TPV-Virtual

Google Pay integration in the Virtual POS

Issue: 1.5
Date: 29/05/2024
Reference: RS.TE.CEL.MAN.0025

Redsys, Servicios de Procesamiento, S.L. – Calle Francisco Sancha, 12 – 28034 Madrid (Spain)
www.redsys.es

Authorisation and version control

AUTHOR: Redsys	VALIDATED BY: Comercio Electrónico	APPROVED BY: Redsys
Company: Redsys	Company: Redsys	Company: Redsys
Signature:	Signature:	Signature:
Date: 24/01/2020	Date:	Date:

Version	Date	Concerns	Brief description of the change
1.0	24/01/2020	ALL	First version
1.1	26/02/2021	Point 3	Ds_Merchant_DirectPayment=true sending parameter removed
1.2	30/12/2021	Point 4	A new form of connection is added for the merchant to the Virtual POS by sending decoded data.
1.3	20/02/2023	Point 4	Documentation update by ECv1/ECv2 versions
1.4	14/06/2023	Point 2	Paymethod options
1.5	29/05/2024	Point 3.1	Main errors and how to fix them.

TABLE OF CONTENTS

1. Introduction

Google Pay allows customers to make payments using any credit or debit card stored in their Google Account, including those on Google Play, YouTube, Chrome or an Android device.

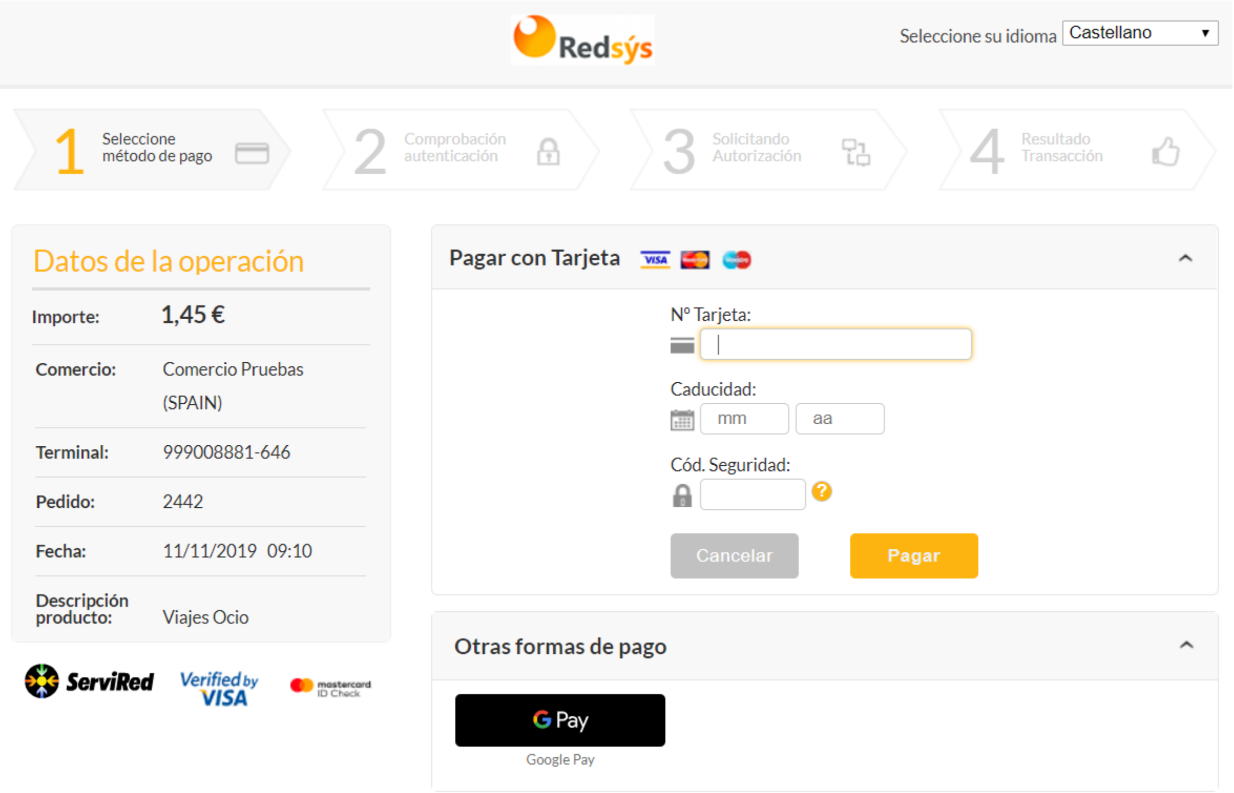
By integrating and making use of Google Pay, the merchant accepts Google 's terms of use.

