ENTAKSISOLUTIONS

SISTEMA DI GESTIONE CERTIFICATO
ISO 9001 | ISO 20000-1 | ISO 22301 | ISO 37001
ISO 27001 | ISO 27017 | ISO 27018 | ISO 27035
SERVIZI FIDUCIARI QUALIFICATI
ETSI 319 401 | ETSI 319 411-1 e 2 | ETSI 319 421 | ETSI 119 511
FIRME E SIGILLI ELETTRONICI - MARCHE TEMPORALI
CONSERVAZIONE A LUNGO TERMINE

eDoc API

Manuale utente

UM 20150928 Manuale utente eDoc API
Version 1.33.6, 17/04/2025

Indice

Informazioni sul documento . . . 1
Revisioni . . . 1
Approvazione del documento . . . 2
1. Introduzione . . . 3
2. Autenticazione delle chiamate alle API . . . 4
2.1. Autenticazione per Applicazioni Web Utilizzando l'Authorization Code Flow . . . 4
2.2. Autenticazione Out-of-Band Utilizzando l'Authorization Code Flow . . . 5
2.3. Autenticazione per Login su Dispositivi Utilizzando il Device Authorization Grant . . . 6
2.4. Documentazione delle API e ambiente di interazione on line . . . 9
2.5. Ambiente di test . . . 9
3. Modello generale del sistema . . . 10
4. Attivazione di una azienda . . . 11
4.1. Associazione di un `client_id` al rivenditore . . . 11
5. Gestione delle fatture XML . . . 12
5.1. Trasmissione della Fattura XML . . . 12
5.2. Monitoraggio delle fatture trasmesse . . . 12
5.3. Ricezione delle fatture . . . 13
5.4. Trasmissione della notifica Esito Committente . . . 14
5.5. Fatturazione elettronica tra Italia e San Marino . . . 14
5.6. Fatture passive di operatori economici di San Marino e cedente Italiano . . . 14
6. Gestione delle strutture dell'azienda . . . 16
6.1. Caricamento diretto della Fattura XML . . . 16
6.2. Diritti di accesso degli utenti . . . 16
7. Pacchetti di versamento . . . 17
7.1. Stati del pacchetto di versamento . . . 17
7.2. Formato `F000` Fatture elettroniche inviate . . . 17
7.3. Formato `F001` Fattura XML gestite dal sistema . . . 18
7.4. Formato `F002` Fatture elettroniche ricevute . . . 19
7.5. Formato `F003` Fatture elettroniche ricevute gestite dal sistema . . . 19
7.6. Formato `F005` PEC eml . . . 19
7.7. Formato `F997` Pacchetto copia digitale di documento analogico . . . 20
7.8. Formato `F998` Pacchetto di precedente conservatore . . . 20
7.9. Formato `F999` Pacchetto generico con indice . . . 20
8. Esempi di Pacchetti di Versamento . . . 21
8.1. Tipi documento . . . 21
8.2. Esempio pacchetto con documenti `D01` Fatture emesse e `D02` Fatture ricevute . . . 25
8.3. Esempio pacchetto con documenti `D15` Registro fatture acquisto . . . 29
8.4. Esempio pacchetto `D17` Registro fatture emesse . . . 32
8.5. Esempio pacchetto `D26` Registro riepilogativo Iva . . . 34
8.6. Esempio pacchetto `D58` Altri registri . . . 38
8.7. Esempio pacchetto `D85` Altri documenti . . . 40
8.8. Esempio pacchetto `D58` Cedolini . . . 42
8.9. Esempio pacchetto `D58` Cartellino presenze . . . 44
8.10. Esempio pacchetto `D58` Elenco dipendenti . . . 46

8.11. Esempio pacchetto D58 Riepilogo LUL . . . 48
8.12. Esempio pacchetto D58 Indice LUL . . . 49
8.13. Esempio pacchetto D58 Riepilogo documenti firmati . . . 51
8.14. Esempio pacchetto D8503 Manuali . . . 53
9. Pacchetti di distribuzione . . . 55
9.1. Stati della richiesta di PDD . . . 55
10. Codice di errore . . . 56
11. Codici . . . 58
11.1. Stati della fattura . . . 58
11.2. Esito fattura . . . 59
11.3. Tabella dei codici di validazione dei PDV . . . 60
11.4. Tabella dei codici degli stati del PDV . . . 61
11.5. Tabella dei codici delle azioni sui PDV . . . 63
11.6. Tabella dei codici degli stati del PDA . . . 63
11.7. Tabella dei codici delle azioni sui PDA . . . 64
11.8. Tabella dei codici degli stati di una richiesta di PDD . . . 64
11.9. Tabella dei codici delle azioni sulle richieste di PDD . . . 65
11.10. Tabella dei codici delle operazioni tracciate nell'audit log . . . 66

Informazioni sul documento

Progetto	eDoc API
ID documento	UM 20150928 Manuale utente eDoc API
Tipo	Manuale utente
Data creazione	28/09/2015
Ultima revisione	17/04/2025
Versione	1.33.6
Autore	Stefano Travelli
Stato	Rilasciato
Classificazione	Confidenziale

Riproduzioni cartacee di questo documento sono da considerarsi copie di lavoro non censite dal SIG.

Revisioni

Data	Versione	Nome	Azione
28/09/2015	0.0.1	Stefano Travelli	Creazione del documento.
15/10/2015	0.0.2	Stefano Travelli	Adeguamento del formato.
23/05/2016	1.0.16	Stefano Travelli	Documentazione API Archiviazione.
02/02/2017	1.6.7	Stefano Travelli	Aggiornamento per fattura B2B.
15/05/2017	1.7.2	Stefano Travelli	Aggiornamento per caricamento PDV.
27/06/2017	1.8.0	Alessia Soccio	Descrizione esempi PDV.
26/07/2017	1.9.0	Stefano Travelli	Aggiunte tabelle dei codici.
09/02/2018	1.10.0	Alessia Soccio	Revisione, aggiunta esempi PDV.
05/03/2018	1.10.1	Stefano Travelli	Aggiunti dettagli sul processo di gestione delle fatture.
05/03/2018	1.10.2	Alessia Soccio	Correzioni minori su esempi PDV.
19/04/2018	1.11.9	Alessia Soccio	Modifica esempio PDV "D85, Altri documenti", inserimento definizione di pacchetto F997.
06/06/2018	1.11.10	Stefano Travelli	Aggiunti alcuni esempi e chiarimenti sull'autenticazione.
26/11/2018	1.13.0	Luigi Ruocco	Documentazione codici.
11/12/2018	1.13.1	Alessia Soccio	Aggiustamenti su esempi PDV.
10/04/2020	1.18.0	Alessandro Ricchiuti	Aggiunti dettagli sulla richiesta di un token di tipo Offline Access.

Data	Versione	Nome	Azione
01/06/2021	1.20.5	Luigi Ruocco	Revisione codici di errore delle risposte delle API.
28/06/2022	1.22.10	Marco Martini	Fatturazione elettronica per operatori economici di San Marino.
17/04/2025	1.33.6	Riccardo Campisano	Aggiunta documentazione ed esempi per autenticazione in modalità Out of band e Device Flow.

Approvazione del documento

Data	Addetto	Mansione	Firma
17/04/2025	Stefano Travelli	DT	

© 2025 Entaksi Solutions SpA

Le informazioni contenute nel presente documento sono di proprietà di Entaksi Solutions, sono fornite ai destinatari in via riservata e confidenziale e non possono essere usate per fini produttivi, né comunicate a terzi o riprodotte, per intero o in parte, senza il consenso scritto di Entaksi Solutions.

1. Introduzione

eDoc API è una collezione di API REST che consente di accedere ai servizi del sistema di archiviazione elettronica Entaksi integrato con la gestione della fattura elettronica.

Le API consentono di operare su tutte le funzionalità per la trasmissione e la ricezione delle Fattura Elettronica, sia nella versione per la Pubblica Amministrazione (Fattura PA) che in quella per privati (Fattura B2B) e sulle funzionalità del sistema di conservazione, dal caricamento dei Pacchetti di Versamento, l'accesso e la ricerca nell'archivio dei documenti conservati e la produzione di Pacchetti di Distribuzione.

Il sistema di archiviazione Entaksi è accreditato presso l'Agenzia per l'Italia Digitale e rispetta quindi i requisiti per essere utilizzato dagli enti pubblici italiani, oltre che offrire lo stesso livello di garanzia a quelli privati.

La gestione delle fatture elettroniche comprende l'interfaccia con un canale di trasmissione e ricezione accreditato presso il Sistema di Interscambio, sistema con il quale le fatture possono essere inviate indirizzandole ad enti della Pubblica Amministrazione (tramite il codice destinatario pubblicato nell'Indice PA) oppure a privati (tramite codice destinatario, per i privati che ne dispongono, oppure tramite PEC).

Mediante il canale di ricezione, anch'esso accreditato presso il Sistema di Interscambio le aziende o gli enti della Pubblica Amministrazione possono ricevere le fatture elettroniche.

Per le aziende private registrate Entaksi assegna un codice destinatario allo scopo di ricevere le fatture elettroniche tramite il sistema anziché tramite PEC. In questo modo è anche possibile, per un'applicazione gestionale opportunamente predisposta, scaricare le fatture ricevute e inserirle automaticamente nella contabilità senza interagire con il protocollo di trasferimento definito dal Sistema di Interscambio, ma utilizzando le API REST e i meccanismi di interazione definiti da *eDoc API*.

Nella parte relativa all'archiviazione elettronica, le API consentono all'utente di eseguire le tre operazioni principali connesse al processo di conservazione in base alla normativa e allo standard OAIS:

- Trasmettere *Pacchetti di Versamento* contenenti documenti da mettere in archiviazione;
- Ricercare i documenti archiviati nei *Pacchetti di Archiviazione* generati a fronte del versamento;
- Ottenere una copia dei documenti archiviati mediante la formazione di un *Pacchetto di Distribuzione*.

Le API utilizzano uno schema di autenticazione basato sul protocollo *OAuth2* per identificare l'utente.

Le possibilità operative dell'utente dipendono poi dalle autorizzazioni derivanti dalle entità gestite dal sistema a cui viene associato.

Queste entità sono:

- il *Produttore* dei documenti, ovvero l'azienda o l'ente pubblico che utilizza il sistema;
- il *Contratto di servizio*, ovvero l'entità che definisce quali servizi possono essere utilizzati nel sistema;
- il *Rivenditore*, ovvero l'entità che, previo un accordo di rivendita con Entaksi Solutions, attiva contratti di erogazione dei servizi con i propri clienti.

Questo documento descrive il modello del servizio implementato dal sistema e illustra le modalità d'uso delle API per attivare, accedere e controllare i vari servizi e funzionalità del sistema.

Alcuni concetti relativi all'archiviazione a norma dei documenti sono descritti in maggiore dettaglio nel Manuale del sistema di archiviazione *MAN eCON 20151222 Conservazione*.

2. Autenticazione delle chiamate alle API

Le API REST del servizio sono documentate attraverso un descrittore Swagger che si trova stabilmente ai seguenti indirizzi:

- Per l'ambiente di test: https://test.entaksi.eu/api/edoc/v1/swagger.json
- Per l'ambiente di produzione: https://entaksi.eu/api/edoc/v1/swagger.json

In entrambi gli ambienti la documentazione è disponibile anche mediante un'interfaccia web:

- Per l'ambiente di test: https://test.entaksi.eu/api/edoc/v1-doc/
- Per l'ambiente di produzione: https://entaksi.eu/api/edoc/v1-doc/

Per invocare le API è necessario predisporre un meccanismo di autenticazione basato su OpenID Connect Core 1.0, che è un protocollo ampiamente diffuso e documentato per questo scopo.

L'autenticazione avviene impostando un header HTTP `Authorization` con il valore `Bearer <access_token>` dove `<access_token>` è il token di accesso ottenuto tramite il meccanismo di autenticazione.

Esistono vari meccanismi di autenticazione che consentono di ottenere il token per accedere alle API.

I tre principali sono descritti nei seguenti paragrafi:

- Autenticazione per Applicazioni Web Utilizzando l'Authorization Code Flow
- Autenticazione Out-of-Band Utilizzando l'Authorization Code Flow
- Autenticazione per Login su Dispositivi Utilizzando il Device Code Flow

2.1. Autenticazione per Applicazioni Web Utilizzando l'Authorization Code Flow

L'Authorization Code Flow è un metodo di autenticazione sicuro e ampiamente utilizzato per le applicazioni web che necessitano di accedere a risorse protette. Questo flusso garantisce che le credenziali dell'utente non vengano esposte direttamente all'applicazione client, migliorando così la sicurezza complessiva del processo di autenticazione.

Per configurare l'Authorization Code Flow, è necessario ottenere un `client_id`. Questo `client_id` sarà di tipo pubblico, ovvero senza un `client_secret`, in quanto la configurazione risulterà esposta nell'applicazione web che la utilizza.

Questo `client_id` è associato a uno o più `redirect_uri`, che rappresentano gli URL dove l'applicazione sarà pubblicata e dove l'utente verrà reindirizzato dopo l'autenticazione.

Il provider di identità espone un endpoint di discovery che fornisce tutte le informazioni necessarie per configurare il flusso di autenticazione.

L'endpoint di discovery è: https://entaksi.eu/auth/realms/entaksi/.well-known/openid-configuration

Da questo endpoint è possibile ottenere gli URL degli endpoint necessari per l'autorizzazione e il token, nonché altre configurazioni rilevanti.

Il flusso di autenticazione è solitamente implementato da librerie conformi allo standard OIDC, come "angular-oauth2-oidc", e segue questi passaggi:

- **Richiesta di autorizzazione**: l'applicazione reindirizza l'utente all'endpoint di autorizzazione del provider di identità, includendo il `client_id`, il `redirect_uri`, e altri parametri necessari.
- **Autenticazione dell'utente**: l'utente inserisce le proprie credenziali e autorizza l'applicazione ad accedere alle risorse richieste.
- **Ricezione del codice di autorizzazione**: dopo l'autenticazione, il provider di identità reindirizza l'utente al `redirect_uri` specificato, includendo un codice di autorizzazione nell'URL.
- **Richiesta del token**: l'applicazione utilizza il codice di autorizzazione per richiedere un token di accesso e un token ID dal provider di identità.
- **Accesso alle risorse**: con il token di accesso, l'applicazione può ora accedere alle risorse protette per conto dell'utente.

Per aumentare ulteriormente la sicurezza del flusso di autenticazione, è preferibile adottare il meccanismo PKCE (Proof Key for Code Exchange).

PKCE aggiunge un ulteriore livello di protezione contro attacchi di intercettazione del codice di autorizzazione, specialmente in

scenari in cui l'applicazione client non può utilizzare un `client_secret`, come quello delle applicazioni web.

Con PKCE, l'applicazione genera un `code_verifier` e un `code_challenge` che vengono utilizzati durante la richiesta di autorizzazione e la richiesta del token. Questo meccanismo assicura che solo l'applicazione client originale possa scambiare il codice di autorizzazione con un token di accesso, migliorando così la sicurezza complessiva del processo.

Questo flusso garantisce che le credenziali dell'utente rimangano sicure e che l'applicazione possa accedere alle risorse necessarie in modo sicuro e controllato.

2.2. Autenticazione Out-of-Band Utilizzando l'Authorization Code Flow

L'Authorization Code Flow out-of-band è un metodo di autenticazione progettato per applicazioni che non sono basate sul web, come le applicazioni desktop.

Questo flusso evita la necessità per l'applicazione di intercettare il redirect del browser al termine dell'autenticazione dell'utente, semplificando il processo.

Per configurare l'Authorization Code Flow out-of-band, è necessario ottenere un `client_id` dal provider di identità.

Questo client_id è associato a un `redirect_uri`, che in questo caso è un codice convenzionale uguale a `urn:ietf:wg:oauth:2.0:oob`. Questo URI speciale indica al provider di identità di utilizzare il flusso out-of-band.

Il flusso di autenticazione funziona in questo modo:

- **Richiesta di autorizzazione**: l'applicazione genera un URI di autorizzazione che include il `client_id`, il `redirect_uri` convenzionale `urn:ietf:wg:oauth:2.0:oob`, e altri parametri necessari. Questo URI può essere aperto manualmente dall'utente nel browser oppure l'applicazione può lanciare il browser in modo che apra questo indirizzo.
- **Autenticazione dell'utente**: l'utente inserisce le proprie credenziali e autorizza l'applicazione ad accedere alle risorse richieste.
- `Ricezione del codice di autorizzazione`: dopo l'autenticazione, il provider di identità mostra il codice di autorizzazione in una pagina web. L'utente copia manualmente questo codice e lo incolla in un form predisposto dall'applicazione. Questo approccio elimina la necessità per l'applicazione di intercettare il redirect del browser, semplificando il processo di autenticazione.
- **Richiesta del token**: l'applicazione utilizza il codice di autorizzazione per richiedere un token di accesso dal provider di identità.
- **Accesso alle risorse**: con il token di accesso, l'applicazione può ora accedere alle risorse protette per conto dell'utente.

Il seguente script bash è un esempio di come può essere implementato il meccanismo di autenticazione out-of-band.

```
#!/bin/bash
#
# requires: curl, jq, openssl, base64

TOKEN_AUTH_PATH="https://entaksi.eu/auth/realms/edoc/protocol/openid-connect/auth"
TOKEN_ENDPOINT="https://entaksi.eu/auth/realms/edoc/protocol/openid-connect/token"

CLIENT_ID=<client id fornito>

SCOPE="openid profile email"
CODE_VERIFIER=$(openssl rand -base64 64 | tr -d '\n' | tr '+/' '-_' | tr -d '=' | cut -c 1-64)
CODE_CHALLENGE=$(echo -n "$CODE_VERIFIER" | openssl dgst -sha256 -binary | base64 | tr -d '\n' | tr '+/' '-_' | tr
-d '=')
NONCE=$(openssl rand -hex 16)
STATE=$(openssl rand -hex 16)



# Build the authorization URL with the oob (out of band) response mode
AUTH_URL="$TOKEN_AUTH_PATH?\
response_type=code&\
client_id=${CLIENT_ID}&\
scope=${SCOPE}&\
state=${STATE}&\
```

```
nonce=${NONCE}&\
code_challenge=${CODE_CHALLENGE}&\
code_challenge_method=S256&\
redirect_uri=urn:ietf:wg:oauth:2.0:oob"

echo "Login in your browser using the following URL:"
echo "${AUTH_URL}"



echo
echo -n "Enter the authorization code: "
read AUTH_CODE



TOKEN_RESPONSE=$(curl -s -X POST "$TOKEN_ENDPOINT" \
                      -d "client_id=$CLIENT_ID" \
                      -d "grant_type=authorization_code" \
                      -d "code=$AUTH_CODE" \
                      -d "code_verifier=$CODE_VERIFIER" \
                      -d "redirect_uri=urn:ietf:wg:oauth:2.0:oob")

if ! echo "$TOKEN_RESPONSE" | jq . &> /dev/null; then
    echo "error: invalid json in response" 1>&2
    exit 1
fi

if echo "$TOKEN_RESPONSE" | jq -e '.error' &> /dev/null; then
    ERROR=$(echo "$TOKEN_RESPONSE" | jq -r '.error')
    ERROR_DESC=$(echo "$TOKEN_RESPONSE" | jq -r '.error_description')
    echo "error: $ERROR" 1>&2
    echo "$ERROR_DESC" 1>&2
    exit 1
fi

echo "success:"
echo "$TOKEN_RESPONSE" | jq

if echo "$TOKEN_RESPONSE" | jq -e '.id_token' &> /dev/null; then
    ID_TOKEN=$(echo "$TOKEN_RESPONSE" | jq -r '.id_token')
    PAYLOAD=$(echo "$ID_TOKEN" | cut -d '.' -f 2 | base64 -d 2>/dev/null || true)
    echo "id token payload:"
    echo "$PAYLOAD" | jq .

    PAYLOAD_NONCE=$(echo "$PAYLOAD" | jq -r '.nonce')
    if test "$PAYLOAD_NONCE" == "$NONCE"; then
        echo "nonce verified"
    else
        echo "nonce not verified"
    fi
fi
```

2.3. Autenticazione per Login su Dispositivi Utilizzando il Device Authorization Grant

OAuth2 Device Authorization Grant è utile quando l'ambiente in cui deve operare l'applicazione ha delle limitazioni nell'interfaccia utente tali da rendere difficile l'inserimento delle credenziali di autenticazione o l'apertura di link in un browser. Uno scenario comune sono i dispositivi embedded, quelli privi di tastiera o anche gli ambienti a linea di comando.

