**SER-STORE EXPORT REPORT**

***Import dei dati del venduto***

Versione documento: 1.0

Sommario

[Stato del documento 3](#_Toc156682559)

[1. Introduzione e obiettivi 4](#_Toc156682560)

[1.1. Scopo del documento 4](#_Toc156682561)

[2. Disegno Interfaccia 4](#_Toc156682562)

[2.1. Componenti dell’Interfaccia 4](#_Toc156682563)

[2.2. Descrizione dettagliata dell’Interfaccia 4](#_Toc156682564)

[2.3. Struttura Logica dei Dati 5](#_Toc156682565)

[3. Tracciato dei dati di input 5](#_Toc156682566)

[3.1. Tracciato con il dettaglio delle transazioni. 5](#_Toc156682570)

[4. File CSV 6](#_Toc156682571)

[4.1. Naming del file 6](#_Toc156682576)

[5. Dati errati 7](#_Toc156682577)

[6. Gestione Errori 7](#_Toc156682578)

[6.1. Riferimento negozio 7](#_Toc156682579)

Stato del documento

Le informazioni contenute in questo documento aggiornano e sostituiscono quelle contenute nelle versioni precedenti. Ogni modifica del documento sarà controllata nella apposita procedura.

|  |  |  |  |
| --- | --- | --- | --- |
| Revisione | data | sintesi dei cambiamenti | Modificato da |
| 1.0 | 20/01/2024 | Prima versione | Carlo Blatti |

1. Introduzione e obiettivi
   1. Scopo del documento

Il presente documento ha lo scopo di descrivere l’interfaccia di caricamento progetto Store Export Report e le caratteristiche dei file utili agli import dei dati del venduto.

1. Disegno Interfaccia
   1. Componenti dell’Interfaccia

I componenti dell’Interfaccia sono qui di seguito elencati:

* **Sistema di origine**: sistemi fatturazione dei negozi.
* **Sistema di destinazione**: sistema SER.
* **Modulo di estrazione**: creazione dei file Excell a seguito di richiesta
* **Modulo di elaborazione**: script Go e Python che rispettivamente elaborano i dati e producono i file xlsx.
* **Modulo di caricamento**: script shell che richiama sqlloader: programma di Oracle adibito a caricamento dei dati da file. Una stored procedure Oracle invocata alla fine del caricamento processa i dati storicizzandoli nelle tabelle relazionali.
* **Dipendenza da altre interfacce**: nessuna.
  1. Descrizione dettagliata dell’Interfaccia

SER è l’interfaccia per caricare dati degli scontrini battuti nei vari negozi aderenti al programma. Ciascun sistema dovrà generare dei file CSV nel formato descritto in questo documento e renderlo disponibile all’elaboratore. I file conterranno i dati relativi agli scontrini battuti.

L’interfaccia quindi è costituita da:

* Processi per creare i file CSV sui sistemi di origine.
* Acquisizione dei file via interfaccia SER.
* Processazione dei dati da parte di script Go e Python.
  1. Struttura Logica dei Dati
* Il caricamento in SLR avverrà secondo la seguente logica: i record verranno caricati se esiste corrispondenza tra IdSocietà e IdNegozio, eventuali correzioni a file già inviati dovranno essere gestiti nel seguente modo:
  + Se i record sono esistenti, vengono sostituiti.

Nel caso di sostituzione, occorre tener presente, che tutti i dati esistenti per la stessa IdSocietà/IdNegozio, vengono cancellati e sostituiti con quelli nuovi.

E’ responsabilità della catena/negozio la predisposizione del file csv in modo corretto.

1. Tracciato dei dati di input

4. 1. Tracciato con il dettaglio delle transazioni.