** This integration can only be conducted in the production environment of the Virtual POS.

2. Connection by redirection

You can start accepting payments with Google Pay automatically at the Virtual POS by requesting the activation of the payment method from your bank. **No technical development is necessary.**

The payment method will be selectable in the Virtual POS interface:



Seleccione su idioma Castellano

1 Seleccione método de pago
2 Comprobación autenticación
3 Solicitando Autorización
4 Resultado Transacción

Datos de la operación

Importe:	1,45 €
Comercio:	Comercio Pruebas (SPAIN)
Terminal:	999008881-646
Pedido:	2442
Fecha:	11/11/2019 09:10
Descripción producto:	Viajes Ocio

Pagar con Tarjeta

Nº Tarjeta:

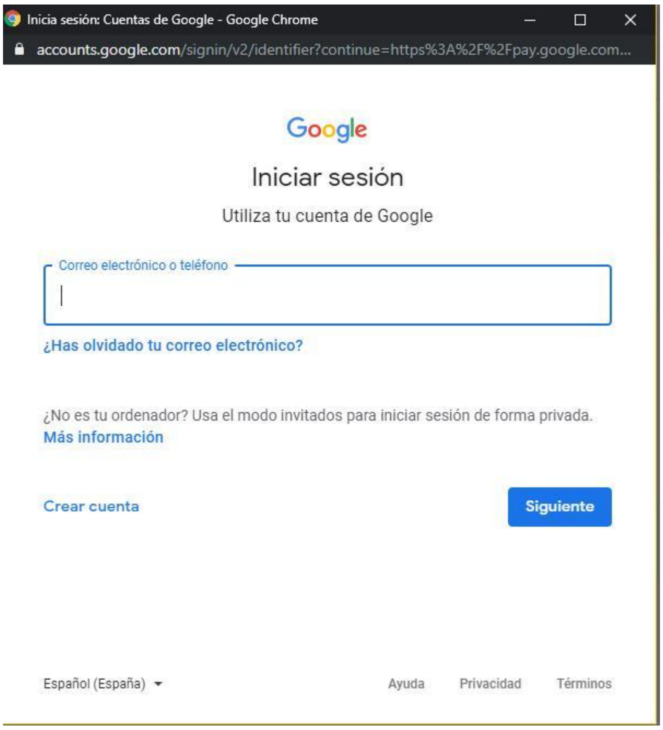
Caducidad: mm aa

Cód. Seguridad:

Cancelar Pagar

Otras formas de pago

By clicking on it, the login will be requested in the Google account (if necessary) and the means of payment with which to make the payment will be selected:



Inicia sesión: Cuentas de Google - Google Chrome

accounts.google.com/signin/v2/identifier?continue=https%3A%2F%2Fpay.google.com...