Il flusso è visibile nella pagina 4 della specifica RFC8628 OAuth 2.0 Device Authorization Grant le cui operazioni sono:

- Il dispositivo che deve accedere alle API richiede un codice di autenticazione al server per uno specifico `client_id`. Con questo codice si ottiene un URL e l'utente viene invitato a navigare su questo indirizzo utilizzando un altro dispositivo con

maggiori capacità di interazione. È anche possibile mostrare l'URL in un QR code, in modo che l'utente possa arrivare all'indirizzo richiesto inquadrando l'immagine con uno smartphone.

- Il dispositivo si mette in attesa e predispone un processo che periodicamente va a verificare sul server se l'autenticazione è stata completata.
- Nel frattempo l'utente apre l'indirizzo URL e segue il processo di autenticazione autorizzando il dispositivo ad accedere con le proprie credenziali.
- Dopo la conferma dell'utente, il dispositivo ottiene un token di autorizzazione.

Il seguente script bash è un esempio di come può essere implementato il meccanismo di autenticazione utilizzando il device code flow.

```
#!/bin/bash
#
# requires: curl, jq, openssl, base64

TOKEN_SERVICE_DEVICE_PATH="https://entaksi.eu/auth/realms/edoc/protocol/openid-connect/auth/device"
TOKEN_ENDPOINT="https://entaksi.eu/auth/realms/edoc/protocol/openid-connect/token"

CLIENT_ID=<client id fornito>

SCOPE="openid offline_access"
CODE_VERIFIER=$(openssl rand -base64 64 | tr -d '\n' | tr '+/' '-_' | tr -d '=' | cut -c 1-64)
CODE_CHALLENGE=$(echo -n "$CODE_VERIFIER" | openssl dgst -sha256 -binary | base64 | tr -d '\n' | tr '+/' '-_' | tr
-d '=')
NONCE=$(openssl rand -hex 16)
STATE=$(openssl rand -hex 16)



# 1) il device esegue la richiesta di codice di inizio autenticazione
DEVICE_CODE_RESPONSE=$(curl -s -X POST "$TOKEN_SERVICE_DEVICE_PATH" \
                            -d "client_id=$CLIENT_ID" \
                            -d "scope=$SCOPE" \
                            -d "code_challenge=$CODE_CHALLENGE" \
                            -d "code_challenge_method=S256" \
                            -d "nonce=$NONCE" \
                            -d "state=$STATE")

DEVICE_CODE=$(echo "$DEVICE_CODE_RESPONSE" | jq -r '.device_code')
VERIFICATION_URI_COMPLETE=$(echo "$DEVICE_CODE_RESPONSE" | jq -r '.verification_uri_complete')
INTERVAL=$(echo "$DEVICE_CODE_RESPONSE" | jq -r '.interval')

echo "Login in your browser using the following URL:"
echo "$VERIFICATION_URI_COMPLETE"
echo -n "waiting"

while true; do
    echo -n .

    # 2) il device esegue un polling fino al login dell'utente via browser
    TOKEN_RESPONSE=$(curl -s -X POST "$TOKEN_ENDPOINT" \
        -d "client_id=$CLIENT_ID" \
        -d "grant_type=urn:ietf:params:oauth:grant-type:device_code" \
        -d "device_code=$DEVICE_CODE" \
        -d "code_verifier=$CODE_VERIFIER")

    if ! echo "$TOKEN_RESPONSE" | jq . &> /dev/null; then
        echo "error: invalid json in response" 1>&2
        exit 1
    fi

    if echo "$TOKEN_RESPONSE" | jq -e '.error' &> /dev/null; then
        ERROR=$(echo "$TOKEN_RESPONSE" | jq -r '.error')

        # 3.a) l'utente nega l'autorizzazione
        if test "$ERROR" != "authorization_pending"; then
```

```
            ERROR=$(echo "$TOKEN_RESPONSE" | jq -r '.error')
            ERROR_DESC=$(echo "$TOKEN_RESPONSE" | jq -r '.error_description')
            echo "error: $ERROR" 1>&2
            echo "$ERROR_DESC" 1>&2
            exit 1
        fi

        sleep "$INTERVAL"
        continue
    fi

    # 3.b) l'utente accetta l'autorizzazione
    break
done

echo "success:"
echo "$TOKEN_RESPONSE" | jq

# 4) il device recupera il token di accesso
if echo "$TOKEN_RESPONSE" | jq -e '.id_token' &> /dev/null; then
    ID_TOKEN=$(echo "$TOKEN_RESPONSE" | jq -r '.id_token')
    PAYLOAD=$(echo "$ID_TOKEN" | cut -d '.' -f 2 | base64 -d 2>/dev/null || true)
    echo "id token payload:"
    echo "$PAYLOAD" | jq .

    PAYLOAD_NONCE=$(echo "$PAYLOAD" | jq -r '.nonce')
    if test "$PAYLOAD_NONCE" == "$NONCE"; then
        echo "nonce verified"
    else
        echo "nonce not verified"
    fi
fi
```

Lo script è configurato per utilizzare queste funzionalità aggiuntive:

- Lo scope `offline_access` richiede l'emissione di un token che può essere rinnovato a distanza di tempo. Il token offline può essere memorizzato dal dispositivo e recuperato anche dopo lo spegnimento.
- La procedura implementa il meccanismo PKCE, per garantire al server che il token non possa essere fornito a un client diverso da quello che ha iniziato la richiesta.
- Il codice `nonce`, invece, garantisce al client che il token proviene dallo stesso server con cui la richiesta era stata iniziata.

2.3.1. Rinnovo del token

Indipendentemente dalla modalità con cui è stato eseguita l'autenticazione il risultato finale è un oggetto JSON con un contenuto di questo tipo:

```
{
  "access_token": "eyJhbGciOiJSUzI1NiJ9.eyJqdG....",
  "expires_in": 1800,
  "refresh_token": "eyJhbGciOiJSUzI1NiJ9.eyJqdGk....",
  "token_type": "bearer",
  "scope": "offline_access"
}
```

Nella proprietà `access_token` è contenuto il token che va indicato nell'intestazione Authorization: Bearer per fare le chiamate alle API. Gli altri valori significativi in questa struttura sono:

- `expires_in`, che indica il tempo di vita in secondi del token, conteggiato a partire dal momento in cui è stato emesso
- `refresh_token`, che riporta un token da utilizzare per innovare l'accesso.

La scadenza della validità del token è un aspetto importante della sicurezza del meccanismo OpenID Connect. Se il token venisse compromesso, sarebbe valido solo per un periodo limitato di tempo.

Quando il token è prossimo alla scadenza, il client può rinnovarlo usando il refresh token. L'operazione consente di ottenere un nuovo token per continuare ad chiamare le API.

Se il client non è pubblico, e quindi è dotato di un `client_secret`, questa chiamata deve usare l'autenticazione Basic con il `client_id` e il `client_secret`, esattamente come la richiesta di Access Token.

Nei parametri va indicato il valore del Refresh Token.

```
POST /auth/realms/edoc/protocol/openid-connect/token
Authorization: Basic abcdefghilmnopqrstuvz
Content-Type: application/x-www-form-urlencoded

grant_type=refresh_token&refresh_token=tGzv3JOkF0XG5Qx2TlKWIA
```

La risposta ha la stessa struttura della richiesta di Access Token e da essa si ricavano i nuovi valori di Access Token e Refresh Token che sostituiscono i precedenti.

2.4. Documentazione delle API e ambiente di interazione on line

Le API REST del sistema sono documentate secondo lo standard OpenAPI (versione 2, nota anche come Swagger) che consiste in un file JSON contenente tutte le informazioni che descrivono le modalità di invocazione delle varie risorse.

La documentazione è sempre disponibile all'indirizzo https://entaksi.eu/api/edoc/v1/swagger.json. Esistono diversi strumenti di sviluppo che, a partire da queste informazioni, possono generare automaticamente i componenti client in vari ambienti di sviluppo, tra i quali Java, .NET, PHP, Ruby, ecc.

Un ambiente di interazione con le API è disponibile all'indirizzo https://entaksi.eu/api/edoc/v1-doc/ dove ciascuna risorsa è descritta in dettaglio ed è anche possibile invocare i servizi componendo manualmente i parametri necessari.

2.5. Ambiente di test

Per gli sviluppatori che intendono provare l'integrazione dei loro prodotti con i servizi *eDoc API* è disponibile un ambiente di test del tutto equivalente a quello di produzione e che risponde agli stessi indirizzi utilizzando `test.entaksi.eu` al posto di `entaksi.eu`.

Tabella 1. Ambienti di test e produzione

Servizio	Test	Produzione
Documentazione OpenAPI	https://test.entaksi.eu/api/edoc/v1/swagger.json	https://entaksi.eu/api/edoc/v1/swagger.json
Ambiente di interazione con le API	https://test.entaksi.eu/api/edoc/v1-doc/	https://entaksi.eu/api/edoc/v1-doc/
Console di gestione del servizio	https://test.entaksi.eu/a/	https://entaksi.eu/a/

L'accesso alla console di gestione di produzione avviene anche tramite il link http://console.entaksi.eu

L'ambiente di test e quello di produzione utilizzano lo stesso sistema di autenticazione, perciò i link che iniziano con https://entaksi.eu/auth rimangono con l'host `entaksi.eu` anche durante l'accesso all'ambiente di test.

3. Modello generale del sistema

Il modello del servizio è organizzato nelle seguenti entità anagrafiche:

- L'**Utente**: è l'utente che accede al servizio identificato dalle credenziali di accesso.
- L'**Azienda**: è la persona fisica o giuridica che ha stipulato un contratto per l'accesso ai servizi del sistema. L'**Azienda** è associata ad uno o più utenti. Dal punto di vista archivistico l'azienda corrisponde al produttore dei documenti. Ogni Azienda è collegata a un Contratto che definisce i termini di accesso al servizio nell'ambito del periodo di validità.
- Il **Contratto**: è l'entità che definisce i termini di accesso al servizio e comprende una o più aziende che condividono l'accesso al sistema. Il contratto ha uno o più utenti i quali supervisionano le aziende gestite dal contratto. Il contratto definisce gli intervalli di validità dei servizi erogati.
- Il **Rivenditore**: è l'entità che stipula i contratti di accesso al servizio. Ha uno o più contratti e uno o più utenti.

Nella parte del sistema che si occupa della gestione della fattura elettronica sono previste le seguenti entità:

- La **Fattura**: è l'entità che rappresenta la Fattura XML, trasmessa o ricevuta, ed è collegata ad una **Azienda**.
- La **Ricevuta**: ogni fattura ha una o più ricevute che corrispondono ai messaggi ricevuti dal Sistema di Interscambio nell'ambito del ciclo di gestione della fattura.

Nella parte del sistema che si occupa dell'archiviazione elettronica si trovano le seguenti entità:

- Il **Pacchetto di versamento** (PDV): rappresenta il contenitore con cui i documenti vengono versati nel sistema di archiviazione. Il PDV è definito formalmente nel Manuale della conservazione.
- Il **Rapporto di versamento** (RDV): è il documento digitale in formato XML con cui il sistema di archiviazione attesta l'avvenuto versamento di un PDV.
- Il **Pacchetto di archiviazione** (PDA): è il contenitore in cui i documenti vengono archiviati nel sistema per la conservazione a lungo termine.
- L'**Indice di conservazione** (`IdPda`): è l'indice che, mediante firma digitale e marca temporale, rappresenta la prova di conservazione dei documenti contenuti nel PDA.
- Il **Pacchetto di distribuzione** (PDD): è il contenitore in cui i documenti vengono raggruppati per essere distribuiti in seguito ad una richiesta di distribuzione. Insieme ai documenti sono distribuite anche le prove di conservazione con cui essi sono stati archiviati in modo dare piena validità fiscale e civile ai documenti recuperati dall'archivio.

Poiché la gestione della fattura elettronica è integrata nel sistema, al termine della fase di gestione le fatture formano automaticamente dei pacchetti di versamento tramite i quali ha luogo il processo di conservazione.

Per gli altri documenti e per le fatture elettroniche gestite al di fuori del sistema eDoc il caricamento nel sistema di conservazione avviene sempre Pacchetti di Versamento formati esternamente al sistema.

4. Attivazione di una azienda

L'utente che ha le credenziali associate ad un rivenditore può, tramite l'invocazione delle API oppure tramite la console di gestione del servizio, definire un'azienda e attivare i servizi che consentono di operare sul sistema.

Tuttavia normalmente risulta più pratico utilizzare un flusso operativo che consenta all'utente finale di registrare autonomamente le proprie credenziali e la propria azienda mediante un prodotto software gestito dal rivenditore e quindi ad esso riconducibile.

eDoc API implementa questa modalità di attivazione del servizio mediante l'uso del `client_id` utilizzato durante la chiamata alle API REST. Associando il `client_id` ad un rivenditore risulta possibile per il sistema riconoscere il software e associare quindi l'azienda a quel rivenditore.

Lo scenario si può riassumere nei seguenti passaggi:

- Il rivenditore X dispone di un client del servizio *eDoc API* (ad esempio un'applicazione desktop o un servizio web per la gestione dei documenti). A questo client è associato un determinato `client_id` e (eventualmente) una password.
- Il cliente finale registra le proprie credenziali sul sistema di autenticazione collegandosi all'indirizzo https://console.entaksi.eu/
- Il cliente finale utilizza le credenziali con l'applicazione fornita dal rivenditore. L'applicazione, utilizzando l'API POST `/aziende` definisce una nuova azienda.

Tramite il valore `client_id` il sistema riconosce l'applicazione e associa l'azienda a quel **Rivenditore** tramite un **Contratto** creato al momento per quella azienda oppure associando l'azienda ad un contratto esistente, in base alla configurazione impostata dal rivenditore.

4.1. Associazione di un `client_id` al rivenditore

Il codice `client_id` generato da un amministratore del sistema può essere associato ad un rivenditore tramite le seguenti API:

- `GET /rivenditori/{idRivenditore}/clients`: recupera l'elenco dei `client_id` associati al rivenditore.
- `PUT /rivenditori/{idRivenditore}/clients/{client_id}`: associa il `client_id` al rivenditore.
- `DELETE /rivenditori/{idRivenditore}/clients/{client_id}`: rimuove l'associazione del `client_id` al rivenditore.

5. Gestione delle fatture XML

eDoc API gestisce il ciclo completo della Fattura XML per la Pubblica Amministrazione (Fattura PA) e per i privati (Fattura B2B), in particolare:

- Accetta il caricamento di una fattura ed esegue alcune verifiche preliminari.
- Firma digitalmente la fattura.
- Invia la fattura al Sistema di Interscambio.
- Riceve le notifiche emesse dal Sistema di Interscambio rendendole disponibili all'utente.
- A ciclo di gestione completato, inserisce la fattura e le notifiche in un Pacchetto di versamento per l'archiviazione a lungo termine.

5.1. Trasmissione della Fattura XML

Per caricare una Fattura XML è possibile utilizzare una delle seguenti API:

1. `POST /fatture/xml`
2. `POST /aziende/{idAzienda}/fatture/xml`
3. `POST /aziende/{idAzienda}/fatture/uploadXml`

Le prime due API contengono nel corpo della richiesta il contenuto XML della fattura.

La terza API, invece, esegue un upload multipart del file XML contenente la fattura che deve essere in una parte di nome `invoice`.

Nella prima API il sistema ricava l'azienda dal contenuto XML della fattura stessa, tuttavia questo è possibile solo se l'azienda non è divisa in diverse unità organizzative oppure se la fattura è riferita all'unità organizzativa di default.

Nella seconda API il valore di `idAzienda` può contenere il codice di una unità organizzativa, in questo caso la fattura viene gestita e archiviata per quella specifica unità organizzativa.

La terza API è equivalente alla seconda e risulta più semplice da usare quando la fattura da trasmettere è già stata salvata in un file.

Dopo il caricamento è possibile seguire il ciclo di gestione della fattura mediante la seguente API:

- `GET /fatture/{idFattura}`

`{idFattura}` può essere l'identificativo numerico assegnato durante il caricamento oppure il nome del file (senza estensione `.xml`), anch'esso assegnato durante il caricamento.

Nell'oggetto restituito è disponibile una proprietà `stato` che può contenere uno dei valori riportati nel capitolo Stati della fattura,

5.2. Monitoraggio delle fatture trasmesse

Dopo la trasmissione di una fattura al sistema, essa viene elaborata secondo le seguenti fasi:

- Firma del file
- Trasmissione al Sistema di Interscambio
- Attesa dei messaggi di notifica
- Conclusione del processo di gestione
- Archiviazione

Per ciascuna fattura è possibile rilevare lo stato con l'API `GET /fatture/{idFattura}`, oppure è possibile rilevare lo stato di tutte le fatture di una azienda con l'API `GET /aziende/{idAzienda}/fatture`. In quest'ultimo caso conviene utilizzate dei criteri di selezione per limitare l'elenco, ad esempio, ad un certo intervallo di date:

```
GET /aziende/{idAzienda}/fatture?data_inizio=2017-10-01&data_fine=2017-10-31
```

Il caso d'uso più comune è quello in cui, dopo la trasmissione della fattura, l'utente vuole verificare l'arrivo dei messaggi del

Sistema di Interscambio che sono:

- RC (ricevuta di consegna), che attesa la consegna della fattura al destinatario
- NS (notifica di scarto), che indica un errore nella fattura
- MC (mancata consegna), che attesta un problema temporaneo nella consegna della fattura al destinatario
- NE (notifica di esito), che riporta l'accettazione o il rifiuto della fattura da parte del destinatario
- DT (decorrenza termini), che attesta la scadenza del periodo a disposizione del destinatario per inviare un esito
- AT (avvenuta trasmissione), che attesta un problema permanente nella consegna della fattura al destinatario

Per esaminare i messaggi ricevuti per ciascuna fattura è possibile usare la seguente API:

```
GET /fatture/{idFattura}/ricevute
```

Oppure, la seguente API riporta i messaggi ricevuti per tutte le fatture di un'azienda:

```
GET /aziende/{idAzienda}/ricevute
```

È anche possibile ottenere i messaggi di tutte le fatture di tutte le aziende a cui l'utente ha accesso:

```
GET /ricevute
```

Le ultime due API possono sfruttare il flag di lettura per filtrare solo i messaggi che non sono stati letti in precedenza, seguendo questo flusso di lavoro:

Recuperare tutti i messaggi da leggere:

```
GET /aziende/{idAzienda}/ricevute?flag=true
```

Elaborare tutti messaggi (ad esempio notificando l'utente o aggiornando il database) e marcare la notifica come "già letta" con l'API:

```
POST /ricevute/{idRicevuta}/unflag
```

Dopo aver impostato la notifica come già letta, essa non verrà più restituita dall'API in cui è indicato `flag=true`.