La seguente tabella illustra il primo tracciato record

|  |  |  |  |
| --- | --- | --- | --- |
| **Campo** | **Tipo** | **Formato** | **Note** |
| Id Società | VARCHAR2 |  | Codice catena nel sistema SER. Valore fornito alla società al momento dell’attivazione del servizio.  IdSocietà non associati al nome Società, provocano il mancato caricamento dell’intero file. |
| Id Negozio | NUMBER |  | Chiave interna del negozio nel sistema SER fornito alla società al momento dell’attivazione del servizio.  IdNegozio non associati all’IdSocietà di riferimento, provocano il mancato caricamento dell’intero file. |
| Id Cassa | VARCHAR2 |  | Identificativo cassa. Si tratta del codice della cassa del negozio, solitamente è quello che viene stampato anche sullo scontrino. |
| Id Scontrino | NUMBER |  | Identificativo scontrino |
| Data | CHAR(8) | YYYYMMDD | Data emissione scontrino |
| Ora | CHAR(4) | HH:24MI | Ora emissione scontrino |
| Categoria prodotto | VARCHAR2 |  | Nei primi caratteri del campo riportare una codifica univoca della categoria (alfanumerico).  Va benissimo anche un cod. categoria già utilizzato dal negozio.  La codifica della categoria dovrà essere separata dalla successiva descrizione utilizzando un carattere separatore “-“, ad esempio, unica nota importante è che non si tratti del carattere “;” )  I rimanenti caratteri contengono la descrizione estesa della categoria.  Ad es.:  123456-SMALL LEATHER GOODS  abcdef-CUFFLINKS / JEWELS  123abc-OTHER ACCESSORIES L |
| Importo Lordo | NUMBER | NNNNNN.NN | Importo lordo. Il separatore per i decimali il punto.  I decimali, così come le ultime cifre delle decine vanno sempre riportati anche se non sono significativi (00). |
| Numero pezzi | NUMBER |  | Numero pezzi venduti per categoria prodotto |
| Codice Iva | NUMBER |  | Codice iva applicato al cliente, utile per i successivi calcoli.  Ad esempio 22 oppure 10 |

(\*) Per ogni categoria di prodotto venduta con lo scontrino deve essere prodotta una riga.

Ad esempio, a fronte dei seguenti scontrini emessi il 17/10/2019:

Esempio

= = = = = = = = = = = = = = = =

File con 6 record:

OVS

1234;5678;01;202011;20191017;0841;000042-categoria A;7.50;1;22

1234;5678;01;202011;20191017;1318;000016-categoria B;15.00;1;10

1234;5678;01;212011;20191017;1318;000042-categoria A;24.00;1;22

1234;5678;01;212011;20191017;1425;000020-categoria C;6.00;1;10

1234;5678;01;222011;20191017;1521;000042-categoria A;12.00;1;22

1234;5678;01;232011;20191017;1521;000042-categoria A;12.00;1;22

6

1. File CSV


5. 1. Naming del file

Il file da produrre da parte del sistema negozio dovrà avere le seguenti caratteristiche:

* Formato File: CSV
* Delimitatore: “;” (punto e virgola). Attenzione, i record non devono contenere un ‘;’ finale. Il semicolon (carattere ‘;’) è usato per delimitare le colonne all’interno del CSV e se ce ne fosse uno finale verrebbe interpretato come esigenza di ulteriore colonna, considerando il file non valido in fase di processamento.
* la prima riga del file deve contenere il nome della società/catena ad esempio: Clayton
* l’ultima riga (record “tappo”) include il totale dei record contenuti

(ad es. 6), sono escluse dal conteggio dei record la prima e l’ultima riga.

1. Dati errati

Nel caso si renda necessario correggere alcuni valori, il file relativo ai giorni errati dovrà essere re-inviato con tutti i dati della giornata corretta.

In questo caso i dati precedentemente caricati verranno interamente sostituiti con i nuovi dati.

1. Gestione Errori

Ogni negozio dovrà fornire il riferimento delle persone da contattare in caso di anomalia:

* nominativo;
* indirizzo mail;
* recapito telefonico;
* preferenza di reperibilità (telefono; mail);
* orari di reperibilità telefonica.

1. Inserimento nuova società e negozi

Quando una nuova società richiede di essere inserita a sistema bisogna configurarla all’interno del file compagnie.json(guardare STF per maggiori dettagli), fornendone i seguenti campi:

* CompanyName (Nome Società)
* CompanyId (Id Società).
* CompanyShopsId (Array che contiene gli Id dei negoni della società) (\*)

Ad esempio:

[

{

"companyName": "Clayton",

"companyId": "070",

"companyShopsId": [

"8362",

"8363"

]

},

]

**(\*)Attenzione:a fronte di richiesta di inserimento di un nuovo negozio per la società, l’id del negozio va inserito nel file e poi fornito alla società richiedente.**