Google

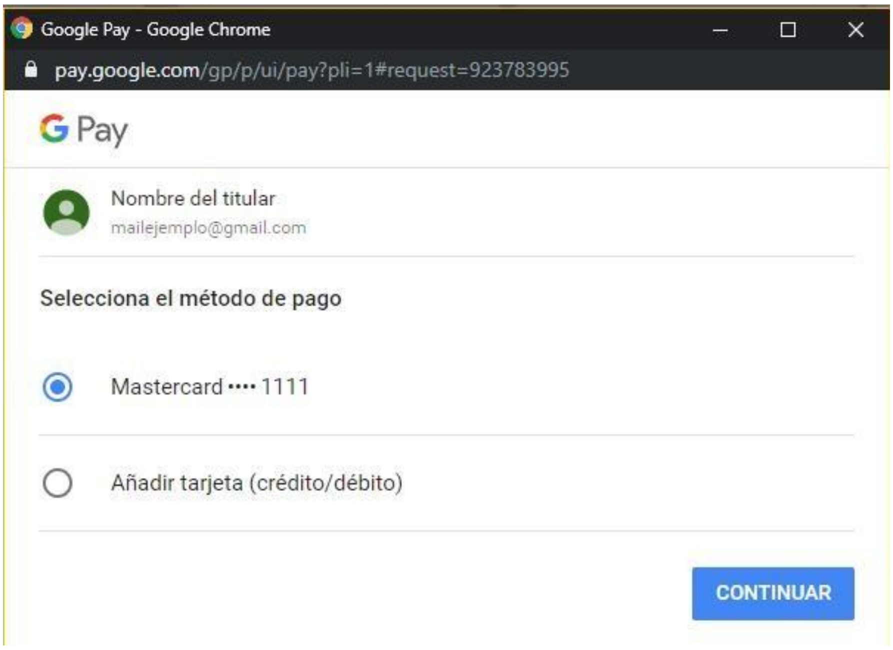
Iniciar sesión

Utiliza tu cuenta de Google

Correo electrónico o teléfono

¿Has olvidado tu correo electrónico?

¿No es tu ordenador? Usa el modo invitados para iniciar sesión de forma privada. Más información



Google Pay - Google Chrome

pay.google.com/gp/p/ui/pay?pli=1#request=923783995

G Pay

Nombre del titular
mailejemplo@gmail.com

Selecciona el método de pago

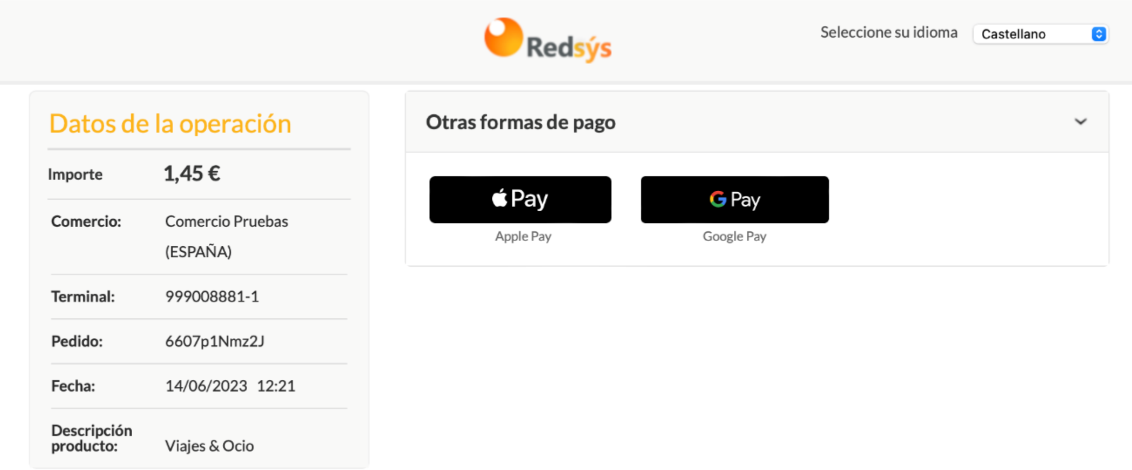
● Mastercard •••• 1111

○ Añadir tarjeta (crédito/débito)

Once selected, the payment will be processed automatically.

If you want to force the execution of the Google Pay payment method (instead of displaying the card data entry page), you must send the value 'google' (without quotes) in the DS_MERCHANT_PAYMETHODS field.

If you also have Apple Pay configured and the device allows it, if you want to force the execution of only one of these payment methods, you must send the value 'xpay' (without quotes) in the DS_MERCHANT_PAYMETHODS field. And then the owner will click on Google Pay or Apple Pay.



Datos de la operación

Importe	1,45 €
Comercio:	Comercio Pruebas (ESPAÑA)
Terminal:	999008881-1
Pedido:	6607p1Nmz2J
Fecha:	14/06/2023 12:21
Descripción producto:	Viajes & Ocio

3. Direct integration by the Merchant

To integrate Google Pay into your app or website, you must follow the Google Pay integration guide.
Google Pay Tutorial Guide
Google Pay Reference Guide

