https://verifone.cloud/docs/online-payments/api-integration/google-paytm
Updated: 13-Mar-2024

Google Pay™

You can accept payments through Google Pay™ via?Checkout?(Hosted Payments Page - HPP) or via?eComm API.

Follow the integration steps below to process transactions via Verifone ecomm API.

Required fields

Parameters	Description
payment_provider_contract	In the Payment Provider Contracts section in Verifone Central, set the *Payment Type* to *Google Pay for web*, select your contract and copy the Payment Provider Contract ID. Please note this value is different in Sandbox and in Production.
amount	Amount of the transaction
merchant_reference	Unique UUID you generate and can link the transaction to when the customer returns
currency_code	More on the all currencies supported here: Verifone eCommerce API | Verifone Developer Portal
wallet_type	string Enum: "GOOGLE_PAY"
wallet_payload	object The encrypted payload object provided by the Wallet on the frontend

A Google Pay web integration consists of implementing both client-side and server-side components. You will need to implement the following:

- Configure the Verifone payment gateway ID and your merchant ID to initialize the Google pay client.

https://developers.google.com/pay/api/web/guides/tutorial

- Use the Google Pay token to make a wallet transaction API call and complete the session based on the response

https://verifone.cloud/api-catalog/verifone-ecommerce-api#operation/walletTransaction

Code sample

```
// Specify verifone as the gateway and define your wallet contract
const tokenizationSpecification = {
  type: 'PAYMENT_GATEWAY',
  parameters: {
    'gateway': 'verifone',
    'gatewayMerchantId': 'YOUR_WALLET_CONTRACT'
  }
};

// Specify your google pay merchant ID and name when setting up your paymentDataR

  paymentDataRequest.merchantInfo = {

// @todo a merchant ID is available for a production environment after approval h


// See {@link https://developers.google.com/pay/api/web/guides/test-and-deploy/in

    merchantId: 'YOUR_MERCHANT_ID',
    merchantName: 'YOUR_MERCHANT_NAME'
  };

// Pass the google pay token to your server to make the wallet transaction API ca

function onGooglePaymentButtonClicked() {
  const paymentDataRequest = getGooglePaymentDataRequest();
  paymentDataRequest.transactionInfo = getGoogleTransactionInfo();

  const paymentsClient = getGooglePaymentsClient();
  paymentsClient.loadPaymentData(paymentDataRequest)
      .then(function(paymentData) {
        // handle the response
  paymentToken = paymentData.paymentMethodData.tokenizationData.token;
      fetch("YOUR_SERVER", {
      method: "POST",
        headers: {"Content-Type": "application/json"},
        body: getPayload(JSON.parse(paymentToken)),
      }).then((res) => res.json())
        .then((res) => {
// handle wallet transaction response
          if (res.status.includes('AUTHORIZED')) {
          alert('Payment Successful');
          }
```

```
      })
         .catch((err) => console.log(err));
      })
      .catch(function(err) {
        // show error in developer console for debugging
        console.error(err);
      });
}
```