



Winning Software Solution

winningsoftwaresolution@gmail.com

Manuale E-commerce Shop Chain

<i>Informazioni</i>	
<i>Redattori</i>	WinningSoftwareSolution
<i>Uso</i>	esterno

Descrizione

Manuale e-commerce.

Contents

1	Introduzione	2
2	Configurazione wallet	2
2.1	Wallet Metamask	2
2.1.1	Configurazione	2
2.1.2	Collegamento a Shop Chain	2
3	Visualizzazione e gestione transazioni	3
3.1	Visualizzazione transazione	3
3.2	Gestione transazione	4
4	Vendere un prodotto con Shop Chain	5
4.1	Requisiti e configurazione	5
4.1.1	Python, Pip e dipendenze	5
4.1.2	Mnemonic phrase	5
4.1.3	Provider	6
4.1.4	Configurazione	6
4.2	Caricamento del prodotto da vendere	7
4.2.1	Sell.py da terminale	7
4.2.2	Script python	7
4.3	Reindirizzamento alla landing page	8

1 Introduzione

Attraverso l'utilizzo della blockchain *Polygon*, Shop Chain fornisce un sistema decentralizzato per il pagamento di prodotti in MATIC. Utilizzando Shop Chain è possibile effettuare una qualsiasi attività di compravendita in maniera sicura per entrambi gli attori dello scambio. L'acquirente che desidera acquistare un prodotto da un venditore in un e-commerce potrà effettuare il suo pagamento in MATIC, ovvero il token nativo della blockchain *Polygon*. Il venditore non può disporre immediatamente del saldo ricevuto poiché sarà bloccato fino alla ricezione del pacco, momento nel quale l'acquirente può sbloccare i fondi attraverso la scannerizzazione di un codice QR presente sulla scatola. Il presente documento ha la funzione di descrivere in dettaglio la procedura da seguire per l'e-commerce al fine di vendere correttamente i propri prodotti.

2 Configurazione wallet

2.1 Wallet Metamask

Per poter usufruire a pieno del nostro servizio sarà necessario l'utilizzo di un wallet Metamask. Visitare metamask.io per maggiori informazioni.

2.1.1 Configurazione

Se il network Polygon non è già stato inserito nel proprio wallet Metamask, procedere con la lettura del seguente paragrafo.

Per l'inserimento e la configurazione del network Polygon nel proprio wallet Metamask è necessario utilizzare i seguenti dati:

- Network Name: **Matic Mainnet**;
- New RPC Url: **<https://rpc-mainnet.maticvigil.com/>**;
- ChainID: **137**;
- Currency Symbol: **MATIC**;
- Block Explorer URL: **<https://explorer.matic.network/>**.

Una volta inseriti i dati e aggiunto il network, ricordare di selezionare il network.

2.1.2 Collegamento a Shop Chain

Una volta inserito correttamente il network è necessario collegare il proprio wallet metamask a Shop Chain. Nella webapp di Shop Chain, se non si ha già collegato in precedenza il proprio wallet:

- selezionare "Connect Wallet" all'apertura del pop-up (Fig. 1);
- approvare il collegamento dal proprio wallet Metamask.

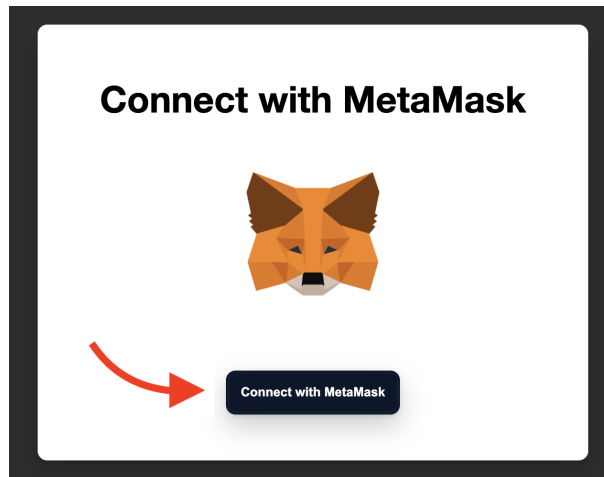


Figure 1: Connessione del wallet Metamask a Shop Chain.

3 Visualizzazione e gestione transazioni

Per la visualizzazione delle transazioni ricevute da un e-commerce in entrata, è necessario recarsi nella sezione **Incoming** (Fig. 2) su Shop Chain. Selezionando una transazione sarà possibile gestirla e vederne i dettagli.



Figure 2: Sezione Incoming.