Including, as mandatory parameterization, the following data:

```
const tokenizationSpecification = {
   type: 'PAYMENT_GATEWAY',
   parameters: {
     'gateway': 'redsys',
     'gatewayMerchantId': 'FUC proporcionado por la entidad'
   }
};
```

```
allowedCardNetworks = ["AMEX", "DISCOVER", "JCB", "MASTERCARD",
"VISA"];
allowedCardAuthMethods = ["PAN_ONLY", "CRYPTOGRAM_3DS"];
```

```
paymentDataRequest.merchantInfo = {
 //a merchant ID provided by Google is available for a production
environment after approval by Google
 merchantId: 'XXXXXXXXXXXXXXXXXXXX',
 merchantName: 'Nombre del comercio',
 merchantOrigin: 'URL del comercio'
};
```

For more information, consult the official documentation:
 PaymentMethodTokenizationSpecification
CardParameters

Once integration into the Google Pay test environment is complete, production access should be requested. The process is as follows:

1 - Request access to production at the URL https://developers.google.com/pay/api/web/guides/test-and-deploy/request-prod-access

2 - Google will review the information sent by the merchant and may propose modifications, which in any case, will be validated in the same test environment

3 - Google will review the information provided by the merchant and in case of approval, its use in production will be allowed.

4- Upon approval, Google will provide the information for testing in the Production environment

In the authorization request to the Virtual POS it will be necessary to include the following data:

Parameter	**Value**
Ds_XPayData	Token returned by Google Pay **encoded in Base64** (paymentData.paymentMethodData.tokenizationData.token)
Ds_XPayType	Google
Ds_XPayOrigen	InApp - If the integration is on a mobile App WEB - If the integration is Web

3.1 Main errors and how to fix them.

- **SIS0499 or 9499**: Error decrypting the message. This error usually arises for one of these reasons:

1.
The Gateway parameter has not been indicated correctly.
When integrating with the merchant, the 'gateway': 'redsys' parameter and the fuc provided by the entity must be indicated correctly.
2.

Because the base64 that is sent is not in a correct format.
The base64 of the paymentData.paymentMethodData.tokenizationData.token field may be being sent incorrectly. It is recommended to do the base64 at the time the data is received from Google. If it is sent to the server and done there, some characters may be lost. An example of the format of the data to be sent is attached. It is not a valid message, it is only an example to be able to compare that what is sent to Redsys has the correct format. The message may have more characters than appear in this example.

Google message

```
{"signature":"MEYCVF312sIkd9q+JKSP2LLp57m6YYsc
Xhu5M131FnBUYmAWpwIhANDyutZiDCoJNSQTAF4K2
WHyEUt11ngqW1r4UZUT1JG4","protocolVersion":"ECv
1","signedMessage":"{\"encryptedMessage\":\"hOdy1
b/mHfrLKTn!DKQOXk15Hc4vRjElsYI5ZPN6ALpAtUgNJv
c8YK+73xn4nQGAL9VkayH1s2pOn+f+voHPYbVOK0PU
SqpxS45KJqyVOUi3jq2BEHjPyksDfGlopfCO1y0s4L6z4n
5Vp0PLBhpLdkFJlNd7D4sJIaUPOVPOFhGIK1zrL3DKDv
YnDQaJ4k6OcWi1wvLgNBSc5vMkLK68MK36nYyEi2ASF
GtNyGFOypvejBt2rX11JDHliMdKVt4JCW04znyaIMnxVL
K1NoxfJXMNT86KBM/yFc8VQ0BXoY6RA6JB1zsugzEFI4
JKLDokeUSUD0xWUTILD3oTJlYiXKkPPdIljLmuIXA1PfM
JkFvwcAS4yXBy5gCUyofg7bABNyf9T5+vnLP/y5hxryBJ
kUFVJ)k1pT21hH9RjLoKWqKdmfSdH1ND2dP3QA329q
LqCRB1hj7NghJqSqtH+ElpoFAyhTgA6i1Aw7+KLPi4Vp
12dfKGCtH2s5LWIKR+s90G6lONNEWgppOJ1256wFGH
RwxXzA43whDGF3WQhBdMvtD2KzbiJeqFS3AhWmYZS
i4PBMZs4oNBDRww0k\\u003d\",\"ephemeralPublicKe
y\":\"JKPiLksnpT+dL3r9bKFFvS123g+/N9Gh4dHhqSfk
3iddfhTWeu9T+LKPtecNaL3DWmYLDF80LwWp9fmk23
QCUFsdl\\u003d\",\"tag\":\"LF5pk3Sdsbd3sGldu7+rK
LXIGK0IDngJ4lnL92wDJOl\\u003d\"}"}
```