5.3. Ricezione delle fatture

Il meccanismo del flag di lettura è applicabile anche alle fatture e può risultare utile per gestire le fatture ricevute mediante il codice destinatario associato al canale di ricezione.

La seguente API consente di ottenere tutte le fatture ricevute da leggere:

```
GET /aziende/{idAzienda}/fatture?flag=true&dir=R
```

Quindi la fattura può essere marcata come già letta con:

```
POST /fatture/{idFattura}/unflag
```

Il contenuto XML delle singole fatture può essere recuperato con:

```
GET /fatture/{idFattura}.xml
```

La stampa PDF della singolo fattura può essere recuperata con:

```
GET /ordini/{idOrdine}.pdf
```

La stampa può essere richiesta in formato ministeriale indicando `stile=sdi` o in formato personalizzato da Entaksi indicando `stile=entaksi`.

5.4. Trasmissione della notifica Esito Committente

Per le fatture ricevute, l'azienda destinataria può inviare una notifica Esito Committente con la quale comunica all'ente che ha emesso la fattura se essa è stata accettata o rifiutata.

Se l'operazione non viene fatta entro 15 giorni, il ciclo di gestione della fattura elettronica si intende comunque concluso e il sistema di interscambio emetterà un messaggio DT (Decorrenza Termini) per entrambi i soggetti. La fattura può essere accettata o rifiutata, ma questa comunicazione tra soggetto emittente e destinatario avverrà con altri mezzi e la fattura dovrò in ogni caso essere registrata nella contabilità di entrambi.

Per trasmettere la notifica di esito committente di una fattura va usata la seguente API:

```
POST /fatture/{idFattura}/ricevute
```

Il corpo della richiesta conterrà i dati della notifica. In JSON:

```
{
  "esito": "string", ①
  "note": "string", ②
  "riferimento": {
    "numero": "string",  ③
    "anno": 0, ④
    "posizione": 0 ⑤
  },
  "codice": "string" ⑥
}
```

1. Esito da comunicare al mittente, `EC01` per "Accettata", `EC02` per "Rifiutata"
2. Note da comunicare per giustificare l'esito
3. Numero del documento
4. Anno del documento
5. Posizione del documento nel lotto (1 se il file conteneva una sola fattura)
6. Codice identificativo del messaggio (opzionale)

5.5. Fatturazione elettronica tra Italia e San Marino

A partire dal 1 luglio 2022, le fatture relative alle cessioni di beni effettuate tra soggetti residenti, stabiliti o identificati nel territorio dello Stato Italiano e soggetti residenti nella Repubblica di San Marino sono trasmesse e ricevute tramite il Sistema di Interscambio (SdI).

L'Ufficio Tributario di San Marino trasmette le fatture elettroniche dei cedenti/prestatori della Repubblica di San Marino e riceve le fatture elettroniche dei cedenti/prestatori Italiani esclusivamente tramite il nodo attestato al Sistema di Interscambio il cui codice destinatario alfanumerico associato è "2R4GTO8".

L'HUB-SM è il servizio dell'Ufficio Tributario di San Marino che si occupa della raccolta, smistamento e trasmissione delle fatture elettroniche agli operatori economici e alle Autorità competenti.

Il formato e la nomenclature del file fattura rispetta il formato definito dall'Amministrazione Finanziaria italiana e utilizzato nel Sistema di Intercambio (SdI).

5.6. Fatture passive di operatori economici di San Marino e cedente Italiano

Quando un operatore economico Italiano emette una fattura destinata a un operatore economico di San Marino, quest'ultimo è tenuto ad inviare un file di integrazione con l'indicazione di tipo merce e aliquota sulle righe di dettaglio e nel riepilogo e con l'indicazione di imposta nel riepilogo, ai fini del calcolo dell'imposta sulle importazioni monofase.

Per caricare un file di integrazione nel formato definito dalle specifiche è possibile utilizzare la API seguente:

```
POST /fatture/{idFattura}/integrazioneSM
```

che esegue un upload multipart del file XML contenente il file di integrazione.

Il sistema accetta il file di integrazione, lo registra come una notifica SM e lo invia ad HUB-SM.

Se la fattura ricevuta tramite HUB-SM è in uno dei seguenti casi:

- errore nel codice identificativo del Cessionario di San Marino
- fattura già precedentemente assolta
- fattura per triangolazione Estero/IT/SM, che non deve arrivare su HUB-SM in formato elettronico

è possibile respingerla verso l'HUB-SM con la API seguente:

```
PUT /fatture/{idFattura}/respingi
```

6. Gestione delle strutture dell'azienda

Utilizzando il flusso di lavoro ordinario, dove le fatture vengono caricate sull'azienda tramite `POST /fatture/xml` o tramite `POST /aziende/{idAzienda}/fatture`, i documenti vengono considerati come versati da un produttore identificato da una struttura di default dell'azienda.

Se l'azienda è organizzata in più strutture organizzative che richiedono di essere differenziate nell'archiviazione occorre identificare la struttura a cui associare i documenti.

Nelle API, per indirizzare un'azienda con una struttura organizzativa va aggiunto un codice alfanumerico alla partita IVA separato da un punto (`.`).

Il codice azienda è quindi definito in generale con le seguenti regole:

```
<suffisso-struttura> ::= "." <id-struttura>
<id-azienda> ::= <partita-iva> <suffisso-struttura> | ""
```

Se il codice azienda è formato dalla sola partita IVA, allora i documenti sono associati alla struttura di default dell'azienda.

Se il codice azienda è formato dalla partita IVA più il suffisso con il codice della struttura separato da un punto, allora i documenti sono associati alla struttura indicata.

6.1. Caricamento diretto della Fattura XML

La API `POST /fatture/xml` consente di caricare una Fattura elettronica direttamente tramite il file XML. Durante il caricamento la partita IVA viene ricavata dal contenuto del file XML per identificare l'azienda.

Questa API, quindi, può essere utilizzata solo se i documenti sono diretti alla struttura di default.

Se i documenti devono essere posizionati in una specifica struttura dell'azienda le informazioni contenute nell'XML non sono sufficienti a identificare la struttura, perciò questa API non può essere utilizzata e deve invece essere utilizzata la seguente API che comprende l'identificazione della struttura: `POST /aziende/{idAzienda}/fatture/xml`.

Il sistema prevede i seguenti controlli:

1. La partita IVA indicata nell'XML deve corrispondere a quella in `{idAzienda}`
2. Il sezionale indicato nel documento non deve essere usato in strutture diverse

6.2. Diritti di accesso degli utenti

La struttura di default individuata tramite la sola partita IVA dell'azienda e la struttura specifica individuata dalla partita IVA più il suffisso hanno una gestione degli accessi distinta per cui, ad esempio, gli utenti che hanno accesso alla struttura di default non ereditano l'accesso alle strutture specifiche, né viceversa.

7. Pacchetti di versamento

Il pacchetto di versamento è l'entità con cui i documenti vengono introdotti nel sistema di conservazione. *eDoc API* gestisce il caricamento dei pacchetti di versamento secondo vari formati definiti nel manuale della conservazione.

Il formato del pacchetto di versamento è definito da un codice che può assumere questi valori:

Codice	Descrizione
`F000`	Fatture elettroniche inviate
`F001`	Fatture elettroniche inviate gestite dal sistema
`F002`	Fatture elettroniche ricevute
`F003`	Fatture elettroniche ricevute gestite dal sistema
`F005`	Pacchetto di versamento PEC eml
`F006`	Ordini elettronici
`F998`	Pacchetto di un precedente conservatore
`F997`	Pacchetto copia digitale documento analogico
`F999`	Pacchetto di versamento eDoc generico

I pacchetti con formato `F999` (generici) devono essere dotati di un indice del contenuto in cui sono dichiarati i file contenuti nel pacchetto, il registro di archiviazione in cui devono essere versati e altre informazioni necessarie per l'archiviazione, tra le quali i metadati da associare ai documenti.

Per tutti gli altri formati, escluso `F001` e `F003`, il pacchetto non contiene un indice del contenuto, ma solo i file da archiviare. Questi formati sono dedicati a quei documenti nativamente elettronici che, per le loro caratteristiche, consentono l'estrazione dei metadati in modo automatico. Mediante apposite convenzioni descritte nelle specifiche di questi formati, i documenti contenuti vengono elaborati dal sistema ricavando automaticamente le informazioni necessarie per l'archiviazione.

I formati `F001` e `F003` (fatture elettroniche trasmesse e ricevute gestite dal sistema) sono del tutto analoghi ai formati `F000` e `F002` (fatture elettroniche trasmesse e ricevute gestite da altri sistemi). Questi pacchetti vengono generati internamente al sistema durante la gestione delle fatture elettroniche e non sono mai caricati direttamente dall'utente.

7.1. Stati del pacchetto di versamento

Il pacchetto di versamento può trovarsi in uno degli stati riportati nel paragrafo Tabella dei codici degli stati del PDV.

7.2. Formato `F000` Fatture elettroniche inviate

`F000` è un particolare formato del pacchetto di versamento che consiste in un file ZIP contenente le Fatture elettroniche in formato XML 1.0, 1.1 o 1.2 il cui invio è già stato gestito con altri sistemi. I file delle fatture devono essere corredati delle relative notifiche.

I file contenuti nell'archivio ZIP devono mantenere il nome file originale così come definito da chi ha emesso la fattura.

I file relativi alle fatture XML contenuti nello ZIP possono essere in formato XML con firma XADeS, oppure in formato P7M, secondo le specifiche definite dal Sistema di Interscambio.

I file delle notifiche sono sempre firmati in modalità XADeS dal Sistema di Interscambio.

Il processo di gestione di questo PDV prevede una fase di validazione in cui il pacchetto viene esaminato e viene prodotto un rapporto di versamento (RDV) e un esito dalla validazione.

In base all'esito il pacchetto può essere accettato o rifiutato dal sistema di conservazione.

Se il pacchetto viene rifiutato il Rapporto di Versamento contiene uno o più errori che spiegano il motivo del rifiuto.

7.2.1. Validazione del PDV `F000`

La validazione consiste nei seguenti controlli:

- Il file ZIP deve contenere solo i file attesi, cioè le fatture XML e le relative notifiche. La presenza di altri file provoca il rifiuto del pacchetto.
- La firma digitale dei file (fatture e notifiche) deve essere valida.

Durante la validazione la procedura predispone le unità documentali relative alle fatture contenute secondo i seguenti criteri:

- I file delle fatture vengono aggregati con le rispettive notifiche e si determina se la fattura è stata scartata oppure ha completato il ciclo di gestione
- Le fatture scartate per le quali esiste un'altra fattura che ha completato il ciclo di gestione vengono annesse (insieme alle relative notifiche) alla fattura con lo stesso numero che ha completato il ciclo di gestione
- Le fatture scartate per le quali *non* esiste un'altra fattura che ha completato il ciclo di gestione vengono considerate come unità documentali indipendenti
- Le fatture rifiutate per le quali esiste un'altra fattura che ha completato il ciclo di gestione vengono annesse (insieme alle relative notifiche) alla fattura che ha completato il ciclo di gestione con lo stesso numero
- Le fatture rifiutate per le quali *non* esiste un'altra fattura che ha completato il ciclo di gestione rimangono come unità documentali indipendenti

Le fatture che non hanno le notifiche producono un messaggio di avvertimento nella ricevuta di versamento. Senza le notifiche non è possibile stabilire se la fattura è stata accettata, scartata, accettata, ecc e neanche se è stata spedita. Pertanto i metadati relativi a questa informazioni non vengono valorizzati.

7.3. Formato `F001` Fattura XML gestite dal sistema

Il formato `F001` si riferisce ai Pacchetti di Versamento tramite i quali le Fatture PA gestite dal sistema vengono messe in conservazione.

Il ciclo di gestione delle Fatture PA dura fino a 15 giorni, includendo la possibilità che il documento vada in decorrenza termini (`DT`) o in impossibilità di recapito (`AT`).

Volendo includere anche la possibilità di riemissione delle fatture dopo notifica di esito (`NE`) negativa (modalità non ammessa da questo sistema), i tempi possono allungarsi anche in modo non precisamente definito.

Questa eventualità dovrebbe essere considerata come un'eccezione da gestire a parte, tuttavia essa viene contemplata nell'algoritmo definito di seguito per la formazione dei PDV F001.

Nella norma una fattura datata alla fine del mese *X* dovrebbe essere trasmessa nei primi giorni del mese *X + 1* e si può ritenere che alla fine del mese *X + 1* il suo processo di gestione si sia concluso.

L'algoritmo descritto di seguito avvia la formazione del PDV delle fatture del mese *X* nel mese *X + 1* e lo conclude appena possibile nel mese *X + 2*. Ad esempio, nel mese di Maggio 2015 si formano i PDV delle fatture con data nel mese di Aprile 2015.

Questi PDV si chiudono appena possibile nel mese di Giugno 2015.

L'algoritmo non consente di aprire un PDV `F001` se prima non si è chiuso il precedente.

Con cadenza quotidiana, alle 18:00, esclusi i primi 10 giorni del mese, si esegue il seguente algoritmo per l'apertura e la formazione del PDV:

- Per ogni struttura (leggi azienda)
  - Verifica se esiste un PDV con formato `F001` in stato `P01` (aperto) per la struttura.
  - Se sì, considera quel PDV, altrimenti crea un nuovo PDV intestato alla struttura registrando la data di creazione
  - Sia data-limite il primo giorno del mese in cui è stato creato il PDV
  - Per ogni fattura intestata alla struttura che rispetta tutte le seguenti condizioni:
    - la data-documento è precedente a data-limite

        - è priva del riferimento al pdv
        - si trova in uno dei seguenti stati: `S06` (attesa notifiche), `S07` (decorrenza termini), `S08` (impossibilità di recapito), `S09` (accettata), `S10` (rifiutata)
            - Imposta il PDV nella fattura

Con cadenza quotidiana, alle 03:00, si esegue il seguente algoritmo per completare la formazione del PDV:

- Per ogni PDV con formato `F001` in stato `P01` (aperto)
    - Per ogni fattura contenuta nel PDV
        - Se la fattura si trova in uno dei seguenti stati: `S07` (decorrenza termini), `S08` (impossibilità di recapito), `S09` (accettata), `S10` (rifiutata)
            - Allora esegue l'azione `A12_ARCHIVE` sulla fattura che porta lo stato a `S16_PRONTA_PER_CONSERVAZIONE`
    - Se tutte le fatture contenute nel PDV si trovano nello stato `S16` e la data di creazione del PDV si trova nel mese precedente alla data attuale
        - Allora avanza il PDV allo stato `S02` Elaborazione in corso

Il meccanismo consente di rispettare il seguente comportamento:

- Finché c'è un PDV con formato `F001` aperto non ne viene creato un altro
- Nel PDV non possono entrare fatture successive alla fine del mese precedente alla sua formazione
- Il PDV non viene chiuso prima del mese successivo alla sua formazione
- Il PDV non viene chiuso se contiene fatture il cui processo non è ancora concluso

7.4. Formato `F002` Fatture elettroniche ricevute

Utilizzando questo formato l'utente ha la possibilità di versare in conservazione le fatture elettroniche ricevute, sia per i soggetti privati che ricevono fatture B2B, sia per le fatture elettroniche ricevute dagli enti della pubblica amministrazione.

Il pacchetto `F002` è un file ZIP che contiene le fatture elettroniche e le relative notifiche in modo analogo al formato `F000`.

7.5. Formato `F003` Fatture elettroniche ricevute gestite dal sistema

Il pacchetto `F003` viene generato internamente dal sistema il versamento delle fatture ricevute.

7.6. Formato `F005` PEC eml

Il formato `F005` si riferisce ad un pacchetto di versamento per l'archiviazione di messaggi di posta elettronica certificata (PEC).

Il formato consiste in un file ZIP contenente uno o più file `*.eml`.

Ciascun file `*.eml` è un messaggio email in formato RFC-822.

Il pacchetto deve contenere esclusivamente file `*.eml` raggruppati in una o più cartelle aventi come nome l'indirizzo della casella PEC che si sta archiviando. Ad esempio:

- mario.rossi@pec.it
    - messaggio1.eml
    - messaggio2.eml
- aldo.bianchi@pec.it
    - messaggio3.eml
    - messaggio4.eml

I file contenuti nella cartella di una casella devono essere messaggi trasmessi o ricevuti da quella casella. Un file `.eml` che si trova nella cartella di una casella senza essere né trasmesso né ricevuto da quella casella provoca un errore di validazione e lo scarto del PDV con errore `E024`.

All'interno delle cartelle relative alle caselle i file `.eml` possono essere organizzati in sotto cartelle, ma questo non influenza l'elaborazione. I file `.eml` di una cartella vengono considerati tutti insieme come se non ci fossero le sottocartelle.

Ognuno dei file di una casella PEC può essere:

1. Un messaggio inviato, privo di firma S/MIME
2. Un messaggio ricevuto da un'altra casella PEC, dotato di S/MIME
3. Un messaggio ricevuto da un'altra casella non PEC, privo di firma S/MIME
4. Un messaggio ricevuto dal gestore della casella PEC che si sta archiviando relativo all'accettazione di un messaggio inviato, dotato di firma S/MIME
5. Un messaggio ricevuto dal gestore della casella PEC del destinatario relativo alla consegna di un messaggio inviato, dotato di firma S/MIME

7.7. Formato F997 Pacchetto copia digitale di documento analogico

Il formato F997 è riservato all'archiviazione di copie digitali di documenti analogici. Qualora il produttore abbia delegato Entaksi alla firma digitale per la conservazione sostitutiva di documenti analogici il caricamento di questo tipo di pacchetto viene abilitato sulla console.

Il metadato `terms:format` sul `fileGroup` valorizzato come `analogico` indica che il documento è una copie digitale di documento analogico. In questo caso se i documenti non sono stati versati nel precedente anno solare, ma hanno una data antecedente, vengono validati con firma digitale e versati con questo tipo di pacchetto.

La mancanza del metadato `terms:format` indica che il processo di trasformazione in documento informatico è stato eseguito dal produttore tramite l'apposizione della firma digitale al documento. In tal caso la procedura di validazione del pacchetto verifica la presenza e la validità della firma digitale.

7.8. Formato F998 Pacchetto di precedente conservatore

Il formato F998 è dedicato all'archiviazione di documenti provenienti da un precedente conservatore.

