

Node.js: pagamenti con PayPal tramite REST API

In questo articolo vedremo come elaborare i pagamenti tramite le REST API di PayPal in ExpressJS.

Dopo aver creato un account da sviluppatore sul sito di PayPal, salviamo le nostre credenziali API in un file di configurazione.

```
'use strict';

module.exports = {
  mode: 'sandbox',
  clientId: '',
  clientSecret: '',
  returnUrl: 'http://localhost:3000/return',
  cancelUrl: 'http://localhost:3000/cancel'
};
```

Installiamo quindi il modulo NPM richiesto:

```
npm install paypal-rest-sdk --save
```

Il flusso del pagamento è il seguente:

1. Viene creato un pagamento su PayPal con l'importo richiesto. PayPal restituisce un oggetto contenente anche il link che l'utente deve usare per autorizzare il pagamento.
2. L'utente autorizza il pagamento su PayPal e viene reindirizzato sull'URL specificato nella variabile `returnUrl`. Qui PayPal restituisce i dati della transazione come parametri GET.

3. Usando i parametri GET di PayPal, completiamo la transazione ed informiamo l'utente sul suo esito.

La route che crea il pagamento deve anche salvare l'importo nella sessione corrente in modo da poter poi ultimare la transazione.

```
'use strict';

const express = require('express');
const router = express.Router();
const paypal = require('paypal-rest-sdk');
const { mode, clientId, clientSecret, returnUrl,
cancelUrl } = require('../config');

paypal.configure({
  mode,
  client_id: clientId,
  client_secret: clientSecret
});

router.post('/checkout', (req, res, next) => {
  const { amount } = req.body;
  const item = {
    name: 'Sample product',
    sku: 'sample',
    price: amount,
    currency: 'EUR',
    quantity: 1
  };
  const paymentData = {
    intent: 'sale',
    payer: {
      payment_method: 'paypal'
    },
  },
```

```

    redirect_urls: {
      return_url: returnUrl,
      cancel_url: cancelUrl
    },
    transactions: [{
      item_list: {
        items: [ item ]
      },
      amount: {
        currency: 'EUR',
        total: amount
      },
      description: 'Sample payment.'
    }]
  };

  paypal.payment.create(paymentData, (error,
payment) => {
    if (error) {
      res.json({ status: false, info: error });
    } else {
      let url = '';
      payment.links.forEach(link => {
        if (link.rel === 'approval_url') {
          url = link.href;
        }
      });
      req.session.amount = amount;
      res.json({ url });
    }
  });
});
});

```

Inviemo l'URL di PayPal via AJAX al codice lato client in modo che quest'ultimo effettui il redirect.

Ora possiamo gestire la fine della transazione quando l'utente viene reindirizzato sulla nostra applicazione.

```
router.get('/return', (req, res, next) => {
  if(req.session.amount) {

    const { paymentId, PayerID } = req.query;
    const { amount } = req.session;

    const executePaymentData = {
      payer_id: PayerID,
      transactions: [{
        amount: {
          currency: 'EUR',
          total: amount
        }
      }]
    };

    paypal.payment.execute(paymentId,
executePaymentData, (error, payment) => {

      delete req.session.amount;

      if (error) {
        res.redirect('/?status=error');
      } else {
        res.redirect('/?status=success');
      }
    });
  } else {
```

```
        res.redirect('/');  
    }  
});
```

Come si può notare, la procedura è veramente lineare e molto semplice da seguire ed implementare.

Codice sorgente

[GitHub](#)