Base64 to send to Redsys

```
eyJzaWduYXR1cmUiOiJNRVlDVkYzMTJzSWtkOXErSktTUDJMTHA1N20
2WVlzY1hodTVNMTMxRm5CVVltQVdwd0loQU5EeXV0WmlEQ29KTlNR
VEFGNEsyV0h5RVV0MTFuZ3FXMXI0VVpVVDFKRzQiLCJwcm90b2Nvb
FZlcnNpb24iOiJFQ3YxIiwic2lnbmVkTWVzc2FnZSI6IntcImVuY3J5cHRl
ZE1lc3NhZ2VcIjpcImhPZHkxYi9tSGZyTEtUbiFES1FPWGsxNUhjNHZSa
kVsc1lJNVpQTjZBTHBBdFVnTkp2YzhZSys3M3huNG5RR0FMOVZrYXlI
MXMycE9uK2Yrdm9IUFliVk9LMFBVU3FweFM0NUtKcXlWT1VpM2pxMk
JFSGpQeWtzRGZHbG9wZkNPMXkwczRMNno0bjVWcDBQTEJocExka0
ZKbE5kN0Q0c0pJYVVQT1ZQT0ZoR0lLMXpyTDNES0R2WW5EUWFKN
Gs2T2NXaTF3dkxnTkJTYzV2TWtMSzY4TUszNm5ZeUVpMkFTRkd0Tnl
HRk95cHZlakJ0MnJYMTFKREhsaU1kS1Z0NEpDVzA0em55YUlNbnhWT
EsxTm94ZkpYTU5UODZLQk0veUZjOFZRMEJYb1k2UkE2SkIxenN1Z3p
FRkk0SktMRG9rZVVTVUQweFdVVElMRDNvVEpsWWlYS2tQUGRJbGpM
bXVJWEExUGZNSmtGdndjQVM0eVhCeTVnQ1V5b2ZnN2JBQk55ZjlUN
St2bkxQL3k1aHhyeUJKa1VGVkopazFwVDIxaEg5UmpMb0tXcUtkbWZ
TZEgxTkQyZFAzUUEzMjlxTHFDUkIxaGo3TmdoSnFTcXRIK0VscG9GQ
XloVGdBNmkxQXc3K0tMUGk0VnAxMmRmS0dDdEgyczVMV0lLUitzOT
BHNmxPTk5FV2dwcE9KMTI1NndGR0hSd3hYekE0M3doREdGM1dRaEJ
kTXZ0RDJLemJpSmVxRlMzQWhXbVlaU2k0UEJNWnM0b05CRFJ3dzBr
XFx1MDAzZFwiLFwiZXBoZW1lcmFsUHVibGljS2V5XCI6XCJKS1BpTGtz
bnBUK2RMM3I5YktGRnZTMTIzZysvTjlHaDRkSGhxU2ZrM2lkZGZoVFd
ldTlUK0xLUHRlY05hTDNEV21ZTERGODBMd1dwOWZtazIzUUNVRnNk
bFxcdTAwM2RcIixcInRhZ1wiOlwiTEY1cGszU2RzYmQzc0dsZHU3K3JLT
FhJR0swSURuZ0o0bG5MOTJ3REpPbFxcdTAwM2RcIn0ifQo=
```

- **9998**: Operation waiting for data request

This error can arise if Google operations need to authenticate. At Google there are 2 types of operations, tokenized operations, which are launched from a mobile device and do not require direct authentication from the client, since the mobile device and Google are in charge of doing the authentication. And untokenized card operations (These operations arise when the client is on a computer with a web browser). In this second type of operations, Google encrypts the card in the message that the merchant subsequently sends to Redsys. This operation must be authenticated. If the merchant cannot authenticate (for example, the impossibility of initiatingPetition-trataPetition in this type of operations), it is recommended to send to the redirection url (https://sis.redsys.es/ sis/performPayment) the operations so that Redsys is responsible for the authentication of the owner.

4. Advanced feature: Sending decrypted data by the Merchant*.

* Advanced functionality with implications for PCI-DSS compliance.

