,
                        dettaglio_vendita:  [{tipo_movimento:           "V"=vendita; "A"=annullo,
                                           codice_minsan:             dddddddddd,
                                           codice_EAN:                eeeeeeeeeeee,
                                           descrizione_prodotto:      cvcvcvcvcvcvcvcvcv,
                                           descrizione_casa_prod:      fdfdfdfdfdfdfdfdfd,
                                           codice_ATC-GMP:            fgfgfggg,
                                           aliquota_IVA:              gg,
                                           quantita:                  YYYYYY,
                                           prezzo_unitario_anagrafico: kkkk,kk ,
                                           prezzo_unitario_netto:      kkkk,kk
                                           }
                                           ]
                        }
                    ]
}
    
```

## 1.4.3 Note aggiuntive

Il formato per trasferire i valori di data è il seguente: "yyyy-mm-dd".

Il formato per trasferire i valori di data e ora è il seguente: "yyyy-mm-dd HH:MM:SS".

Tutti i numeri decimali vanno trasferito utilizzando il carattere "." come posizionatore decimale, non la virgola.

## 1.4.4 Esempio codice Json validato - Esempio validato RFC-8259

```
{
  "codice_farmacia": "PV-00616",
  "data": "2021-06-09",
  "codice_gestionale": "Wingestfar",
  "versione_file": "0.0.01",
  "vendite": [
    {
      "numero_scontrino": 1,
      "data_ora": "2021-06-09 10:43:00",
      "sconto_%": 0.0,
      "sconto_value": 0.0,
      "dettaglio_vendita": [
        {
          "tipo_movimento": "V",
          "codice_minsan": "004763330",
          "codice_EAN": "",
          "descrizione_prodotto": "Aspirina C - 20cpr Eff 400+240mg",
          "descrizione_casa_prod": "Bayer Spa",
          "codice_ATC-GMP": "N02BA51",
          "aliquota_IVA": 10,
          "quantita": 1,
          "prezzo_unitario_anagrafico": 6.9,
          "prezzo_unitario_netto": 6.9
        },
        {
          "tipo_movimento": "V",
          "codice_minsan": "026089019",
          "descrizione_prodotto": "Augmentin - 12cpr Riv875mg+125mg",
          "codice_EAN": "",
          "descrizione_casa_prod": "Glaxosmithkline Spa",
          "codice_ATC-GMP": "J01CR02",
          "aliquota_IVA": 10,
          "quantita": 1,
          "prezzo_unitario_anagrafico": 10.2,
          "prezzo_unitario_netto": 10.2
        },
        {
          "tipo_movimento": "V",
          "codice_minsan": "025940053",
          "codice_EAN": "",
          "descrizione_prodotto": "AULIN*OS GRAT 30BUST 100MG",
          "descrizione_casa_prod": "Helsinn Birex Pharmac.Ltd",
          "codice_ATC-GMP": "M01AX17",
          "aliquota_IVA": 10,
          "quantita": 3,
          "prezzo_unitario_anagrafico": 4.43,
          "prezzo_unitario_netto": 4.43
        }
      ]
    }
  ]
}
```



```

{"numero_scontrino":2,
"data_ora":"2021-06-09 11:13:00",
"sconto_%":0.0,
"sconto_value":0.0,
"dettaglio_vendita": [
  {"tipo_movimento":"V",
  "codice_minsan":"004763330",
  "codice_EAN": "",
  "descrizione_prodotto":"Aspirina C - 20cpr Eff 400+240mg",
  "descrizione_casa_prod":"Bayer Spa",
  "codice_ATC-GMP":"N02BA51",
  "aliquota_IVA":10,
  "quantita":3,
  "prezzo_unitario_anagrafico": 6.9 ,
  "prezzo_unitario_netto": 6.9
  },
  {"tipo_movimento":"V",
  "codice_minsan":"026089019",
  "descrizione_prodotto":"Augmentin - 12cpr Riv875mg+125mg",
  "codice_EAN": "",
  "descrizione_casa_prod":"Glaxosmithkline Spa",
  "codice_ATC-GMP":"J01CR02",
  "aliquota_IVA":10,
  "quantita":2,
  "prezzo_unitario_anagrafico":10.2,
  "prezzo_unitario_netto":10.2
  }
]
}
]
}
    