Il passaggio dei documenti da un conservatore all'altro avviene mediante i Pacchetti di Distribuzione per interoperabilità con i quali il precedente conservatore rende disponibili i documenti archiviati.

Questi Pacchetti di Distribuzione possono essere caricati nel sistema di conservazione Entaksi come pacchetti di versamento F998 e archiviati come tali. In una fase successiva il contenuto di questi pacchetti può essere riorganizzato secondo la struttura del sistema di conservazione mantenendo traccia della prova di conservazione originale già generata dal precedente conservatore.

Prima di versare dei pacchetti F998 nel sistema è opportuno contattare l'assistenza all'indirizzo assistenza@entaksi.eu.

7.9. Formato F999 Pacchetto generico con indice

Il formato F999 definisce i pacchetti di versamento di documenti generici per i quali è necessario includere nel pacchetto anche un file indice dove sono riportate le informazioni necessarie per l'archiviazione.

Nel capitolo seguente sono illustrati alcuni esempi dell'indice dei pacchetti di versamento in formato F999 per varie tipologie documentali.

8. Esempi di Pacchetti di Versamento

8.1. Tipi documento

I tipi di documento presenti nel sistema sono:

Tabella 2. Tipi documento

Codice	Descrizione
D01	Fattura attiva
D02	Fattura passiva
D03	Nota variazione aumento
D04	Nota variazione diminuzione
D05	Documento di trasporto
D06	Scontrino
D07	Ricevuta
D08	Bolla
D09	Libro giornale
D10	Libro inventari
D11	Libro mastro
D12	Registro cronologico
D13	Libro cespiti
D14	Registro Irpef
D15	Registro fatture acquisto
D16	Registro acquisti agenzie viaggio
D17	Registro fatture emesse
D18	Registro fatture in sospeso
D19	Registro corrispettivi
D20	Giornale fondo
D21	Registro corrispettivi agenzie viaggio
D22	Registro emergenza Iva
D23	Bollettario
D24	Registro prima nota

Codice	Descrizione
D25	Registro unico Iva
D26	Registro riepilogativo Iva
D27	Registro sezionale Iva acquisiti intra-UE
D28	Registro acquisti intra-UE non commerciali
D29	Registro trasferimenti intra-UE
D30	Registro dichiarazioni d'intenti emesse
D31	Registro dichiarazioni d'intenti ricevute
D32	Registro omaggi
D33	Registro memoria produzione contrassegno
D34	Registro lavorazione produzione contrassegno
D35	Registro carico produzione contrassegno
D36	Registro scarico produzione contrassegno
D37	Registro di beni in deposito
D38	Registro di beni in conto lavorazione
D39	Registro di beni in comodato
D40	Registro di beni in prova
D41	Registro sezionale Iva interno
D42	Registro carico stampati fiscali
D43	Registro società controllanti e controllate
D44	Registro carico scarico regime margine metodo analitico
D45	Registro acquisti regime margine metodo globale
D46	Registro vendite regime margine metodo globale
D47	Registro carico centri elaborazione dati
D48	Registro scarico centri elaborazione dati
D49	Registro somme ricevute in deposito
D50	Registro editori
D51	Libro soci
D52	Libro obbligazioni

Codice	Descrizione
D53	Libro adunanze e delibere di assemblee
D54	Libro adunanze e delibere del consiglio di amministrazione
D55	Libro adunanze e delibere del collegio sindacale
D56	Libro adunanze e delibere del comitato esecutivo
D57	Libro adunanze e delibere delle assemblee azionisti
D58	Altri registri
D5802	Determine
D5803	Nota variazione aumento ricevuta
D5804	Nota variazione diminuzione ricevuta
D59	Unico persone fisiche
D60	Unico società di persone
D61	Unico società di capitali
D62	Unico enti non commerciali
D63	Irap persone fisiche
D64	Irap Società persone
D65	Irap Società capitale
D66	Irap enti non commerciali ed equiparati
D67	Irap amministrazioni ed enti pubblici
D68	Modello730
D69	Modello consolidato nazionale e mondiale
D70	Modello Iva
D71	Modello Iva VR richiesta rimborso credito Iva
D72	Modello Iva 26LP/2006 prospetto liquidazioni periodiche
D73	Modello Iva 74 bis
D74	Comunicazione annuale dati Iva
D75	Modello richiesta rimborso credito Iva trimestrale
D76	Modello dati contenuti dichiarazione intento ricevute
D77	Modello 770 semplificato

Codice	Descrizione
D78	Modello 770 ordinario
D79	Modello certificazione CUD
D80	Modello F23
D81	Modello F24
D82	Modelli allegati alla Dichiarazione dei Redditi Modello Unico
D83	Modelli annotazione separata
D84	Ricevuta presentazione modelli dichiarazione
D85	Altri documenti
D8501	Contratti
D8502	Cedolini
D8503	Manuali
D85PEC	PEC
meta	Registro meta

8.2. Esempio pacchetto con documenti D01 Fatture emesse e D02 Fatture ricevute

Il pacchetto contiene un documento appartenente alla tipologia D01, *Fatture emesse* e un documento appartenente alla tipologia D02, *Fatture ricevute*.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<pdv xmlns:terms="http://purl.org/dc/terms/" xmlns="http://entaksi.eu/schemas/econ/1.0/"
xmlns:dc="http://purl.org/dc/elements/1.1/">
    <formato>F999</formato> ①
    <fileGroup> ②
        <dc> ③
            <terms:type>Fattura Emessa</terms:type> ④
            <terms:subject>Fattura SEZ/33 del 01/04/2015 Cliente Srl</terms:subject> ⑤
            <terms:format>analogico</terms:format> ⑥
        </dc>
        <metadata key="produttore:ragionesociale">Acme Srl</metadata> ⑦
        <metadata key="produttore:idfiscale">IT12345678904</metadata>

        <metadata key="destinatario:ragionesociale">Cliente Srl</metadata> ⑧
        <metadata key="destinatario:idfiscale">IT01621900479</metadata>

        <metadata key="documento:data">2015-04-01</metadata> ⑨
        <metadata key="documento:anno">2015</metadata>
        <metadata key="documento:tipo">D01</metadata>
        <metadata key="documento:sezionale">SEZ</metadata> ⑩
        <metadata key="fattura:numero">SEZ/D33</metadata> ⑪
        <registro>urn:entaksi:IT12345678904:_default:reg:2015:D01:SEZ</registro> ⑫
        <file> ⑬
            <dc>
                <terms:title>fattura_emessa.pdf</terms:title> ⑭
                <terms:format>application/pdf</terms:format>
            </dc>
        </file>
    </fileGroup>
    <fileGroup> ⑮
        <dc>
            <terms:type>Fattura Ricevuta</terms:type> ⑯
            <terms:subject>Fattura XXX/12 del 10/05/2015 Fornitore Srl</terms:subject>
            <terms:format>analogico</terms:format>
        </dc>
        <metadata key="produttore:ragionesociale">Acme Srl</metadata> ⑰
        <metadata key="produttore:idfiscale">IT12345678904</metadata>

        <metadata key="mittente:ragionesociale">Fornitore Srl</metadata> ⑱
        <metadata key="mittente:idfiscale">IT01621900479</metadata>

        <metadata key="documento:data">2015-05-10</metadata>
        <metadata key="documento:anno">2015</metadata>
        <metadata key="documento:tipo">D02></metadata> ⑲
        <metadata key="documento:sezionale">_default</metadata> ⑳
        <metadata key="fattura:numero">XXX/12</metadata>
        <registro>urn:entaksi:IT12345678904:_default:reg:2015:D02:_default</registro>
        <file>
            <dc>
                <terms:title>fattura_ricevuta.pdf</terms:title>
                <terms:format>application/pdf</terms:format>
            </dc>
        </file>
    </fileGroup>
</pdv>
```

Metadato	Descrizione
1	Il formato F999 indica che si tratta di un pacchetto con indice. Si tratta di una informazione interna che serve per la validazione del pacchetto.
2	Il file è composto da una sequenza di una o più tag `fileGroup` ognuna delle quali rappresenta una unità documentaria (ovvero un documento) inviato in conservazione. La tag si chiama `fileGroup` perché può contenere più di un file, dove il primo file è il file principale relativo al documento vero e proprio, mentre i file successivi sono annessi o allegati al file. Nella maggior parte dei casi ci sarà un solo file, ma è possibile, ad esempio, archiviare la scansione della ricevuta di ritorno della raccomandata con cui è stata inviata la fattura. Questa scansione verrebbe archiviata come secondo file, cioè come annesso al documento e insieme ad esso costituisce un'unica unità documentaria.
3	I metadati dell'unità documentaria sono indicati mediante un gruppo di tag con namespace `terms` che si trovano all'interno della tag `dc` e una serie di tag `metadata` che hanno un attributo key e un valore. I metadati `dc` sono i metadati Dublin Core, uno standard internazionale per la classificazione dei documenti in base a criteri archivistici. I metadati `metadata` sono altri metadati relativi ai criteri gestionali.
4	Il termine Dublin Core `terms:type` contiene una descrizione comprensibile per l'utente del tipo di documento archiviato. Per le fatture emesse (come in questo caso) deve essere uguale "Fatture Emesse".
5	Il termine Dublin Core `terms:subject` contiene una descrizione del documento archiviato. Per le fatture può essere uguale a "Fattura `numero-documento` del `data-documento` `cliente`". Il contenuto è arbitrario ma è importante che contenga il numero del documento così come riportato nella fattura.
6	Il termine Dublin Core `terms:format` a livello di fileGroup ovvero a livello di unità documentaria, indica se il documento è analogico o digitale. Se il documento è analogico (cioè se è una scansione di un documento cartaceo) il processo di versamento completa la digitalizzazione a norma del documento. Il documento analogico va dichiarato inserendo questa tag con il valore "analogico" come nell'esempio. In assenza di questa tag il documento è considerato digitale. Il documento analogico deve essere un file PDF firmato digitalmente, quindi un file PDF con firma PaDES oppure un file PDF firmato nella vecchia modalità CaDES, ovvero con estensione .pdf.p7m.
7	Le tag `metadata` consentono di indicare una vasta varietà di metadati per l'archiviazione dei documenti. In questo esempio sono riportati solo quelli obbligatori a norma di legge. La prima coppia di metadati è quella che identifica il "produttore". Il produttore è colui che versa il documento nel sistema di conservazione. Nel caso delle fatture emesse il produttore coincide con l'azienda a cui è intestata la fattura (il cedente di beni o prestatore di servizi in termini fiscali), anche nel caso in cui la fattura dovesse essere stata emessa da un intermediario. Mediante la chiave `produttore:ragionesociale` va indicata la ragione sociale del produttore. Con la chiave `produttore:idfiscale` va indicato invece l'identificativo fiscale del produttore, composto dal codice paese (IT per Italia) e dalla partita IVA.
8	In modo del tutto analogo la coppia di tag `metadata` con chiavi uguali a `destinatario:ragionesociale` e `destinatario:idfiscale` indica i dati del destinatario della fattura, ovvero il cliente (in termini fiscali il cessionario di beni o il committente di servizi).

Metadato	Descrizione
9	Le tag `metadata` con la chiave che ha come prefisso `documento:` definiscono i dati che servono per identificare e posizionare il documento nell'archivio. • `documento:data` definisce la data del documento che, per le fatture emesse, è la data della fattura. • `documento:anno` deve coincidere con l'anno indicato in documento:data. • `documento:tipo` è il codice del tipo di documento. Le fatture emesse vanno archiviate con il codice D01. • `documento:sezionale` è il numeratore utilizzato per la numerazione progressiva delle fatture. Se l'azienda non usa più di un numeratore, in questo dato può essere indicato il valore `_default`. Il numeratore e il progressivo costituiscono di norma il "numero del documento" che è riportato in `terms:subject`. Il modo in cui numeratore e progressivo sono combinati in un'unica stringa è a discrezione dell'azienda ed è stato oggetto nel tempo di alcune precisazione dell'Agenzia delle Entrate. Per semplificare il processo di digitalizzazione è utile suggerire alle aziende di adottare meccanismi semplici di combinazione, ad esempio separare il numeratore e il progressivo con una sbarra (SEZ/33) o con un trattino (SEZ-33) • `documento:numero` è il progressivo della fattura estratto dal numero della fattura senza il sezionale. Questo valore deve essere numerico e univoco all'interno del sezionale. Ovvero, il sistema non accetterà l'archiviazione di due documenti con lo stesso anno, tipo, sezionale e numero
10	Il metadato `documento:sezionale` corrisponde al sezionale delle fatture emesse, non è necessario indicarlo per le fatture ricevute.
11	Il metadato `fattura:numero` è il numero della fattura, che comprende sezionale e progressivo.
12	la tag `registro` specifica il registro di archiviazione del documento che si può ricavare dai metadati precedenti secondo questa sintassi: `urn:entaksi:<id-fiscale-produttore>:<unità-organizzativa>:reg:<anno>:<tipo-documento>:<sezionale>` Se l'azienda ha più unità organizzative (il che corrisponde anche alla registrazione di più entità azienda diverse nel sistema), il codice dell'unità organizzativa va indicato nel registro. Altrimenti in questa posizione verrà indicato `_default`.
13	Una o più tag `file` consentono di dichiarare i file che devono essere archiviati nell'unità documentaria. In questo esempio c'è un solo file.
14	Ciascun file va dichiarato con due o tre metadati Dublin Core che sono i seguenti: • `terms:title` deve contenere il nome del file così com'è all'interno del file zip del pacchetto di versamento • `terms:format` deve riportare il mime type del file. Per i documenti PDF il mime type è application/pdf. Se il file è firmato in modalità CaDES e quindi ha estensione .pdf.p7m deve essere riportato anche il seguente termine: • `<terms:medium>application/pkcs7-mime</terms:medium>`. Il termine `terms:format` rimane application/pdf.
15	Nel secondo `fileGroup` dell'esempio viene dichiarata una fattura ricevuta da un fornitore. Il Pacchetto di Versamento può contenere un numero arbitrario di `fileGroup`, nei limiti in cui il file pdv.xml e il file ZIP mantengono delle dimensioni ragionevoli per essere elaborati. Mediamente non dovrebbero esserci più di 100 documenti.
16	I metadati Dublin Core della fattura ricevuta sono analoghi a quelli della fattura emessa, ma in `terms:type` è indicato "Fattura Ricevuta".

Metadato	Descrizione
17	Il produttore è sempre l'azienda che sta versando il documento in conservazione. In questo caso coincide con il destinatario della fattura (in termini fiscali il cessionario di beni o committente di servizi).
18	L'azienda da cui proviene la fattura (in termini fiscali il cedente di beni o prestatore di servizi) è indicata con mittente.
19	il tipo documento per le fatture ricevute è D02.
20	Il sezionale o serie archivistica consente di archiviare i documenti dello stesso tipo in registri diversi. In questo caso il sezionale non ha niente a che fare con il sezionale della fattura, ma può essere usato per dividere le fatture secondo un criterio scelto dal produttore. Ad esempio potrebbe essere definito un sezionale per ogni cliente, oppure per categoria commerciale. Indicando `_default` le fatture vengono archiviate tutte nello stesso registro di default. Notare che per le fatture ricevute non deve essere indicato il metadato `documento:numero`. Le fatture ricevute vengono numerate automaticamente dal sistema con un progressivo di archiviazione.
21	Il metadato `fattura:numero` è il numero della fattura, che comprende sezionale e progressivo.
22	Il registro di archiviazione si compone come nel caso delle fatture trasmesse.
23	In questo esempio la fattura in formato originario analogico è stata firmata in formato PaDES, e dunque i metadati si riferiscono a un file pdf "fattura_ricevuta.pdf", pertanto non è necessario apporre il metadato `terms:medium`.

8.3. Esempio pacchetto con documenti D15 Registro fatture acquisto

Il pacchetto contiene due documenti appartenenti alla tipologia D15, *Registro fatture acquisto*.

```
<?xml version="1.0" encoding="UTF-8"?>
<pdv xmlns="http://entaksi.eu/schemas/econ/1.0/" xmlns:terms="http://purl.org/dc/terms/">
  <formato>F999</formato> ①
  <fileGroup> ②
    <dc> ③
      <terms:type>Registro fatture acquisto</terms:type> ④
      <terms:subject>REGISTRO IVA ACQUISTI dal 01/01 al 31/12</terms:subject> ⑤
    </dc>
    <metadata key="produttore:idfiscale">IT01234567890</metadata> ⑥
    <metadata key="produttore:nome">Nome produttore</metadata>
    <metadata key="produttore:cognome">Cognome produttore</metadata>

    <metadata key="destinatario:idfiscale">IT01234567891</metadata> ⑦
    <metadata key="destinatario:nome">Nome destinatario</metadata>
    <metadata key="destinatario:cognome">Cognome destinatario</metadata>

    <metadata key="documento:anno">2015</metadata> ⑧
    <metadata key="documento:tipo">D15</metadata>
    <metadata key="documento:data">2015-12-31</metadata>
    <metadata key="documento:datainizio">2015-01-01</metadata>
    <metadata key="documento:datatermine">2015-12-31</metadata>
    <registro>urn:entaksi:IT01234567890:_default:reg:2015:D15:_default</registro> ⑨
    <file> ⑩
      <dc>
        <terms:title>d000000045_0001_0001.pdf</terms:title> ⑪
        <terms:format>application/pdf</terms:format>
      </dc>
    </file>
  </fileGroup>
  <fileGroup> ⑫
    <dc>
      <terms:type>Registro fatture acquisto</terms:type> ⑬
      <terms:subject>DOCUMENTI NON PAGATI/INCASSATI</terms:subject>
    </dc>
    <metadata key="produttore:idfiscale">IT01234567890</metadata> ⑭
    <metadata key="produttore:nome">Nome produttore</metadata>
    <metadata key="produttore:cognome">Cognome produttore</metadata>

    <metadata key="destinatario:idfiscale">IT01234567891</metadata> ⑮
    <metadata key="destinatario:nome">Nome destinatario</metadata>
    <metadata key="destinatario:cognome">Cognome destinatario</metadata>