The merchant in this case will decode the data, obtaining an object with the following format (example):

https://developers.google.com/pay/api/web/guides/resources/sample-tokens?hl=es

```
{
    "gatewayMerchantId": "some-merchant-id",
"messageExpiration": "1561533871082",
    "messageId": "AH2Ejtc8qBlP_MCAV0jJG7ErQKeDrkEUt…",
    "paymentMethod": " TOKENIZED_CARD ",
    "paymentMethodDetails": {
        "expirationYear": 2026,
        "expirationMonth": 12,
        "pan": "489537******3478",
        "authMethod": "CRYPTOGRAM_3DS",
        "eciIndicator": "05",
        "cryptogram": "AgAAAAAABk4DWZ4C28yUQAAAAAA="
    }
}
```

The necessary data will be sent to the Virtual POS in the parameter Ds_**XPayDecodedData**, in Json format:

```
{
    "cryptogram": "AgAAAAAABk4DWZ4C28yUQAAAAAA=",
    "eciInd": "05",
    "expirationDate":"2612",
    "token": "489537******3478",
    "paymentMethod": "TOKENIZED_CARD"
}
```

cryptogram = Reference to the cryptogram. Optional, required when the card is tokenized.

eciInd = Reference to the eciIndicator.Optional, it will be sent when received, in a tokenized operation expirationDate = Expiration of the card in **YYMM format**.

token = Reference to the pan field.

paymentMethod= It can only be "CARD" and "TOKENIZED_CARD". The mode in which the card travels is indicated here. If there is a cryptogram in the Google message(ECv1 → 3dsCryptogram o ECv2 → cryptogram), the card will be tokenized. For more information, visit the official Google guide on the different ways of receiving data.

In the authorisation transaction sent to the Virtual POS, the following additional parameters must be added:

Parameter	**Value**
Ds_XPayDecodedData	JSON object with the information obtained from Google
Ds_XPayType	Google
Ds_XPayOrigen	InApp - If the integration is on a mobile App WEB - If the integration is Web

Below is an example of a request to the Virtual POS with tokenized card.

```
<DATOSENTRADA>
<DS_MERCHANT_AMOUNT>42</DS_MERCHANT_AMOUNT>
<DS_MERCHANT_ORDER>813734</DS_MERCHANT_ORDER>
<DS_MERCHANT_MERCHANTCODE>999008881</DS_MERCHANT_MERCHANTCODE>
<DS_MERCHANT_CURRENCY>978</DS_MERCHANT_CURRENCY>
<DS_MERCHANT_TRANSACTIONTYPE>0</DS_MERCHANT_TRANSACTIONTYPE>
<DS_MERCHANT_TERMINAL>871</DS_MERCHANT_TERMINAL>
<DS_XPAYDECODEDDATA>
{
    "cryptogram": "AgAAAAAABk4DWZ4C28yUQAAAAAA=",
    "eciInd": "05",
    "expirationDate":"2612",
    "token": "489537******3478",
    "paymentMethod": "TOKENIZED_CARD",
}
</DS_XPAYDECODEDDATA>
<DS_XPAYTYPE>Google</DS_XPAYTYPE>
<DS_XPAYORIGEN>InApp</DS_XPAYORIGEN>
</DATOSENTRADA>
```

Example of a request to the Virtual POS with Non-tokenized card.

```
<DATOSENTRADA>
<DS_MERCHANT_AMOUNT>42</DS_MERCHANT_AMOUNT>
<DS_MERCHANT_ORDER>813734</DS_MERCHANT_ORDER>
<DS_MERCHANT_MERCHANTCODE>999008881</DS_MERCHANT_MERCHANTCODE>
<DS_MERCHANT_CURRENCY>978</DS_MERCHANT_CURRENCY>
<DS_MERCHANT_TRANSACTIONTYPE>0</DS_MERCHANT_TRANSACTIONTYPE>
<DS_MERCHANT_TERMINAL>871</DS_MERCHANT_TERMINAL>
<DS_XPAYDECODEDDATA>
{
    "expirationDate":"2511",
    "token": "489537******4315",
    "paymentMethod": "CARD",
}
</DS_XPAYDECODEDDATA>
<DS_XPAYTYPE>Google</DS_XPAYTYPE>
<DS_XPAYORIGEN>InApp</DS_XPAYORIGEN>
</DATOSENTRADA>
```