3.1 Visualizzazione transazione

Selezionando una transazione è possibile visualizzare le seguenti informazioni (Fig. 3 riquadro 1):

- **Price:** indica il prezzo dell'oggetto acquistato;
- **From:** indica l'address del wallet dell'acquirente;
- **To:** indica l'address del proprio wallet, ovvero il wallet che ha ricevuto il pagamento;
- **Opened:** indica la data in cui è stato eseguito l'acquisto;

- **Expire:** indica la data in cui scade l'ordine e in cui i fondi vengono automaticamente rimborsati, si tratta di 14 giorni dopo l'acquisto.
- **Stato dell'ordine:**
 - **Open:** l'ordine è ancora aperto e in attesa di conferma;
 - **Confirmed:** l'ordine è già stato confermato attraverso la scannerizzazione del codice QR dall'acquirente;
 - **Expired:** l'utente ha chiesto il refund poiché non ha ricevuto il pacco entro 14 giorni;
 - **Canceled:** l'ordine è stato rifiutato dall'e-commerce. I fondi utilizzati per il pagamento vengono restituiti all'acquirente nella valuta stabile DAI.

3.2 Gestione transazione

Per ciascuna vendita l'e-commerce può scegliere se (Fig. 3 riquadro 2):

- **Accettare il pagamento:** l'e-commerce che decide di accettare il pagamento deve necessariamente seguire la procedura sotto riportata per permettere all'acquirente di sbloccare i fondi nel momento della ricezione del pacco.

La procedura prevede di:

- Selezionare l'opzione "Download QR";
- stampare il codice QR scaricato in formato .png;
- attaccare la stampa del codice QR sul pacco contenente l'oggetto acquistato prima di spedirlo.

Una volta spedito il pacco è necessario aspettare la ricezione e lo sblocco fondi da parte dell'acquirente. Se il pacco non viene spedito l'acquirente riceverà il rimborso in DAI dopo 14 giorni dall'acquisto.

- **Annullare il pagamento:** l'e-commerce che decide di annullare il pagamento deve selezionare l'opzione "Cancel transaction". L'acquirente viene rimborsato in DAI con l'esatto importo pagato al momento dell'acquisto.

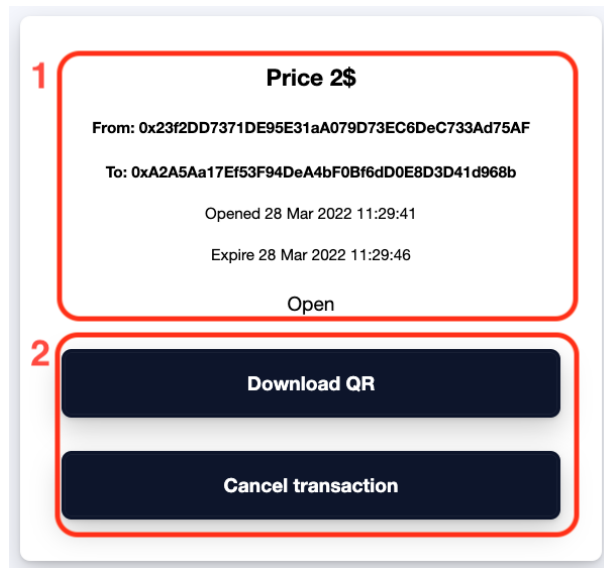


Figure 3: Visualizzazione e gestione vendite.

4 Vendere un prodotto con Shop Chain

Il venditore che desidera vendere il proprio prodotto attraverso Shop Chain deve utilizzare uno script python.

La seguente sezione ha lo scopo di illustrare la corretta procedura per la vendita dei propri prodotti con Shop Chain.

4.1 Requisiti e configurazione

4.1.1 Python, Pip e dipendenze

Al fine di utilizzare correttamente lo script è necessario:

- python3.8;
- pip3.8;
- installare le seguenti dipendenze:
 - web3;
 - python-dotenv;

4.1.2 Mnemonic phrase

La mnemonic phrase del proprio wallet sarà necessaria per la configurazione. Se non si ha già salvato in precedenza la propria mnemonic, è possibile ottenerla da Metamask attraverso la seguente procedura:

1. accedere al proprio wallet Metamask;
2. selezionare il proprio account in alto a destra (Fig. 4);
3. selezionare l'opzione "Settings";
4. selezionare l'opzione "Security & Privacy";
5. selezionare l'opzione "Reveal Secret Recovery Phrase";
6. inserire la propria password.

Una volta ottenuta la propria mnemonic phrase salvarla per i passi successivi.

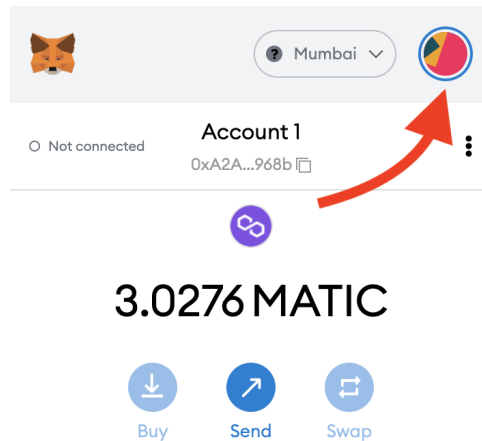


Figure 4: Selezione account metamask.