    <metadata key="documento:anno">2015</metadata> ⑯
    <metadata key="documento:tipo">D15</metadata>
    <metadata key="documento:data">2015-12-31</metadata>
    <registro>urn:entaksi:IT01234567890:_default:reg:2015:D15:_default</registro> ⑰
    <file> ⑱
      <dc>
        <terms:title>d000000051_0001_0001.pdf</terms:title> ⑲
        <terms:format>application/pdf</terms:format>
      </dc>
    </file>
  </fileGroup>
</pdv>
```

Metadato	Descrizione
1	Il formato `F999` indica che si tratta di un pacchetto con indice. Si tratta di una informazione interna che serve per la validazione del pacchetto.
2	Il file è composto da una sequenza di una o più tag `fileGroup` ognuna delle quali rappresenta una unità documentaria (ovvero un documento) inviato in conservazione. La tag si chiama `fileGroup` perché può contenere più di un file, dove il primo file è il file principale relativo al documento vero e proprio, mentre i file successivi sono annessi o allegati al file. Nella maggior parte dei casi ci sarà un solo file, ma è possibile, ad esempio, archiviare la scansione della ricevuta di ritorno della raccomandata con cui è stata inviata la fattura. Questa scansione verrebbe archiviata come secondo file, cioè come annesso al documento e insieme ad esso costituisce un'unica unità documentaria.
3	I metadati dell'unità documentaria sono indicati mediante un gruppo di tag con namespace `terms` che si trovano all'interno della tag `dc` e una serie di tag `metadata` che hanno un attributo key e un valore. I metadati `dc` sono i metadati Dublin Core, uno standard internazionale per la classificazione dei documenti in base a criteri archivistici. I metadati `metadata` sono altri metadati relativi ai criteri gestionali.
4	Il termine Dublin Core `terms:type` contiene una descrizione comprensibile per l'utente del tipo di documento archiviato. In questo caso corrisponde a "Registro fatture acquisto".
5	Il termine Dublin Core `terms:subject` contiene una descrizione del documento archiviato. In questo caso si è indicato, dal momento che si tratta di un registro, come "Registro `tipo-registro` dal `data-inizio` al `data-termine`". Il contenuto è arbitrario ma è importante che contenga il tipo di registro.
6	Le tag `metadata` consentono di indicare una vasta varietà di metadati per l'archiviazione dei documenti. In questo esempio sono riportati solo quelli obbligatori a norma di legge. I primi tre metadati identificano il produttore, dichiarando con la chiave `produttore:idfiscale` l'identificativo fiscale (composto dal codice paese, IT per Italia, e dalla partita IVA), e con `produttore:nome` e `produttore:cognome` il nome e il cognome di colui che versa il documento nel sistema di conservazione.
7	In modo del tutto analogo i tre tag `metadata` con chiavi uguali a `destinatario:idfiscale`, `destinatario:nome` e `destinatario:cognome`, indicano i dati del destinatario della fattura, ovvero il cliente (in termini fiscali il cessionario di beni o il committente di servizi).
8	Le tag `metadata` con la chiave che ha come prefisso `documento:` definiscono i dati che servono per identificare e posizionare il documento nell'archivio. • `documento:anno` deve coincidere con l'anno indicato in documento:data. • `documento:tipo` è il codice del tipo di documento. I registri fatture acquisto hanno codice `D15`. • `documento:data` definisce la data del documento che, per le fatture emesse, è la data della fattura. • `documento:datainizio` che in presenza di documenti che coprono un arco temporale come i registri indicano la data di inizio del documento. • `documento:datatermine` sempre per i documenti che coprono un arco temporale indica la data di termine del documento.

Metadato	Descrizione
9	la tag `registro` specifica il registro di archiviazione del documento che si può ricavare dai metadati precedenti secondo questa sintassi: `urn:entaksi:<id-fiscale-produttore>:<unità-organizzativa>:reg:<anno>:<tipo-documento>:<sezionale>` Se l'azienda ha più unità organizzative (il che corrisponde anche alla registrazione di più entità azienda diverse nel sistema), il codice dell'unità organizzativa va indicato nel registro. Altrimenti in questa posizione verrà indicato `_default`.
10	Una o più tag `file` consentono di dichiarare i file che devono essere archiviati nell'unità documentaria. In questo esempio ci sono due file.
11	Ciascun file va dichiarato con due o tre metadati Dublin Core che sono i seguenti: • `terms:title` deve contenere il nome del file così com'è all'interno del file zip del pacchetto di versamento • `terms:format` deve riportare il mime type del file. Per i documenti PDF il mime type è application/pdf. Se il file è firmato in modalità CaDES e quindi ha estensione .pdf.p7m deve essere riportato anche il seguente termine: • `<terms:medium>application/pkcs7-mime</terms:medium>`. Il termine `terms:format` rimane application/pdf.
12	Nel secondo `fileGroup` dell'esempio viene dichiarato una secondo registro registro di fatture acquisto contenente la registrazione di documenti non pagati / incassati. Il Pacchetto di Versamento può contenere un numero arbitrario di `fileGroup`, nei limiti in cui il file pdv.xml e il file ZIP mantengono delle dimensioni ragionevoli per essere elaborati. Mediamente non dovrebbero esserci più di 100 documenti.
14	Come sopra è indicato come `terms:type` "Registro fatture acquisto", cambia la descrizione in "DPCUMENTI NON PAGATI/INCASSATI".
15	Il produttore è sempre l'azienda che sta versando i documenti in conservazione.
16	Destinatario del registro.
18	Il sezionale o serie archivistica consente di archiviare i documenti dello stesso tipo in registri diversi. In questo caso il sezionale non ha niente a che fare con il sezionale della fattura, ma può essere usato per dividere le fatture secondo un criterio scelto dal produttore. Ad esempio potrebbe essere definito un sezionale per ogni cliente, oppure per categoria commerciale. Indicando `_default` le fatture vengono archiviate tutte nello stesso registro di default. Notare che per le fatture ricevute non deve essere indicato il metadato `documento:numero`. Le fatture ricevute vengono numerate automaticamente dal sistema con un progressivo di archiviazione.
19	Il registro di archiviazione si compone come nel caso delle fatture trasmesse.

8.4. Esempio pacchetto D17 Registro fatture emesse

Il pacchetto contiene un documento appartenente alla tipologia D17, *Registro fatture emesse*:

```
<?xml version="1.0" encoding="UTF-8"?>
<pdv xmlns="http://entaksi.eu/schemas/econ/1.0/" xmlns:terms="http://purl.org/dc/terms/">
  <formato>F999</formato> ①
  <fileGroup> ②
    <dc> ③
      <terms:type>Registro fatture emesse</terms:type> ④
      <terms:subject>REGISTRO IVA VENDITE dal 01/01 al 31/12</terms:subject> ⑤
    </dc>
    <metadata key="produttore:idfiscale">IT01234567890</metadata> ⑥
    <metadata key="produttore:nome">Nome produttore</metadata>
    <metadata key="produttore:cognome">Cognome produttore</metadata>

    <metadata key="destinatario:idfiscale">IT01234567890</metadata> ⑦
    <metadata key="destinatario:nome">Nome destinatario</metadata>
    <metadata key="destinatario:cognome">Cognome destinatario</metadata>

    <metadata key="documento:anno">2015</metadata> ⑧
    <metadata key="documento:tipo">D17</metadata>
    <metadata key="documento:data">2015-12-31</metadata>
    <metadata key="documento:datainizio">2015-01-01</metadata>
    <metadata key="documento:datatermine">2015-12-31</metadata>
    <registro>urn:entaksi:IT01234567890:_default:reg:2015:D17:_default</registro> ⑨
    <file> ⑩
      <dc>
        <terms:title>d000000046_0001_0001.pdf</terms:title> ⑪
        <terms:format>application/pdf</terms:format>
      </dc>
    </file>
  </fileGroup>
</pdv>
```

Metadato	Descrizione
1	Il formato F999 indica che si tratta di un pacchetto con indice. Si tratta di una informazione interna che serve per la validazione del pacchetto.
2	Il file è composto da una sequenza di una o più tag `fileGroup` ognuna delle quali rappresenta una unità documentaria (ovvero un documento) inviato in conservazione. La tag si chiama `fileGroup` perché può contenere più di un file, dove il primo file è il file principale relativo al documento vero e proprio, mentre i file successivi sono annessi o allegati al file. Nella maggior parte dei casi ci sarà un solo file, ma è possibile, ad esempio, archiviare la scansione della ricevuta di ritorno della raccomandata con cui è stata inviata la fattura. Questa scansione verrebbe archiviata come secondo file, cioè come annesso al documento e insieme ad esso costituisce un'unica unità documentaria.
3	I metadati dell'unità documentaria sono indicati mediante un gruppo di tag con namespace `terms` che si trovano all'interno della tag `dc` e una serie di tag `metadata` che hanno un attributo key e un valore. I metadati `dc` sono i metadati Dublin Core, uno standard internazionale per la classificazione dei documenti in base a criteri archivistici. I metadati `metadata` sono altri metadati relativi ai criteri gestionali.
4	Il termine Dublin Core `terms:type` contiene una descrizione comprensibile per l'utente del tipo di documento archiviato. In questo caso corrisponde a "Registro fatture emesse".

Metadato	Descrizione
5	Il termine Dublin Core `terms:subject` contiene una descrizione del documento archiviato. In questo caso si è indicato, dal momento che si tratta di un registro, come "Registro `tipo-registro` dal `data-inizio` al `data-termine`". Il contenuto è arbitrario ma è importante che contenga il tipo di registro.
6	Le tag `metadata` consentono di indicare una vasta varietà di metadati per l'archiviazione dei documenti. In questo esempio sono riportati solo quelli obbligatori a norma di legge. I primi tre metadati identificano il produttore, dichiarando con la chiave `produttore:idfiscale` l'identificativo fiscale (composto dal codice paese, IT per Italia, e dalla partita IVA), e con `produttore:nome` e `produttore:cognome` il nome e il cognome di colui che versa il documento nel sistema di conservazione.
7	In modo del tutto analogo i tre tag `metadata` con chiavi uguali a `destinatario:idfiscale`, `destinatario:nome` e `destinatario:cognome`, indicano i dati del destinatario della fattura, ovvero il cliente (in termini fiscali il cessionario di beni o il committente di servizi).
8	Le tag `metadata` con la chiave che ha come prefisso `documento:` definiscono i dati che servono per identificare e posizionare il documento nell'archivio. • `documento:anno` deve coincidere con l'anno indicato in documento:data. • `documento:tipo` è il codice del tipo di documento. I registri fatture acquisto hanno codice `D15`. • `documento:data` definisce la data del documento che, per le fatture emesse, è la data della fattura. • `documento:datainizio` che in presenza di documenti che coprono un arco temporale come i registri indicano la data di inizio del documento. • `documento:datatermine` sempre per i documenti che coprono un arco temporale indica la data di termine del documento.
9	la tag `registro` specifica il registro di archiviazione del documento che si può ricavare dai metadati precedenti secondo questa sintassi: `urn:entaksi:<id-fiscale-produttore>:<unità-organizzativa>:reg:<anno>:<tipo-documento>:<sezionale>` Se l'azienda ha più unità organizzative (il che corrisponde anche alla registrazione di più entità azienda diverse nel sistema), il codice dell'unità organizzativa va indicato nel registro. Altrimenti in questa posizione verrà indicato `_default`.
10	Una o più tag `file` consentono di dichiarare i file che devono essere archiviati nell'unità documentaria. In questo esempio c'è un solo file.
11	Ciascun file va dichiarato con due o tre metadati Dublin Core che sono i seguenti: • `terms:title` deve contenere il nome del file così com'è all'interno del file zip del pacchetto di versamento • `terms:format` deve riportare il mime type del file. Per i documenti PDF il mime type è application/pdf. Se il file è firmato in modalità CaDES e quindi ha estensione .pdf.p7m deve essere riportato anche il seguente termine: • `<terms:medium>application/pkcs7-mime</terms:medium>`. Il termine `terms:format` rimane application/pdf.

8.5. Esempio pacchetto D26 Registro riepilogativo Iva

Il pacchetto contiene quattro documenti appartenenti alla tipologia D26, *Registro riepilogativo IVA*.

```
<?xml version="1.0" encoding="UTF-8"?>
<pdv xmlns="http://entaksi.eu/schemas/econ/1.0/" xmlns:terms="http://purl.org/dc/terms/">
  <formato>F999</formato> ①
  <fileGroup> ②
    <dc> ③
      <terms:type>Registro riepilogativo Iva</terms:type> ④
      <terms:subject>LIQUIDAZIONE IVA TIMESTRE 1</terms:subject> ⑤
    </dc>
    <metadata key="produttore:idfiscale">IT01234567890</metadata> ⑥
    <metadata key="produttore:ragionesociale">Ragione sociale produttore</metadata>

    <metadata key="destinatario:idfiscale">IT01234567890</metadata> ⑦
    <metadata key="destinatario:ragionesociale">Ragione sociale destinatario</metadata>

    <metadata key="documento:anno">2015</metadata> ⑧
    <metadata key="documento:tipo">D26</metadata>
    <metadata key="documento:data">2015-12-31</metadata>
    <metadata key="documento:datainizio">2015-01-01</metadata>
    <metadata key="documento:datatermine">2015-03-31</metadata>
    <registro>urn:entaksi:IT01234567890:_default:reg:2015:D26:_default</registro> ⑨
    <file> ⑩
      <dc>
        <terms:title>d000000047_0001_0001.pdf</terms:title> ⑪
        <terms:format>application/pdf</terms:format>
      </dc>
    </file>
  </fileGroup>
  <fileGroup> ⑫
    <dc>
      <terms:type>Registro riepilogativo Iva</terms:type> ⑬
      <terms:subject>LIQUIDAZIONE IVA TIMESTRE 2</terms:subject>
    </dc>
    <metadata key="produttore:idfiscale">IT01234567890</metadata> ⑭
    <metadata key="produttore:ragionesociale">Ragione sociale produttore</metadata>

    <metadata key="destinatario:idfiscale">IT01234567890</metadata> ⑮
    <metadata key="destinatario:ragionesociale">Ragione sociale destinatario</metadata>

    <metadata key="documento:anno">2015</metadata> ⑯
    <metadata key="documento:tipo">D26</metadata>
    <metadata key="documento:data">2015-12-31</metadata>
    <metadata key="documento:datainizio">2015-04-01</metadata>
    <metadata key="documento:datatermine">2015-06-30</metadata>
    <registro>urn:entaksi:IT01234567890:_default:reg:2015:D26:_default</registro> ⑰
    <file> ⑱
      <dc>
        <terms:title>d000000048_0001_0001.pdf</terms:title> ⑲
        <terms:format>application/pdf</terms:format>
      </dc>
    </file>
  </fileGroup>
  <fileGroup>
    <dc>
      <terms:type>Registro riepilogativo Iva</terms:type> ⑳
      <terms:subject>LIQUIDAZIONE IVA TIMESTRE 3</terms:subject>
    </dc>
    <metadata key="produttore:idfiscale">IT01234567890</metadata>
    <metadata key="produttore:ragionesociale">Ragione sociale produttore</metadata>

    <metadata key="destinatario:idfiscale">IT01234567890</metadata>
```

```
    <metadata key="destinatario:ragionesociale">Ragione sociale destinatario</metadata>

    <metadata key="documento:anno">2015</metadata>
    <metadata key="documento:tipo">D26</metadata>
    <metadata key="documento:data">2015-12-31</metadata>
    <metadata key="documento:datainizio">2015-07-01</metadata>
    <metadata key="documento:datatermine">2015-09-30</metadata>
    <registro>urn:entaksi:IT01234567890:_default:reg:2015:D26:_default</registro>
    <file>
      <dc>
        <terms:title>d000000049_0001_0001.pdf</terms:title>
        <terms:format>application/pdf</terms:format>
      </dc>
    </file>
  </fileGroup>
  <fileGroup>
    <dc>
      <terms:type>Registro riepilogativo Iva</terms:type>
      <terms:subject>LIQUIDAZIONE IVA TIMESTRE 4</terms:subject>
    </dc>
    <metadata key="produttore:idfiscale">IT01234567890</metadata>
    <metadata key="produttore:ragionesociale">Ragione sociale produttore</metadata>
    <metadata key="destinatario:idfiscale">IT01234567890</metadata>
    <metadata key="destinatario:ragionesociale">Ragione sociale destinatario</metadata>
    <metadata key="documento:anno">2015</metadata>
    <metadata key="documento:tipo">D26</metadata>
    <metadata key="documento:data">2015-12-31</metadata>
    <metadata key="documento:datainizio">2015-10-01</metadata>
    <metadata key="documento:datatermine">2015-12-31</metadata>
    <registro>urn:entaksi:IT01234567890:_default:reg:2015:D26:_default</registro>
    <file>
      <dc>
        <terms:title>d000000050_0001_0001.pdf</terms:title>
        <terms:format>application/pdf</terms:format>
      </dc>
    </file>
  </fileGroup>
</pdv>
```

Metadato	Descrizione
1	Il formato F999 indica che si tratta di un pacchetto con indice. Si tratta di una informazione interna che serve per la validazione del pacchetto.
2	Il file è composto da una sequenza di una o più tag `fileGroup` ognuna delle quali rappresenta una unità documentaria (ovvero un documento) inviato in conservazione. La tag si chiama `fileGroup` perché può contenere più di un file, dove il primo file è il file principale relativo al documento vero e proprio, mentre i file successivi sono annessi o allegati al file. Nella maggior parte dei casi ci sarà un solo file, ma è possibile, ad esempio, archiviare la scansione della ricevuta di ritorno della raccomandata con cui è stata inviata la fattura. Questa scansione verrebbe archiviata come secondo file, cioè come annesso al documento e insieme ad esso costituisce un'unica unità documentaria.
3	I metadati dell'unità documentaria sono indicati mediante un gruppo di tag con namespace `terms` che si trovano all'interno della tag `dc` e una serie di tag `metadata` che hanno un attributo key e un valore. I metadati `dc` sono i metadati Dublin Core, uno standard internazionale per la classificazione dei documenti in base a criteri archivistici. I metadati `metadata` sono altri metadati relativi ai criteri gestionali.
4	Il termine Dublin Core `terms:type` contiene una descrizione comprensibile per l'utente del tipo di documento archiviato. In questo caso corrisponde a "Registro riepilogativo IVA".

Metadato	Descrizione
5	Il termine Dublin Core `terms:subject` contiene una descrizione del documento archiviato. In questo caso si è indicato semplicemente "LIQUIDAZIONE IVA TIMESTRE 1". Il pacchetto è composto da 5 file, con numerazione crescente nella descrizione.
6	Le tag `metadata` consentono di indicare una vasta varietà di metadati per l'archiviazione dei documenti. In questo esempio sono riportati solo quelli obbligatori a norma di legge. I primi tre metadati identificano il produttore, dichiarando con la chiave `produttore:idfiscale` l'identificativo fiscale (composto dal codice paese, IT per Italia, e dalla partita IVA), e con `produttore:nome` e `produttore:cognome` il nome e il cognome di colui che versa il documento nel sistema di conservazione.
7	In modo del tutto analogo i tre tag `metadata` con chiavi uguali a `destinatario:idfiscale`, `destinatario:nome` e `destinatario:cognome`, indicano i dati del destinatario della fattura, ovvero il cliente (in termini fiscali il cessionario di beni o il committente di servizi).
8	Le tag `metadata` con la chiave che ha come prefisso `documento:` definiscono i dati che servono per identificare e posizionare il documento nell'archivio. • `documento:anno` deve coincidere con l'anno indicato in documento:data. • `documento:tipo` è il codice del tipo di documento. I registri fatture acquisto hanno codice D15. • `documento:data` definisce la data del documento che, per le fatture emesse, è la data della fattura. • `documento:datainizio` che in presenza di documenti che coprono un arco temporale come i registri indicano la data di inizio del documento. • `documento:datatermine` sempre per i documenti che coprono un arco temporale indica la data di termine del documento.
9	la tag `registro` specifica il registro di archiviazione del documento che si può ricavare dai metadati precedenti secondo questa sintassi: `urn:entaksi:<id-fiscale-produttore>:<unità-organizzativa>:reg:<anno>:<tipo-documento>:<sezionale>` Se l'azienda ha più unità organizzative (il che corrisponde anche alla registrazione di più entità azienda diverse nel sistema), il codice dell'unità organizzativa va indicato nel registro. Altrimenti in questa posizione verrà indicato `_default`.
10	Una o più tag `file` consentono di dichiarare i file che devono essere archiviati nell'unità documentaria. In questo esempio ci sono cinque file.
11	Ciascun file va dichiarato con due o tre metadati Dublin Core che sono i seguenti: • `terms:title` deve contenere il nome del file così com'è all'interno del file zip del pacchetto di versamento • `terms:format` deve riportare il mime type del file. Per i documenti PDF il mime type è application/pdf. Se il file è firmato in modalità CaDES e quindi ha estensione .pdf.p7m deve essere riportato anche il seguente termine: • `<terms:medium>application/pkcs7-mime</terms:medium>`. Il termine `terms:format` rimane application/pdf.
12	Nel secondo `fileGroup` dell'esempio viene dichiarato un secondo registro riepilogativo IVA. Il Pacchetto di Versamento può contenere un numero arbitrario di `fileGroup`, nei limiti in cui il file pdv.xml e il file ZIP mantengono delle dimensioni ragionevoli per essere elaborati. Mediamente non dovrebbero esserci più di 100 documenti.

Metadato	Descrizione
14	Come sopra è indicato come `terms:type` "Registro riepilogativo IVA".
15	Il produttore è sempre l'azienda che sta versando i documenti in conservazione.
16	Destinatario del registro.
18	Il sezionale o serie archivistica consente di archiviare i documenti dello stesso tipo in registri diversi. In questo caso il sezionale non ha niente a che fare con il sezionale della fattura, ma può essere usato per dividere le fatture secondo un criterio scelto dal produttore. Ad esempio potrebbe essere definito un sezionale per ogni cliente, oppure per categoria commerciale. Indicando `_default` le fatture vengono archiviate tutte nello stesso registro di default. Notare che per le fatture ricevute non deve essere indicato il metadato `documento:numero`. Le fatture ricevute vengono numerate automaticamente dal sistema con un progressivo di archiviazione.
19	Il registro di archiviazione si compone come nel caso delle fatture trasmesse.

8.6. Esempio pacchetto D58 Altri registri

Il pacchetto contiene un documento appartenente alla tipologia D58, *Altri registri*. Questa tipologia raggruppa altre tipologie di registro non censite (in questo caso un libro unico del lavoro). I metadati per questa tipologia, esclusi quelli obbligatori, sono concordati con il produttore.