```

## 2 DIALOGO J

Il dialogo J viene utilizzato per trasferire i dati relativi a tutti i prodotti caricati tramite specifico DDT in farmacia. In base alla convenienza del sistema gestionale è possibile inviare un dialogo J ad ogni singolo DDT caricato in farmacia oppure fare un invio unico che riepiloghi tutti i prodotti caricati in farmacia durante la giornata.

Si propone di ricevere i dati all'interno di chiamate POST verso il box contenenti un tracciato JSON.

Ogni messaggio prevede un suo URL univoco (**endpoint**), e una risposta, sempre in formato JSON, sia in caso di successo che in caso di errore.

E' necessario che la chiamata sia autenticata, ottenendo un token di autenticazione, da utilizzare nelle chiamate successive.

Negli esempi l'indirizzo IP dell'endpoint è identificato come **IP\_server**, le credenziali di login al sistema server remoto sono invece definite come **User** e **Password**.

### 2.1 STEP 1 – LOGIN

Si effettua una POST a `/api/v1/login` con parametri `username` e `password` (URL encoded), esempio con cURL:

```
curl \
  --data-urlencode "username=User" \
  --data-urlencode "password=Password" \
  -X POST \
  'http://IP\_server /api/v1/login' -i
```

Il server risponde con content type `application/json` contenente un JSON ed imposta nel client il cookie di sessione e quello per CSRF:

```
HTTP/1.0 200 OK
Date: Tue, 02 Oct 2018 16:51:44 GMT
Vary: Cookie
X-Frame-Options: SAMEORIGIN
Content-Type: application/json
Content-Language: en
Set-Cookie: csrftoken=vL8byz5JxgdM7qqUoNgAs86HKyOQ8XiC; expires=Tue, 01-Oct-2019 16:51:44 GMT; Max-Age=31449600; Path=/
Set-Cookie: sessionid=hr5cjzobv6y0vgn3tn7bng34vhljccug; expires=Tue, 16-Oct-2018 16:51:44 GMT; httponly; Max-Age=1209600; Path=/

{"csrftoken": "vL8byz5JxgdM7qqUoNgAs86HKyOQ8XiC", "message": "Logged in",
"result": "OK"}
```

Il CSRF token è riportato per comodità anche nel JSON di risposta, benché di fatto inutile perché già impostato come cookie.

## 2.2 STEP 2 - SALVATAGGIO DEL DEPOSITO

Si effettua una nuova POST a `/api/v1/depositato`, con inserito nel corpo della post i dati del messaggio, usando il cookie di sessione ed impostando l'header X-CSRFToken. Esempio con cURL:

```
CSRF_TOKEN=iZwVZXsO3aS6TIn9fDI0lpNF4ogCNgfI  
SESSION_ID=l6sb0owhph9cqjnj9thjfj36ccyvnl62
```

```
curl \  
  -X POST \  
  -H "X-CSRFToken: ${CSRF_TOKEN}" \  
  -b "csrftoken=${CSRF_TOKEN}; sessionid=${SESSION_ID}" \  
  
  --data "{ <QUI CI VA IL MESSAGGIO, VEDI ESEMPIO>" \  
  'http://IP\_server /api/v1/deposito' -i
```

Il server risponde con un JSON contenente un messaggio di OK o errore.

## 2.3 ESEMPI JSON RISPOSTA

Esempio di risposte:

```
{  
  
  "result": "OK",  
  
  "msg": "xDati registrati"  
}
```

oppure

```
{  
  
  "result": "KO",  
  
  "msg": "Errore durante il salvataggio"  
}
```

## 2.4 JSON DATI

### 2.4.1 DEFINIZIONE STRUTTURA

La struttura del file JSON ed i relativi campi da compilare, è la seguente:

```
{codice_farmacia:      xxxxxxxxxxxxxxxxxxxx,  
  
data:                  yyyy-mm-dd,  
  
codice_gestionale:    zzzzzzzzzzzzzzzzzzzz,  
  
versione_file:        wwwwwwwwwwwwwwwww,  
  
carico:                [{codice_minsan:      XXXXXXXXX,      (intero a 9 cifre)  
  
                        descrizione:        sssssssssssssssssssss, (stringa di lunghezza variabile)  
  
                        quantita:          YYYYYYY  
  
                        }  
  
                ]  
  
}
```

### 2.4.2 NOTE AGGIUNTIVE

Il formato per trasferire i valori di data è il seguente: "yyyy-mm-dd".

## 2.4.3 ESEMPIO CODICE JSON VALIDO – ESEMPIO VALIDATO RFC-8259

```
{
  "codice_farmacia": "PV-00616",
  "data": "2021-06-09",
  "codice_gestionale": "Wingestfar",
  "versione_file": "0.0.01",
  "carico": [
    {
      "codice_minsan": "801117235",
      "descrizione": "KINDISTIP 800GL",
      "quantita": 9
    },
    {
      "codice_minsan": "905954982",
      "descrizione": "RESTIVOIL OLIO SHAMPOO FISILOG",
      "quantita": 12
    }
  ]
}
```