4.1.3 Provider

L'ultima fase prima della configurazione prevede di ottenere il provider per la Mainnet di Polygon. Sarà dunque necessario registrarsi su moralis.io. Una volta effettuata la registrazione:

1. selezionare la sezione "SpeedyNodes" dal menù;
2. selezionare l'endpoint per Polygon Network;
3. selezionare il protocollo WS;
4. copiare l'url dell'endpoint per la Mainnet (Fig. 5).

Una volta effettuata questa procedura salvare l'url per i passi successivi.

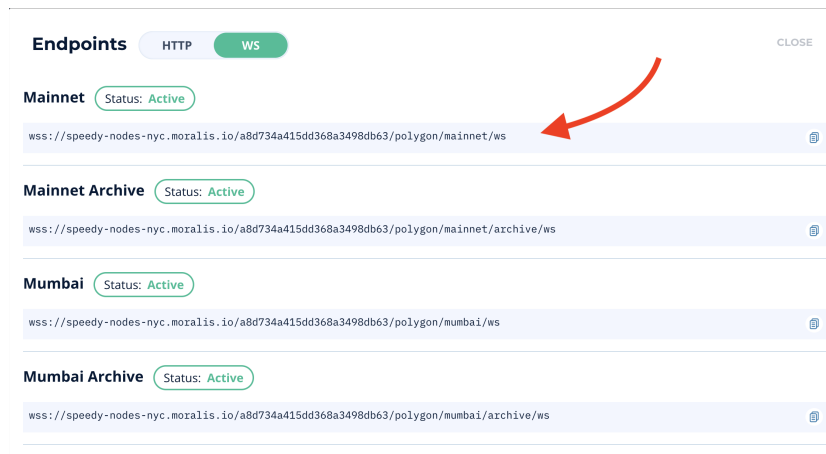


Figure 5: Provider su moralis.io.

4.1.4 Configurazione

Infine è necessario creare un file chiamato `.env` all'interno della stessa cartella contenente lo script `sell.py`. Il file dovrà contenere:

- la mnemonic phrase del proprio wallet, ottenuta al punto 4.1.2;

- il provider ottenuto al punto 4.1.3.

Di seguito un esempio del formato corretto da utilizzare:

```
MNEMONIC=mnemonic phrase del proprio wallet
PROVIDER=provider url
```

4.2 Caricamento del prodotto da vendere

Una volta soddisfatti i requisiti e ultimata la configurazione è possibile procedere con l'esecuzione dello script python. Verranno illustrati due modi per la corretta esecuzione dello script 'sell.py'.

4.2.1 Sell.py da terminale

È possibile eseguire lo script 'sell.py' direttamente da terminale, attraverso il comando:

```
$ python3 sell.py [item price]
```

Sostituire [item price] con il prezzo con il quale si desidera vendere il proprio prodotto. Una volta eseguito lo script, verrà ritornato e stampato l'id e il prezzo del prodotto messo in vendita.

Esempio

Se vogliamo vendere un prodotto al prezzo di 2\$, basterà inviare il comando:

```
$ python3 sell.py 2
```

l'output visualizzato sarà:

```
$ [Adding] Price: 2
$ [Added] Payment entry id: 3, price: 2
```

Dove :

- **payment entry id: 3**: indica che l'id dell'oggetto messo in vendita è 3;
- **price: 2**: indica che il prezzo a cui è stato messo in vendita il prodotto è di 2\$.

4.2.2 Script python

È possibile realizzare un proprio script in python per la vendita di prodotti.

La funzione per mettere in vendita il proprio prodotto è **sell_item** di 'sell.py'. La funzione richiede in input un solo parametro, ovvero il prezzo in dollari del prodotto che si desidera vendere.

È dunque necessario includere nel proprio script la funzione come segue:

```
from sell import sell_item
```

Esempio

Uno script per vendere il proprio prodotto può essere impostato come segue:

```
from sell import sell_item
```

```
sell_item(2)
```


L'output visualizzato sarà:

```
$ [Adding] Price: 2
$ [Added] Payment entry id: 3, price: 2
```

Dove :

- **payment entry id: 3:** indica che l'id dell'oggetto messo in vendita è 3;
- **price: 2:** indica che il prezzo a cui è stato messo in vendita il prodotto è di 2\$.

4.3 Reindirizzamento alla landing page

Al fine di permettere all'acquirente, direttamente dal proprio e-commerce, di essere reindirizzato sulla landing page di Shop Chain per effettuare l'acquisto è necessario strutturare l'url di reindirizzamento come segue:

```
\land?id=<id_item>&r=<pagina_di_ritorno>.
```

e dunque sostituire:

- **<id_item>** con l'id del prodotto che l'acquirente desidera acquistare;
- **<pagina_di_ritorno>** con l'url della pagina alla quale si vuole reindirizzare l'acquirente al termine dell'acquisto.

Dopo aver confermato il pagamento l'acquirente verrà direttamente reindirizzato alla pagina di ritorno, l'url sarà strutturato come segue:

```
<pagina_di_ritorno>?transaction_id=<id_transazione>.
```

dove **<id_transazione>** corrisponde all'id della transazione effettuata dall'acquirente per l'acquisto del prodotto.