```
<?xml version="1.0" encoding="UTF-8"?>
<pdv xmlns="http://entaksi.eu/schemas/econ/1.0/" xmlns:terms="http://purl.org/dc/terms/">
  <formato>F999</formato> ①
  <fileGroup> ②
    <dc> ③
      <terms:type>Altri registri</terms:type> ④
      <terms:subject>libro unico del lavoro agosto 2016</terms:subject> ⑤
    </dc>
    <metadata key="produttore:idfiscale">IT01234567890</metadata> ⑥
    <metadata key="produttore:ragionesociale">Ragione sociale produttore</metadata>
    <metadata key="destinatario:idfiscale">IT01234567891</metadata> ⑦
    <metadata key="destinatario:ragionesociale">Ragione sociale destinatario</metadata>
    <metadata key="documento:anno">2016</metadata> ⑧
    <metadata key="documento:tipo">D58</metadata>
    <metadata key="documento:data">2016-09-26</metadata>
    <registro>urn:entaksi:IT01234567890:_default:reg:2016:D58:_default</registro> ⑨
    <file> ⑩
      <dc> ⑪
        <terms:title>d000000011_0001_0001.pdf</terms:title>
        <terms:format>application/pdf</terms:format>
      </dc>
    </file>
  </fileGroup>
</pdv>
```

Metadato	Descrizione
1	Il formato F999 indica che si tratta di un pacchetto con indice. Si tratta di una informazione interna che serve per la validazione del pacchetto.
2	Il file è composto da una sequenza di una o più tag `fileGroup` ognuna delle quali rappresenta una unità documentaria (ovvero un documento) inviato in conservazione. La tag si chiama `fileGroup` perché può contenere più di un file, dove il primo file è il file principale relativo al documento vero e proprio, mentre i file successivi sono annessi o allegati al file. Nella maggior parte dei casi ci sarà un solo file, ma è possibile, ad esempio, archiviare la scansione della ricevuta di ritorno della raccomandata con cui è stata inviata la fattura. Questa scansione verrebbe archiviata come secondo file, cioè come annesso al documento e insieme ad esso costituisce un'unica unità documentaria.
3	I metadati dell'unità documentaria sono indicati mediante un gruppo di tag con namespace `terms` che si trovano all'interno della tag `dc` e una serie di tag `metadata` che hanno un attributo key e un valore. I metadati `dc` sono i metadati Dublin Core, uno standard internazionale per la classificazione dei documenti in base a criteri archivistici. I metadati `metadata` sono altri metadati relativi ai criteri gestionali.
4	Il termine Dublin Core `terms:type` contiene una descrizione comprensibile per l'utente del tipo di documento archiviato. In questo caso corrisponde a "Altri registri".
5	Il termine Dublin Core `terms:subject` contiene una descrizione del documento archiviato. In questo caso è indicato "libro unico del lavoro agosto 2016", una tipologia di registro non censita altrimenti.

Metadato	Descrizione
6	Le tag `metadata` consentono di indicare una vasta varietà di metadati per l'archiviazione dei documenti. In questo esempio sono riportati solo quelli obbligatori a norma di legge. I primi tre metadati identificano il produttore, dichiarando con la chiave `produttore:idfiscale` l'identificativo fiscale (composto dal codice paese, IT per Italia, e dalla partita IVA), e con `produttore:nome` e `produttore:cognome` il nome e il cognome di colui che versa il documento nel sistema di conservazione. Nel caso il produttore sia una persona giuridica, come in questo caso può essere indicato `produttore:ragionesociale`.
7	In modo del tutto analogo i tre tag `metadata` con chiavi uguali a `destinatario:idfiscale`, `destinatario:nome` e `destinatario:cognome`, indicano i dati del destinatario della fattura, ovvero il cliente (in termini fiscali il cessionario di beni o il committente di servizi). Nel caso il destinatario sia una persona giuridica, come in questo caso può essere indicato `destinatario:ragionesociale`.
8	Le tag `metadata` con la chiave che ha come prefisso `documento:` definiscono i dati che servono per identificare e posizionare il documento nell'archivio. • `documento:anno` deve coincidere con l'anno indicato in documento:data. • `documento:tipo` è il codice del tipo di documento. I registri fatture acquisto hanno codice D15. • `documento:data` definisce la data del documento che, per le fatture emesse, è la data della fattura. • `documento:datainizio` che in presenza di documenti che coprono un arco temporale come i registri indicano la data di inizio del documento. • `documento:datatermine` sempre per i documenti che coprono un arco temporale indica la data di termine del documento.
9	la tag `registro` specifica il registro di archiviazione del documento che si può ricavare dai metadati precedenti secondo questa sintassi: `urn:entaksi:<id-fiscale-produttore>:<unità-organizzativa>:reg:<anno>:<tipo-documento>:<sezionale>` Se l'azienda ha più unità organizzative (il che corrisponde anche alla registrazione di più entità azienda diverse nel sistema), il codice dell'unità organizzativa va indicato nel registro. Altrimenti in questa posizione verrà indicato `_default`.
10	Una o più tag `file` consentono di dichiarare i file che devono essere archiviati nell'unità documentaria. In questo esempio c'è un solo file.
11	Ciascun file va dichiarato con due o tre metadati Dublin Core che sono i seguenti: • `terms:title` deve contenere il nome del file così com'è all'interno del file zip del pacchetto di versamento • `terms:format` deve riportare il mime type del file. Per i documenti PDF il mime type è application/pdf. Se il file è firmato in modalità CaDES e quindi ha estensione .pdf.p7m deve essere riportato anche il seguente termine: • `<terms:medium>application/pkcs7-mime</terms:medium>`. Il termine `terms:format` rimane application/pdf.

8.7. Esempio pacchetto D85 Altri documenti

Il pacchetto contiene un documento appartenente alla tipologia D85, *Altri documenti*. Questa tipologia raggruppa altre tipologie di documento non censite (in questo caso un bilancio di verifica). I metadati per questa tipologia, esclusi quelli obbligatori, sono concordati con il produttore.

```
<?xml version="1.0" encoding="UTF-8"?>
<pdv xmlns="http://entaksi.eu/schemas/econ/1.0/" xmlns:terms="http://purl.org/dc/terms/">
  <formato>F997</formato> ①
    <fileGroup> ②
      <dc> ③
        <terms:type>Altri documenti</terms:type> ④
        <terms:subject>Bilancio di verifica dal 01/01/15 al 31/12/15</terms:subject> ⑤
      </dc>
      <metadata key="produttore:idfiscale">IT01234567890</metadata> ⑥
      <metadata key="produttore:nome">Nome produttore</metadata>
      <metadata key="produttore:cognome">Cognome produttore</metadata>

      <metadata key="destinatario:idfiscale">IT01234567891</metadata> ⑦
      <metadata key="destinatario:nome">Nome destinatario</metadata>
      <metadata key="destinatario:cognome">Cognome destinatario</metadata>

      <metadata key="documento:anno">2015</metadata> ⑧
      <metadata key="documento:tipo">D85</metadata>
      <metadata key="documento:data">2015-12-31</metadata>
      <registro>urn:entaksi:IT01234567890:_default:reg:2015:D85:_default</registro> ⑨
      <terms:format>analogico</terms:format> ⑩
      <file> ⑪
        <dc>
          <terms:title>d000000052_0001_0001.pdf</terms:title> ⑫
          <terms:format>application/pdf</terms:format>
        </dc>
      </file>
    </fileGroup>
</pdv>
```

Metadato	Descrizione
1	Il formato F997 indica che si tratta di un pacchetto copia digitale di documento analogico. Il documento è dichiarato analogico (v. punto 10), pertanto il sistema provvede alla conversione in PDF/A e alla successiva validazione con firma digitale in presenza di delega per il Responsabile della Conservazione. Si tratta di una informazione interna che serve per la validazione del pacchetto.
2	Il file è composto da una sequenza di una o più tag `fileGroup` ognuna delle quali rappresenta una unità documentaria (ovvero un documento) inviato in conservazione. La tag si chiama `fileGroup` perché può contenere più di un file, dove il primo file è il file principale relativo al documento vero e proprio, mentre i file successivi sono annessi o allegati al file. Nella maggior parte dei casi ci sarà un solo file, ma è possibile, ad esempio, archiviare la scansione della ricevuta di ritorno della raccomandata con cui è stata inviata la fattura. Questa scansione verrebbe archiviata come secondo file, cioè come annesso al documento e insieme ad esso costituisce un'unica unità documentaria.
3	I metadati dell'unità documentaria sono indicati mediante un gruppo di tag con namespace `terms` che si trovano all'interno della tag `dc` e una serie di tag `metadata` che hanno un attributo key e un valore. I metadati `dc` sono i metadati Dublin Core, uno standard internazionale per la classificazione dei documenti in base a criteri archivistici. I metadati `metadata` sono altri metadati relativi ai criteri gestionali.

Metadato	Descrizione
4	Il termine Dublin Core `terms:type` contiene una descrizione comprensibile per l'utente del tipo di documento archiviato. In questo caso corrisponde a "Altri documenti".
5	Il termine Dublin Core `terms:subject` contiene una descrizione del documento archiviato. In questo caso è indicato "Bilancio di verifica dal 01/01/15 al 31/12/15", una tipologia di registro non censita altrimenti.
6	Le tag `metadata` consentono di indicare una vasta varietà di metadati per l'archiviazione dei documenti. In questo esempio sono riportati solo quelli obbligatori a norma di legge. I primi tre metadati identificano il produttore, dichiarando con la chiave `produttore:idfiscale` l'identificativo fiscale (composto dal codice paese, IT per Italia, e dalla partita IVA), e con `produttore:nome` e `produttore:cognome` il nome e il cognome di colui che versa il documento nel sistema di conservazione.
7	In modo del tutto analogo i tre tag `metadata` con chiavi uguali a `destinatario:idfiscale`, `destinatario:nome` e `destinatario:cognome`, indicano i dati del destinatario della fattura, ovvero il cliente (in termini fiscali il cessionario di beni o il committente di servizi).
8	Le tag `metadata` con la chiave che ha come prefisso `documento:` definiscono i dati che servono per identificare e posizionare il documento nell'archivio. • `documento:anno` deve coincidere con l'anno indicato in documento:data. • `documento:tipo` è il codice del tipo di documento. I registri fatture acquisto hanno codice D15. • `documento:data` definisce la data del documento che, per le fatture emesse, è la data della fattura. • `documento:datainizio` che in presenza di documenti che coprono un arco temporale come i registri indicano la data di inizio del documento. • `documento:datatermine` sempre per i documenti che coprono un arco temporale indica la data di termine del documento.
9	la tag `registro` specifica il registro di archiviazione del documento che si può ricavare dai metadati precedenti secondo questa sintassi: `urn:entaksi:<id-fiscale-produttore>:<unità-organizzativa>:reg:<anno>:<tipo-documento>:<sezionale>` Se l'azienda ha più unità organizzative (il che corrisponde anche alla registrazione di più entità azienda diverse nel sistema), il codice dell'unità organizzativa va indicato nel registro. Altrimenti in questa posizione verrà indicato `_default`.
10	Il metadato a livello di fileGroup `terms:format` con valore "analogico" indica che il documento trattato era precedentemente analogico, ed è stata effettuata la sua conversione in PDF/A ed è stata apposta la firma digitale in fase di creazione del pacchetto.
11	Una o più tag `file` consentono di dichiarare i file che devono essere archiviati nell'unità documentaria. In questo esempio c'è un solo file.
12	Ciascun file va dichiarato con due o tre metadati Dublin Core che sono i seguenti: • `terms:title` deve contenere il nome del file così com'è all'interno del file zip del pacchetto di versamento • `terms:format` deve riportare il mime type del file. Per i documenti PDF il mime type è application/pdf.

8.8. Esempio pacchetto D58 Cedolini

Con il prefisso D58 possono essere individuate delle tipologie documentali particolari, non censite in altre categorie. In questo caso la tipologia comprende i cedolini.

```
<?xml version="1.0" encoding="UTF-8"?>
<pdv xmlns="http://entaksi.eu/schemas/econ/1.0/" xmlns:terms="http://purl.org/dc/terms/"> ①
  <dataVersamento>2017-12-18T15:15:15+02:00</dataVersamento> ②
  <formato>F999</formato> ③
  <fileGroup> ④
    <dc> ⑤
      <terms:type>Cedolino</terms:type> ⑥
      <terms:subject>Cedolino ottobre 2017 Mario Rossi</terms:subject> ⑦
    </dc>

  <metadata key="produttore:idfiscale">IT01234567890</metadata> ⑧
     <metadata key="produttore:nome">Nome produttore</metadata>

    <metadata key="destinatario:codicefiscale">RSSMRA87N70L113F</metadata> ⑨
    <metadata key="destinatario:nome">Nome destinatario</metadata>

    <metadata key="documento:anno">2017</metadata> ⑩
    <metadata key="documento:tipo">D58</metadata>
    <metadata key="documento:sezionale">lul</metadata>
    <metadata key="documento:data">2017-10-27</metadata>
    <metadata key="documento:datainizio">2017-10-01</metadata>
    <metadata key="documento:datatermine">2017-10-31</metadata>

    <metadata key="_extra:tipologia">Dipendente</metadata> ⑪
    <metadata key="_extra:ccosto">1204040</metadata>
    <metadata key="_extra:matricola">6160056</metadata>
    <metadata key="_extra:livello">2</metadata>
    <metadata key="_extra:qualifica">6</metadata>
    <metadata key="_extra:stabilimento">11</metadata>

    <registro>urn:entaksi:IT01234567891:_default:reg:2017:D58:lul</registro> ⑫

    <file>
      <dc>
        <terms:title>DATA/201710/IT01234567891.PDF</terms:title> ⑬
        <terms:format>application/pdf</terms:format> ⑭
      </dc>
    </file>
  </fileGroup>
  <fileGroup>
    ...
  </fileGroup>
</pdv>
```

Metadato	Descrizione
1	Dichiarazioni dei namespace XML come indicato nel manuale
2	La data del versamento (opzionale, viene impostata dal sistema durante la validazione e riportata nel rapporto di versamento)
3	Formato del pacchetto di versamento. Impostare a F999.
4	Gruppo di file relativo al documento “Cedolino”. Il gruppo è composto da un solo file che è il PDF del cedolino
5	Gruppo di metadati Dublin Core relativo al documento Cedolino

Metadato	Descrizione
6	Impostare terms:type a Cedolino
7	Contenuto del documento. Per i cedolini impostare a Cedolino <mese> <anno> <nome dipendente>. Se si tratta di una tredicesima, quattordicesima o altro indicare “tredicesima”, “quattordicesima”, ecc al posto di “mese”.
8	Il soggetto produttore è l’azienda datore di lavoro indicata con identificativo fiscale e ragione sociale.
9	Il soggetto destinatario è il dipendente indicato con codice fiscale e nome. Se il nome e cognome sono disponibili separatamente indicare il nome in destinatario:nome e il cognome in destinatario:cognome, altrimenti tutti insieme in destinatario:nome.
10	Dati del documento. Indicare l’anno di emissione del cedolino (cioè quello relativo alla mensilità) in documento:anno. `documento:tipo` è sempre uguale a `D58`, `documento:sezionale` è sempre uguale a `lul`, `documento:data` contiene la data di emissione del cedolino in formato yyyy-MM-dd. Il periodo va rappresentato mediante i due valori documento:datainizio e documento:datatermine che riportano, rispettivamente il primo e l’ultimo giorno del mese a cui si riferisce il cedolino.
11	Informazioni addizionali per la pubblicazione.
12	Registro di archiviazione. È un URI composto dalle seguenti parti tutte già descritte nelle tag precedenti: `urn:entaksi:<idfiscale-produttore>:<unità organizzativa>:reg:<anno>:<tipo documento>:<sezionale>`.
13	Nome del file (relativo alla posizione del file pdv.xml).
14	Mime type del file (application/pdf per i file PDF).

8.9. Esempio pacchetto D58 Cartellino presenze

Con il prefisso D58 possono essere individuate delle tipologie documentali particolari, non censite in altre categorie. In questo caso la tipologia comprende i cartellini presenze.

```
<fileGroup> ①
  <dc>
    <terms:type>Cartellino presenze</terms:type> ②
    <terms:subject>Cartellino presenze ottobre 2017 Mario Rossi</terms:subject> ③
  </dc>
  <metadata key="produttore:idfiscale">IT01234567891</metadata> ④
  <metadata key="produttore:ragionesociale">Nome produttore</metadata>

  <metadata key="destinatario:codicefiscale">RSSMRA87N70L113F</metadata> ⑨
  <metadata key="destinatario:nome">Nome destinatario</metadata>

  <metadata key="documento:anno">2017</metadata> ⑥
  <metadata key="documento:tipo">D58</metadata>
  <metadata key="documento:sezionale">lul</metadata>
  <metadata key="documento:data">2017-10-27</metadata>
  <metadata key="documento:datainizio">2017-10-01</metadata>
  <metadata key="documento:datatermine">2017-10-31</metadata>

  <metadata key="_extra:tipologia">Dipendente</metadata> ⑦
  <metadata key="_extra:ccosto">1204040</metadata>
  <metadata key="_extra:matricola">6160056</metadata>
  <metadata key="_extra:livello">2</metadata>
  <metadata key="_extra:qualifica">6</metadata>
  <metadata key="_extra:stabilimento">11</metadata>

  <registro>urn:entaksi:IT01234567891:_default:reg:2017:D58:lul</registro> ⑧

  <file>
    <dc>
      <terms:title>DATA/201710PR/IT01234567891.PDF</terms:title> ⑨
      <terms:format>application/pdf</terms:format>
    </dc>
  </file>
</fileGroup>
```

Metadato	Descrizione
1	Gruppo di file relativo al documento “Cartellino presenze”. Il gruppo è composto da un solo file che è il PDF del cartellino presenze.
2	Impostare `terms:type` a `Cartellino presenze`.
3	Contenuto del documento. Per i cedolini impostare a `Cartellino presenze <mese> <anno> <nome dipendente>`.
4	Il soggetto produttore è l’azienda datore di lavoro indicata con identificativo fiscale e ragione sociale.
5	Il soggetto destinatario è il dipendente indicato con codice fiscale e nome. Se il nome e cognome sono disponibili separatamente indicare il nome in `destinatario:nome` e il cognome in `destinatario:cognome`, altrimenti tutti insieme in `destinatario:nome`.

Metadato	Descrizione
6	Dati del documento. Indicare l'anno di emissione del cartellino presenze (cioè quello relativo alla mensilità) in `documento:anno`. `documento:tipo` è sempre uguale a `D58`, `documento:sezionale` è sempre uguale a `lul`, `documento:data` contiene la data di emissione del cedolino in formato yyyy-MM-dd. Il periodo va rappresentato mediante i due valori `documento:datainizio` e `documento:datatermine` che riportano, rispettivamente il primo e l'ultimo giorno del mese a cui si riferisce il cartellino presenze.
7	Informazioni addizionali per la pubblicazione.
8	Registro di archiviazione. È un URI composto dalle seguenti parti tutte già descritte nelle tag precedenti: `urn:entaksi:<idfiscale-produttore>:<unità organizzativa>:reg:<anno>:<tipo documento>:<sezionale>`.
9	Nome del file (relativo alla posizione del file pdv.xml)

8.10. Esempio pacchetto D58 Elenco dipendenti

Con il prefisso D58 possono essere individuate delle tipologie documentali particolari, non censite in altre categorie. In questo caso la tipologia comprende l'elenco dipendenti.

```
<fileGroup> ①
  <dc>
    <terms:type>Elenco dipendenti</terms:type> ②
    <terms:subject>Elenco dipendenti ottobre 2017</terms:subject> ③
  </dc>
  <metadata key="produttore:idfiscale">IT01234567891</metadata> ④
  <metadata key="produttore:ragionesociale">Ragione sociale produttore</metadata>

  <metadata key="destinatario:idfiscale">IT01234567891</metadata> ⑤
  <metadata key="destinatario:ragionesociale">Ragione sociale produttore</metadata>

  <metadata key="documento:anno">2017</metadata> ⑥
  <metadata key="documento:tipo">D58</metadata>
  <metadata key="documento:sezionale">lul</metadata>
  <metadata key="documento:data">2017-10-27</metadata>
  <metadata key="documento:datainizio">2017-10-01</metadata>
  <metadata key="documento:datatermine">2017-10-31</metadata>

  <registro>urn:entaksi:IT01234567891:_default:reg:2017:D58:lul</registro> ⑦

  <file>
    <dc>
      <terms:title>DATA/STAMPE/Elenco_Dip_201710_INF.PDF</terms:title> ⑧
      <terms:format>application/pdf</terms:format>
    </dc>
  </file>
</fileGroup>
```

Metadato	Descrizione
1	Gruppo di file relativo al documento "Elenco dipendenti". Il gruppo è composto da un solo file che è il PDF contenente l'elenco dei dipendenti.
2	Impostare `terms:type` a `Elenco dipendenti`.
3	Contenuto del documento. Per i cedolini impostare a `Elenco dipendenti <mese> <anno>`.
4	Il soggetto produttore è l'azienda datore di lavoro indicata con identificativo fiscale e ragione sociale.
5	Utilizzare come soggetto destinatario le stesse informazioni del soggetto produttore.
6	Dati del documento. Indicare l'anno di emissione del cartellino presenze (cioè quello relativo alla mensilità) in `documento:anno`. `documento:tipo` è sempre uguale a D58, `documento:sezionale` è sempre uguale a `lul`, documento:data contiene la data di emissione del cedolino in formato yyyy-MM-dd. Il periodo va rappresentato mediante i due valori `documento:datainizio` e `documento:datatermine` che riportano, rispettivamente il primo e l'ultimo giorno del mese a cui si riferisce l'elenco dipendenti.
7	Registro di archiviazione. È un URI composto dalle seguenti parti tutte già descritte nelle tag precedenti: `urn:entaksi:<idfiscale-produttore>:<unità organizzativa>:reg:<anno>:<tipo documento>:<sezionale>`.

Metadato	Descrizione
8	Nome del file (relativo alla posizione del file pdv.xml).

8.11. Esempio pacchetto D58 Riepilogo LUL

Con il prefisso D58 possono essere individuate delle tipologie documentali particolari, non censite in altre categorie. In questo caso la tipologia comprende il riepilogo LUL.

```
<fileGroup> ①
  <dc>
    <terms:type>Riepilogo LUL</terms:type> ②
    <terms:subject>Riepilogo LUL ottobre 2017</terms:subject> ③
  </dc>
  <metadata key="produttore:idfiscale">IT01234567891</metadata> ④
  <metadata key="produttore:ragionesociale">Ragione sociale produttore</metadata>

  <metadata key="destinatario:idfiscale">IT01234567891</metadata> ⑤
  <metadata key="destinatario:ragionesociale">Ragione sociale produttore</metadata>

  <metadata key="documento:anno">2017</metadata> ⑥
  <metadata key="documento:tipo">D58</metadata>
  <metadata key="documento:sezionale">lul</metadata>
  <metadata key="documento:data">2017-10-27</metadata>
  <metadata key="documento:datainizio">2017-10-01</metadata>
  <metadata key="documento:datatermine">2017-10-31</metadata>

  <registro>urn:entaksi:IT01234567891:_default:reg:2017:D58:lul</registro> ⑦

  <file>
    <dc>
      <terms:title>DATA/STAMPE/DatiLul_201710_INF.PDF</terms:title> ⑧
      <terms:format>application/pdf</terms:format>
    </dc>
  </file>
</fileGroup>
```

Metadato	Descrizione
1	Gruppo di file relativo al documento “Riepilogo LUL”. Il gruppo è composto da un solo file che è il PDF contenente il riepilogo del LUL.
2	Impostare `terms:type` a `Riepilogo LUL`.
3	Contenuto del documento. Per i cedolini impostare a `Riepilogo LUL <mese> <anno>`.
4	Il soggetto produttore è l’azienda datore di lavoro indicata con identificativo fiscale e ragione sociale
5	Utilizzare come soggetto destinatario le stesse informazioni del soggetto produttore.
6	Dati del documento. Indicare l’anno di emissione del cartellino presenze (cioè quello relativo alla mensilità) in `documento:anno`. `documento:tipo` è sempre uguale a D58, `documento:sezionale` è sempre uguale a `lul`, `documento:data` contiene la data di emissione del cedolino in formato yyyy-MM-dd. Il periodo va rappresentato mediante i due valori `documento:datainizio` e `documento:datatermine` che riportano, rispettivamente il primo e l’ultimo giorno del mese a cui si riferisce l’elenco dipendenti.

8.12. Esempio pacchetto D58 Indice LUL

Con il prefisso D58 possono essere individuate delle tipologie documentali particolari, non censite in altre categorie. In questo caso la tipologia comprende l'indice LUL.

```
<fileGroup> ①
  <dc>
    <terms:type>Indice LUL</terms:type> ②
    <terms:subject>Indice LUL ottobre 2017</terms:subject> ③
    <terms:date>2017-11-22T15:15:15+02:00</terms:date> ④
  </dc>
  <metadata key="produttore:idfiscale">IT01234567891</metadata> ⑤
  <metadata key="produttore:ragionesociale">Ragione sociale produttore</metadata>

  <metadata key="destinatario:idfiscale">IT01234567891</metadata> ⑥
  <metadata key="destinatario:ragionesociale">Ragione sociale produttore</metadata>

  <metadata key="documento:anno">2017</metadata> ⑦
  <metadata key="documento:tipo">D58</metadata>
  <metadata key="documento:sezionale">lul</metadata>
  <metadata key="documento:data">2017-10-27</metadata>
  <metadata key="documento:datainizio">2017-10-01</metadata>
  <metadata key="documento:datatermine">2017-10-31</metadata>

  <registro>urn:entaksi:IT01234567891:_default:reg:2017:D58:lul</registro> ⑧

  <file>
    <dc>
      <terms:title>DATA/MARCHE/Sincro_Indice_201710_Div1_INF.xml.p7m.m7m</terms:title> ⑨
      <terms:medium>application/pkcs7-mime</terms:medium> ⑩
      <terms:format>text/xml</terms:format> ⑪
    </dc>
  </file>
  <file>
    <dc>
      <terms:title>DATA/MARCHE/Sincro_Indice_201710_Div1_INF.xml.p7m</terms:title> ⑫
      <terms:medium>application/pkcs7-mime</terms:medium>
      <terms:format>text/xml</terms:format>
    </dc>
  </file>
  <file>
    <dc>
      <terms:title>DATA/MARCHE/Sincro_Indice_201710_Div1_INF.xml</terms:title> ⑬
      <terms:format>text/xml</terms:format>
    </dc>
  </file>
</fileGroup>
```

Metadato	Descrizione
1	Gruppo di file relativo al documento "Indice LUL". Il gruppo è composto da tre file, il primo dei quali è il file con firma e marca digitale (p7m.m7m). Gli altri due (opzionali) sono il file XML prima della firma e il file PDF contenente il riepilogo delle firme. I file successivi al primo sono archiviati come annessi.
2	Impostare `terms:type` a `Indice LUL`.
3	Contenuto del documento. Per i cedolini impostare a `Indice LUL <mese> <anno>`.
4	Se disponibile, indicare la data della marca temporale in `terms:date` (opzionale).

Metadato	Descrizione
5	Il soggetto produttore è l'azienda datore di lavoro indicata con identificativo fiscale e ragione sociale
6	Utilizzare come soggetto destinatario le stesse informazioni del soggetto produttore.
7	Dati del documento. Indicare l'anno di emissione del cartellino presenze (cioè quello relativo alla mensilità) in `documento:anno`. documento:tipo è sempre uguale a `D58`, `documento:sezionale` è sempre uguale a `lul`, `documento:data` contiene la data di emissione del cedolino in formato yyyy-MM-dd. Il periodo va rappresentato mediante i due valori `documento:datainizio` e `documento:datatermine` che riportano, rispettivamente il primo e l'ultimo giorno del mese a cui si riferisce l'elenco dipendenti.
8	Registro di archiviazione. È un URI composto dalle seguenti parti tutte già descritte nelle tag precedenti: `urn:entaksi:<idfiscale-produttore>:<unità organizzativa>:reg:<anno>:<tipo documento>:<sezionale>`.
9	Nome del file indice (relativo alla posizione del file pdv.xml) firmato e marcato. Il primo file è il file principale del documento.
10	Indicare in `terms:medium` il mime type del contenitore PCKS#7 `application/pkcs7-mime`.
11	Indicare in `terms:format` il mime type del file contenuto `text/xml`.
12	Nome del file indice firmato (relativo alla posizione del file pdv.xml) in formato P7M. Opzionale, è considerato un file annesso al file principale.
13	Nome del file indice (relativo alla posizione del file pdv.xml) in formato XML. Opzionale, è considerato un file annesso al file principale.

8.13. Esempio pacchetto D58 Riepilogo documenti firmati

Con il prefisso D58 possono essere individuate delle tipologie documentali particolari, non censite in altre categorie. In questo caso la tipologia comprende il riepilogo dei documenti firmati.

```
<fileGroup> ①
  <dc>
    <terms:type>Riepilogo firme LUL</terms:type> ②
    <terms:subject>Riepilogo documenti firmati ottobre 2017</terms:subject> ③
    <terms:date>2017-11-22T15:15:15+02:00</terms:date>
  </dc>
  <metadata key="produttore:idfiscale">IT01234567891</metadata> ④
  <metadata key="produttore:ragionesociale">Ragione sociale produttore</metadata>

  <metadata key="destinatario:idfiscale">IT01234567891</metadata> ⑤
  <metadata key="destinatario:ragionesociale">Ragione sociale produttore</metadata>

  <metadata key="documento:anno">2017</metadata> ⑥
  <metadata key="documento:tipo">D58</metadata>
  <metadata key="documento:sezionale">lul</metadata>
  <metadata key="documento:data">2017-10-27</metadata>
  <metadata key="documento:datainizio">2017-10-01</metadata>
  <metadata key="documento:datatermine">2017-10-31</metadata>

  <registro>urn:entaksi:IT01234567891:_default:reg:2017:D58:lul</registro> ⑦

  <file>
    <dc>
      <terms:title>DATA/STAMPE/Firma_201710_Marche_INF.PDF</terms:title> ⑧
      <terms:format>application/pdf</terms:format>
    </dc>
  </file>
</fileGroup>
```

Metadato	Descrizione
1	Gruppo di file relativo al documento “Riepilogo firme LUL”. Il gruppo è composto da un file, che è il PDF con il riepilogo dei documenti firmati.
2	Impostare `terms:type` a `Riepilogo firme LUL`.
3	Contenuto del documento. Per i cedolini impostare a `Riepilogo documenti firmati <mese> <anno>`.
4	Il soggetto produttore è l’azienda datore di lavoro indicata con identificativo fiscale e ragione sociale
5	Utilizzare come soggetto destinatario le stesse informazioni del soggetto produttore.
6	Dati del documento. Indicare l’anno di emissione del cartellino presenze (cioè quello relativo alla mensilità) in `documento:anno`. documento:tipo è sempre uguale a D58, `documento:sezionale` è sempre uguale a `lul`, `documento:data` contiene la data di emissione del cedolino in formato yyyy-MM-dd. Il periodo va rappresentato mediante i due valori `documento:datainizio` e `documento:datatermine` che riportano, rispettivamente il primo e l’ultimo giorno del mese a cui si riferisce l’elenco dipendenti.
7	Registro di archiviazione. È un URI composto dalle seguenti parti tutte già descritte nelle tag precedenti: `urn:entaksi:<idfiscale-produttore>:<unità organizzativa>:reg:<anno>:<tipo documento>:<sezionale>`.

Metadato	Descrizione
8	Nome del file con il riepilogo delle firme (relativo alla posizione del file pdv.xml). Opzionale, è considerato un file annesso al file principale.

8.14. Esempio pacchetto `D8503` Manuali

Con il prefisso `D85` possono essere individuate delle tipologie documentali particolari, in questo caso la tipologia `D8503` indica dei documenti interni generici di natura non fiscale.

Il pacchetto può contenere, ad esempio, dei manuali di procedure interne.

```
<?xml version="1.0" encoding="UTF-8"?>
<pdv xmlns="http://entaksi.eu/schemas/econ/1.0/" xmlns:terms="http://purl.org/dc/terms/">
  <formato>F999</formato> ①
    <fileGroup> ②
    <dc> ③
      <terms:type>Documenti interni</terms:type> ④
      <terms:subject>IO ISO 20140720 Regolamento interno 1.6.0</terms:subject> ⑤
      <terms:format>application/pdf</terms:format> ⑥
    </dc>
    <registro>urn:entaksi:IT01234567890:_default:reg:2017:D85DOC:_default</registro> ⑦
    <file>
      <dc>
        <terms:title>IO ISO 20140720 Regolamento interno 1.6.0.pdf</terms:title> ⑧
        <terms:creator>Entaksi Solutions SpA</terms:creator> ⑨
        <terms:date>17/07/2017</terms:date> ⑩
      </dc>
    </file>
  </fileGroup>
</pdv>
```

Metadato	Descrizione
1	Il formato `F999` indica che si tratta di un pacchetto generico con indice. Si tratta di una informazione interna che serve per la validazione del pacchetto.
2	Il file è composto da una sequenza di una o più tag `fileGroup` ognuna delle quali rappresenta una unità documentaria (ovvero un documento) inviato in conservazione. La tag si chiama `fileGroup` perché può contenere più di un file, dove il primo file è il file principale relativo al documento vero e proprio, mentre i file successivi sono annessi o allegati al file. Nella maggior parte dei casi ci sarà un solo file, ma è possibile, ad esempio, archiviare la scansione della ricevuta di ritorno della raccomandata con cui è stata inviata la fattura. Questa scansione verrebbe archiviata come secondo file, cioè come annesso al documento e insieme ad esso costituisce un'unica unità documentaria.`
3	I metadati dell'unità documentaria sono indicati mediante un gruppo di tag con namespace `terms` che si trovano all'interno della tag `dc` e una serie di tag `metadata` che hanno un attributo `key` e un valore. I metadati `dc` sono i metadati *Dublin Core*, uno standard internazionale per la classificazione dei documenti in base a criteri archivistici. I metadati `metadata` sono altri metadati relativi ai criteri gestionali. In questo caso i metadati utilizzati appartengono tutti al gruppo *Dublin Core*.
4	Il termine *Dublin Core* `terms:type` contiene una descrizione comprensibile per l'utente del tipo di documento archiviato. In questo caso indica che il file è di tipo "Documenti interni".
5	Il termine *Dublin Core* `terms:subject` contiene una descrizione comprensibile per l'utente del tipo di documento archiviato. Nel caso dei documenti interni si è scelto di indicare il titolo del documento.
6	Il termine `terms:format` riporta il mime type del file. Per i documenti PDF il mime type è `application/pdf`.

Metadato	Descrizione
7	La tag `registro` specifica il registro di archiviazione del documento che si può ricavare dai metadati precedenti secondo questa sintassi: `urn:entaksi:<id-fiscale-produttore>:<unità-organizzativa>:reg:<anno>:<tipodocumento>:<sezionale>`.
8	Il termine *Dublin Core* `terms:title` contiene il nome del file del documento, comprensivo di estensione.
9	Il termine `terms:creator` indica l'entità che ha provveduto alla creazione del documento.
10	Il termine *Dublin Core* `terms:date` indica la data in cui il file è stato firmato digitalmente.

9. Pacchetti di distribuzione

Il Pacchetto di Distribuzione è l'entità con cui i documenti conservati possono essere estratti al fine della distribuzione degli stessi.

In base ai criteri di selezione dei documenti il Pacchetto di Distribuzione viene assemblato dal sistema di conservazione includendo:

- le unità documentarie all'interno dell'archivio corrispondenti ai criteri di selezione;
- l'insieme delle prove di conservazione delle unità documentarie selezionate, cioè gli indici firmati dei PDA in cui sono contenute.

Il Pacchetto di Distribuzione viene reso disponibile sotto forma di un file ZIP contenente:

- un indice di distribuzione firmato digitalmente dal Responsabile del Servizio di Conservazione, che costituisce anche il rapporto di distribuzione
- le unità documentarie corrispondenti ai criteri di selezione
- l'insieme delle prove di conservazione

L'utente può effettuare sul sistema una ricerca in base ad un insieme di criteri di selezione. La ricerca produce come risultato un insieme di unità documentarie.

L'utente può poi richiedere la produzione di un Pacchetto di Distribuzione contenente il risultato di questa ricerca. Il risultato di questa operazione è in realtà costituto da uno o più Pacchetti di Distribuzione, nel caso in cui l'insieme dei documenti selezionato sia troppo ampio per rientrare in un solo pacchetto.

I Pacchetti di Distribuzione vengono messi a disposizione per il download esclusivamente da parte dell'utente che li ha richiesti o eventualmente veicolati tramite le modalità definite tra l'utente e il Responsabile del Servizio di Conservazione.

I Pacchetti di Distribuzione rimangono disponibili per il download per un periodo di tempo concordato tra l'utente e il Responsabile del Servizio di Conservazione, prima di essere scartati.

9.1. Stati della richiesta di PDD

Gli stati della richiesta di PDD sono riportati nel paragrafo Tabella dei codici degli stati di una richiesta di PDD.

10. Codice di errore

In caso di errore le API restituiscono un messaggio contenente alcune informazioni sul problema.

Il messaggio ha un formato di questo tipo in XML:

```
<errore>
  <esito>
    <codice>codice di errore</codice>
    <messaggio>messaggio di errore</messaggio>
  </esito>
</errore>
```

In JSON:

```
{
  "esito": {
    "codice": "codice di errore",
    "messaggio": "messaggio di errore"
  }
}
```

Il codice di errore è un numero di tre cifre che può assumere i seguenti valori:

Codice	Descrizione
000	Nessun errore
001	Accesso negato
002	Entità non trovata
003	L'utente non ha accesso al rivenditore
004	L'utente non ha accesso al contratto
005	L'utente non ha accesso all'azienda
006	L'utente non ha accesso alla fattura
007	L'utente non ha accesso alla ricevuta
008	L'utente non ha accesso al volume di conservazione
009	L'utente non ha accesso all'utente
010	L'utente non ha accesso alla funzione
011	Valore obbligatorio mancante
012	Valore errato per il parametro
013	Azienda duplicata
014	Azienda disabilitata
015	Fattura duplicata

Codice	Descrizione
016	Errore nel formato XML
017	Transizione di stato non consentita
018	Creazione diretta dell'azienda non prevista
019	L'utente ha raggiunto il numero massimo di aziende
020	Nessun file caricato
021	Rapporto di versamento non disponibile
022	File del Pacchetto di Versamento non disponibile
023	Dati mancanti
024	Superata la dimensione massima del file XML
025	Documento con numerazione duplicata non ammesso
026	Modifica dell'identificativo fiscale non consentita
027	Modifica del codice unità organizzativa non consentita
028	Rivenditore duplicato
029	Client non disponibile
030	Modifica del PDD non consentita in questo stato
031	Rimozione dell'utente non consentita
032	Contenuto del PDD non disponibile
033	Formato obsoleto
034	Servizio sospeso
035	Servizio non ancora attivo
036	La fattura contiene errori di validazione
037	L'utente non ha accesso all'archivio dell'azienda
038	Modifica del codice fiscale non consentita
999	Errore interno

Il messaggio di riporta una descrizione più specifica dell'errore inclusi eventuali dati di contesto.

11. Codici

11.1. Stati della fattura

Tabella 3. Stati della fattura

Codice	Descrizione	Stato
`S00`	Nuova fattura	Fattura appena creata
`S01`	Attesa del caricamento XML	Deve essere caricato il contenuto XML
`S02`	Attesa della conferma	La fattura è in stato di bozza e attende di essere confermata
`S03`	Attesa delega	La fattura è in attesa della delega sull'azienda
`S04`	Attesa firma	La fattura è pronta e in attesa di essere firmata digitalmente
`S05`	Attesa invio	La fattura è pronta e in attesa di essere inviata allo SDI
`S06`	Attesa risposta da SDI	La fattura è stata inviata allo SDI ed è in attesa di risposta
`S07`	Ricevuta notifica decorrenza termini	È stata ricevuta una notifica di decorrenza termini
`S08`	Ricevuta attestazione di impossibilità di recapito	È stata ricevuta una attestazione di impossibilità di recapito
`S09`	Fattura accettata	La fattura è stata accettata dal destinatario
`S10`	Fattura rifiutata	La fattura è stata rifiutata dal destinatario
`S11`	Fattura scartata	È stata ricevuta una notifica di scarto dallo SDI
`S12`	Fattura annullata	La fattura è stata annullata dall'utente
`S13`	In conservazione	La fattura è in conservazione
`S14`	Invio in corso	L'invio della fattura è in corso
`S15`	Attesa firma	La fattura è in attesa dalla firma digitale
`S16`	Pronta per la conservazione	La fattura è pronta per essere messa in conservazione
`S17`	Fattura ricevuta	La fattura è stata ricevuta dal canale ricezione
`S18`	Conferma notifica EC	L'invio della notifica EC è in attesa di conferma da parte dell'operatore
`S19`	Notifica EC confermata	La notifica EC è stata confermata dall'operatore
`S20`	Attesa firma notifica EC	Attesa firma notifica EC

Codice	Descrizione	Stato
S21	Notifica EC pronta per l'invio	Notifica EC pronta per l'invio
S22	Invio notifica EC in corso	Invio notifica EC in corso
S23	Notifica EC inviata	La notifica EC è stata inviata correttamente
S24	Notifica EC scartata	La notifica EC è stata scartata perchè non valida
S25	Errore interno	Errore interno durante il processo di avanzamento
S26	Fattura emessa	Valido per le fatture B2B, indica che la fattura è stata consegnata al destinatario o messa a disposizione nell'area riservata del sito dell'AdE
S27	Fattura ricevuta su AdE	Fattura ricevuta sul sito dell'Agenzia delle Entrate
S28	Fattura eliminata	La fattura è stata elimitata
S29	Fattura da integrare	Indica che la fattura ricevuta dall'operatore economico di San Marino da un operatore economico italiano è stata importata e che va integrata caricando un file di integrazione
S30	File di integrazione da inviare	Indica che è stato caricato un file di integrazione da inviare a HUB-SM per una fattura ricevuta da un operatore economico italiano
S31	Invio file di integrazione in corso	Indica che il file di integrazione è in corso di invio a HUB-SM
S32	Fattura da rifiutare	Indica che la fattura ricevuta è da respingere su HUB-SM
S33	Invio rifiuto in corso	Indica che il sistema sta inviando il rifiuto della fattura a HUB-SM

11.2. Esito fattura

Tabella 4. Esito della fattura

Codice	Descrizione
E00	File fattura in elaborazione.
E01	File fattura inviato al Sistema di Interscambio.
E02	File fattura consegnato al destinatario (è stata ricevuta una notifica di consegna).
E03	File fattura scartato (è stata ricevuta una notifica di scarto).
E04	Fattura accettata (è stata ricevuta un notifica di esito con accettazione da parte del destinatario).
E05	Fattura rifiutata (è stata ricevuta una notifica di esito con rifiuto da parte del destinatario).

Codice	Descrizione
E06	Decorrenza termini (è stata ricevuta una notifica DT, decorrenza termini).
E07	Impossibilità di recapito (è stata ricevuta una notifica AT, attestato di trasmissione).
E08	L'elaborazione della fattura è stata annullata.
E09	Il file fattura è stato eliminato.
E10	Fattura ricevuta.
E11	File fattura non consegnato al destinatario. Valido per le fatture B2B, indica che per la fattura è presente una notifica di mancata consegna. La fattura è disponibile in consultazione nell'area riservata del sito AdE.
E12	Fattura archiviata ma non transitata da SdI.

11.3. Tabella dei codici di validazione dei PDV

Tabella 5. Codici validazione PDV

Codice	Descrizione
E000	Il pacchetto di versamento non contiene allegati.
E001	Il file allegato al pacchetto di versamento non è uno zip.
E002	Il pacchetto di versamento non contiene file.
E003	Impossibile creare le unità documentali.
E004	Impossibile estrarre i metadati sull''unità documentale.
E005	Il file contiene fatture non conformi.
E006	Il file non ha un nome valido.
E007	Il file non è valido.
E008	Il file non può essere verificato.
E009	Il file non è firmato.
E010	Il file non ha una firma valida.
E011	La fattura non contiene il numero di documento.
E012	Il nome del file {0} indicato nella notifica non corrisponde al nome del file del documento principale.
E013	Sono presenti delle notifiche senza la fattura.
E014	L'anno della fattura {0} deve essere uguale a quello delle altre fatture del pacchetto di versamento.
E015	Errore nell''estrazione del zip della notifica AT {0}.

Codice	Descrizione
E016	Il pacchetto di versamento non è conforme. {0}
E017	L'unità archivistica {0} non ha indicato il metadato {1}.
E018	Il registro dell''unità archivistica {0} non è corretto. {1}
E019	L''hash del file {0} non è valido.
E020	Il pacchetto di versamento contiene fatture con data successiva alla data di fine delega dell'azienda.
E021	Il metadato {0} dell''unità archivistica {1} non è valido.
E022	L''unità archivistica {0} non è valida. {1}
E023	Errore generico.
E024	Il file {0} non è trasmesso o ricevuto dalla casella PEC {1}.
E025	I messaggi dovrebbero essere tutti dell''anno {0}, ma il file {0} contiene un messaggio dell''anno {1}.
E026	Firma elettronica non valida sul file {0}. Errore: {1}
E027	Il file {0} non è dichiarato nell'indice.
E028	Il file {0} è dichiarato nell'indice ma non è presente nel pacchetto.
E029	I file eml devono essere posizionati in una cartella con il nome della casella PEC da archiviare.
E030	Il file {0} è vuoto.
A000	Messaggio informativo generico.
A001	La fattura {0} non ha le notifiche.
A002	Le notifiche della fattura {0} non sono complete.
A003	La fattura {0} non ha la notifica {1}.
A005	Il file della notifica AT {0} deve avere estensione .zip
A006	Il file {0} non è firmato.
A007	Il file {0} è una notifica di ACCETTAZIONE PEC, ma manca il messaggio inviato.
A008	Il file {0} non ha validità legale in quanto manca la notifica di ACCETTAZIONE PEC.
A009	Il file {0} non ha una firma valida.
A010	La firma digitale del file {0} non può essere verificata.

11.4. Tabella dei codici degli stati del PDV

Tabella 6. Codici degli stati del PDV

Codice	Descrizione	Azioni possibili
`P00`	Stato iniziale. È lo stato in cui si trova il pacchetto dopo che è stato creato e prima che venga caricato il contenuto. I pacchetti creati con l'API `POST /aziende/{idAzienda}/pdv/{pdvFormat}` non passano da questo stato.	Caricamento del contenuto del pacchetto.
`P01`	In costruzione. È uno stato valido solo per i pacchetti di tipo `F001` e `F003`, cioè quelli relativi alle fatture elettroniche gestite dal sistema. Indica che il pacchetto è stato definito ed è in corso l'assegnazione delle fatture elettroniche destinate all'archiviazione.	Il sistema di gestione delle fatture elettroniche assegna le fatture che hanno completato il ciclo di gestione.
`P02`	Elaborazione in corso. È lo stato in cui si trova il pacchetto dopo essere stato caricato quando sono in corso le procedure di verifica e validazione.	L'utente attende il completamento dell'operazione.
`P03`	Elaborato È lo stato in ci si trova il pacchetto al termine delle operazioni di verifica e validazione. Il pacchetto verrà preso in carico per le operazioni successive in base all'esito della verifica e validazione.	L'utente attende il completamento dell'operazione.
`P04`	Firma in corso È in corso la firma del rapporto di versamento.	L'utente attende il completamento dell'operazione.
`P05`	Archiviato Il contenuto del pacchetto di versamento è stato caricato nell'archivio. Il pacchetto verrà preso in carico per le operazioni successive relative alla pubblicazione del rapporto di versamento.	L'utente attende il completamento dell'operazione.
`P06`	Accettato Il pacchetto di versamento è stato accettato.	L'utente può scaricare il rapporto di versamento firmato.
`P07`	Rapporto archiviato Il pacchetto di versamento è stato accettato e il rapporto di versamento è stato archiviato. I rapporti di versamento, in quanto documenti digitali, vengono automaticamente versati nel sistema di conservazione dopo l'accettazione del pacchetto.	L'utente può scaricare una duplicato del rapporto di versamento firmato.
`P08`	Rifiutato Il pacchetto di versamento è stato rifiutato a causa di errori bloccanti nel processo di verifica e validazione.	L'utente può scaricare il rapporto di versamento in cui si trovano i messaggi di errore che hanno determinato il rifiuto.
`P09`	Verifica PDD Valido solo per i pacchetti di tipo `F998` (pacchetti contenente pacchetti di distribuzione provenienti da un altro conservatore), significa che è in corso la verifica sulla leggibilità del pacchetto proveniente da un precedente conservatore.	L'utente attende il completamento dell'operazione. In alcuni casi la verifica del PDD di un precedente conservatore richiede l'intervento manuale degli operatori di backend. In questo caso l'operazione ha una durata maggiore.

Codice	Descrizione	Azioni possibili
P10	Eliminato Il pacchetto di versamento è stato eliminato.	I file del pacchetto di versamento e il rapporto di versameno non sono più disponibili.

11.5. Tabella dei codici delle azioni sui PDV

Tabella 7. Codici delle azioni sui PDV

Codice	Descrizione
A00	*Non usato.*
A01	Definizione del PDV.
A02	Caricamento del PDV.
A03	Conferma del PDV.
A04	Archiviazione del PDV.
A05	Validazione completata con successo.
A06	Invio alla firma del rapporto di versamento.
A07	Firma del rapporto di versamento.
A08	Archiviazione del PDV.
A09	Scarto dei file zip che compongono il PDV. Il rapporto di versamento rimane disponibile.
A10	Validazione completata con errori.
A11	Rimozione dei file non rilevanti.
A12	Prepara il rapporto di versamento per l'invio alla firma.
A13	Scarto dei file del PDV e del rapporto di versamento.

11.6. Tabella dei codici degli stati del PDA

Tabella 8. Codici degli stati del PDA

Codice	Descrizione
T00	Stato iniziale.
T01	Elaborato.
T02	Incompleto.
T03	Da firmare.
T04	Firma in corso.

Codice	Descrizione
T05	Firmato.
T06	*Non usato.*
T07	Chiuso.
T08	Scartato.

11.7. Tabella dei codici delle azioni sui PDA

Tabella 9. Codici delle azioni sui PDA

Codice	Descrizione
A00	*Non usato.*
A01	Creazione del PDA.
A02	Validazione completata con successo.
A03	Validazione completata con errori.
A04	Invio alla firma dell'indice del PDA.
A05	Inizio del processo di firma.
A06	Conclusione del processo di firma.
A07	Chiusura del PDA.
A08	Invio per nuova validazione.
A09	Avvio del controllo di integrità.
A10	Controllo di integrità completato con successo.
A11	Controllo di integrità completato con errore.
A12	Scarto del PDA.

11.8. Tabella dei codici degli stati di una richiesta di PDD

Tabella 10. Codici degli stati del PDD

Codice	Descrizione	Stato
R00	Richiesta in bozza	La ricerca è in bozza e può essere modificata dall'utente.
R01	Richiesta salvata	La ricerca è stata salvata come definitiva. Non può più essere modificata dall'utente e verrà eseguita dal primo processo disponibile. L'utente può riportare la ricerca in stato di bozza.
R02	Ricerca in corso	La ricerca è in corso. Non può essere modificata dall'utente.

Codice	Descrizione	Stato
R03	Risultati disponibili	I risultati della ricerca sono disponibili. Dopo un determinato periodo di tempo passa direttamente in stato R07 oppure l'utente richiede la formazione di un PDD portando lo stato in R04.
R04	PDD richiesto	È stato richiesto che i risultati della richiesta siano assemblati in un PDD. L'utente può annullare questa richiesta riportando lo stato in R03.
R05	PDD in costruzione	Il processo di assemblaggio del PDD è in corso. L'utente non può modificare.
R06	PDD pronto	Il PDD è pronto per essere scaricato. Dopo un determinato periodo di tempo la richiesta passa nello stato R07.
R07	PDD da eliminare	La richiesta è destinata ad essere eliminata automaticamente. Dopo un determinato periodo di tempo lo stato passa a R08.
R08	PDD eliminato	La richiesta è stata cancellata per cui l'eventuale PDD generato non è più disponibile per il download.
R09	PDD pronto per la firma	L'indice del PDD è stato assemblato ed è pronto per essere firmato.
R10	Firma del PDD in corso	Il processo di firma del PDD è in corso.
R11	PDD firmato	Il processo di firma del PDD è stato concluso.
R12	Raccolta firmata	La raccolta di documenti è stata firmata.

11.9. Tabella dei codici delle azioni sulle richieste di PDD

Tabella 11. Codici delle azioni sulle richieste di PDD

Codice	Descrizione
PDD01	Creazione della richiesta di documenti.
PDD02	Azione eseguita dall'utente con la quale viene finalizzata la richiesta di documenti
PDD03	Azione eseguita dal sistema che indica che è stato programmato il lavoro di ricerca.
PDD04	Azione eseguita dal lavoro di ricerca al termine dell'esecuzione per indicare che la ricerca è stata completata e i risultati sono disponibili.
PDD05	Azione eseguita dall'utente con cui viene chiesto di formare un PDD in base ai risultati della ricerca.
PDD06	Azione eseguita dal sistema che indica che è stato programmato il lavoro di formazione del PDD.
PDD07	Azione eseguita dal lavoro di formazione del PDD al termine dell'esecuzione per indicare che il PDD è disponibile.
PDD08	Azione eseguita dall'utente che indica il download del PDD.

Codice	Descrizione
PDD09	Azione eseguita dal sistema, che avvia il lavoro di rimozione del PDD dopo che è stato reso disponibile per un certo periodo di tempo.
PDD10	Azione eseguita dal lavoro di rimozione del PDD che indica il completamento delle operazioni di rimozione del PDD.
PDD11	Azione eseguita dal lavoro di formazione del PDD al termine dell'esecuzione dell'operazione di creazione dell'indice del PDD.
PDD12	Azione eseguita dal sistema che indica che è stato avviato il processo di firma del PDD.
PDD13	Azione eseguita dal lavoro di formazione del PDD al termine dell'esecuzione dell'operazione di firma.
PDD14	Azione eseguita dall'utente con la quale viene creata una nuova raccolta di documenti.

11.10. Tabella dei codici delle operazioni tracciate nell'audit log

Tabella 12. Codici delle operazioni tracciate nell'audit log

Codice	Descrizione
V000	Azione generica non specificata.
V001	(READ) Accesso in lettura dell'entità.
V002	(UPDATE) Accesso in aggiornamento dell'entità.
V003	(DELETE) Cancellazione dell'entità.
V004	(READ_LOG) Lettura del log dell'entità.