

Documento de Integración

Librería Android "Pago App"

Índice

1. Introducción
2. Objetivo
3. Beneficios de la librería de VisaNet
4. Definiciones
5. Integración
 - a. Consideraciones
 - b. Implementación
 - i. Registrar la librería.
 - ii. Configurar la librería.
 - iii. Invocar la librería.
 - iv. Respuesta de la librería.
 - v. Comportamiento de la librería.
 - vi. Pruebas, Certificación y Producción.
6. Anexos
 - a. Carga de logo del comercio en la librería
 - b. Códigos de acción (denegaciones)
 - c. Diccionario de atributos enviados a la librería
 - d. Diccionario de atributos devueltos por la librería
 - e. Transición de estados para una venta
 - f. API de autorización de venta
 - g. API de anulación de venta
 - h. API de confirmación de venta
 - i. API de consulta de venta
 - j. API de antifraude para la venta

1. Introducción

El proceso de afiliación a Pago App tiene una serie de etapas por las cuales debe pasar todo comercio, en el orden indicado, sin excepción alguna y cumpliendo con los requisitos que conlleva cada una de ellas.

El proceso es el siguiente:

- a) Creación de código VisaNet
- b) Integración
- c) Certificación
- d) Pase a producción

2. Objetivo

Este documento tiene como objetivo dar lineamientos generales, a los comercios y/o a los proveedores encargados del desarrollo, para la integración y adecuación de las aplicaciones Android de los comercios afiliados al servicio de Pago App.

3. Beneficios de la librería de VisaNet

La librería de VisaNet simplifica y asegura los pagos online con los siguientes beneficios:

- Personalizable con el logo (imagen y texto) y color del botón “Pagar”
- Permite la tokenización de tarjetas para transacciones con integración a plataforma Pago Programado VisaNet
- Integración a plataforma Tele Pago VisaNet
- Adaptable en Tablet y Móviles

El comercio podrá integrar rápidamente esta librería a su tienda virtual y brindar una experiencia fácil, rápida y segura a sus clientes.

4. Definiciones

- **Integración:** Proceso de pruebas y adecuaciones para poder realizar ventas por aplicativos móviles utilizando la librería de VisaNet Perú y cumplir con los requerimientos especificados en este documento.
- **Formulario de Pago:** Formulario donde el tarjetahabiente ingresa los datos de su tarjeta para completar la transacción electrónica.
- **Tarjetahabiente:** Persona titular de una tarjeta, en este caso, de la marca VISA.
- **App:** Es un aplicativo informático diseñado para ser ejecutado en teléfonos inteligentes, tabletas y otros dispositivos móviles.
- **TLS (Transport Layer Security):** Es un protocolo de seguridad que utilizan los navegadores web y los servidores web para ayudar a los usuarios a proteger la transferencia de sus datos. Se debe considerar utilizar la versión de TLS 1.2

5. Integración

El proceso de integración consta de cuatro pasos:

1. Registrar la librería.
2. Configurar la librería.
3. Invocar la librería.
4. Respuesta de la librería.

a. Consideraciones

- El comercio afiliado recibirá un **Usuario** y una **Contraseña** para poder invocar a la librería.
- La librería funciona desde la distribución **API 19: Android 4.4 (KitKat)**.

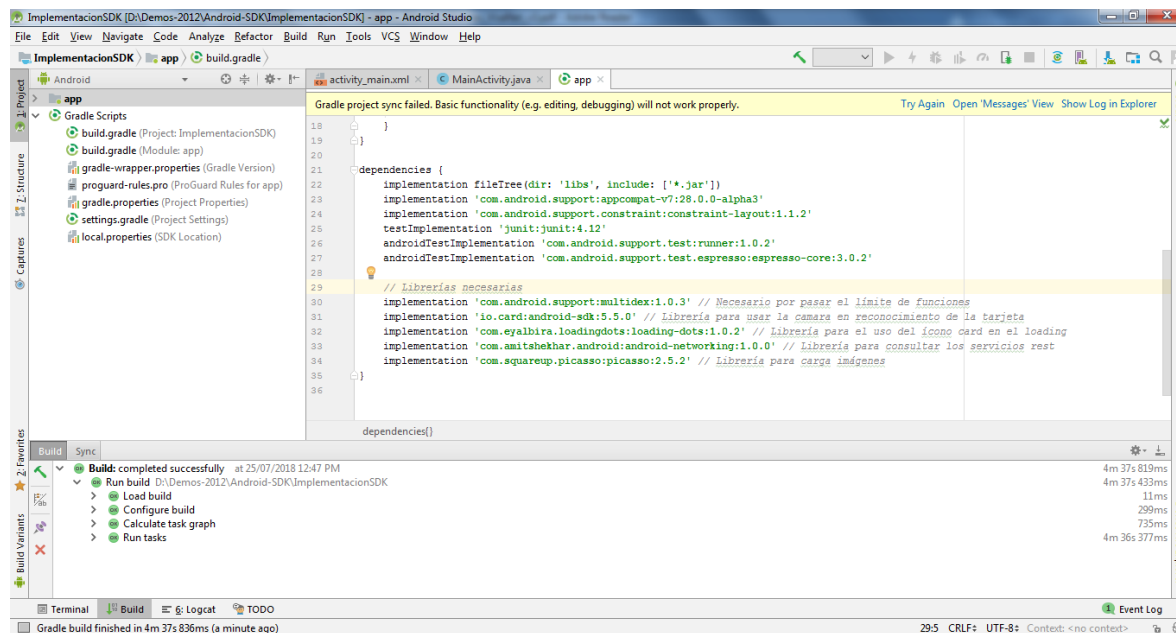
b. Implementación

i. Paso 1: Registrar la librería

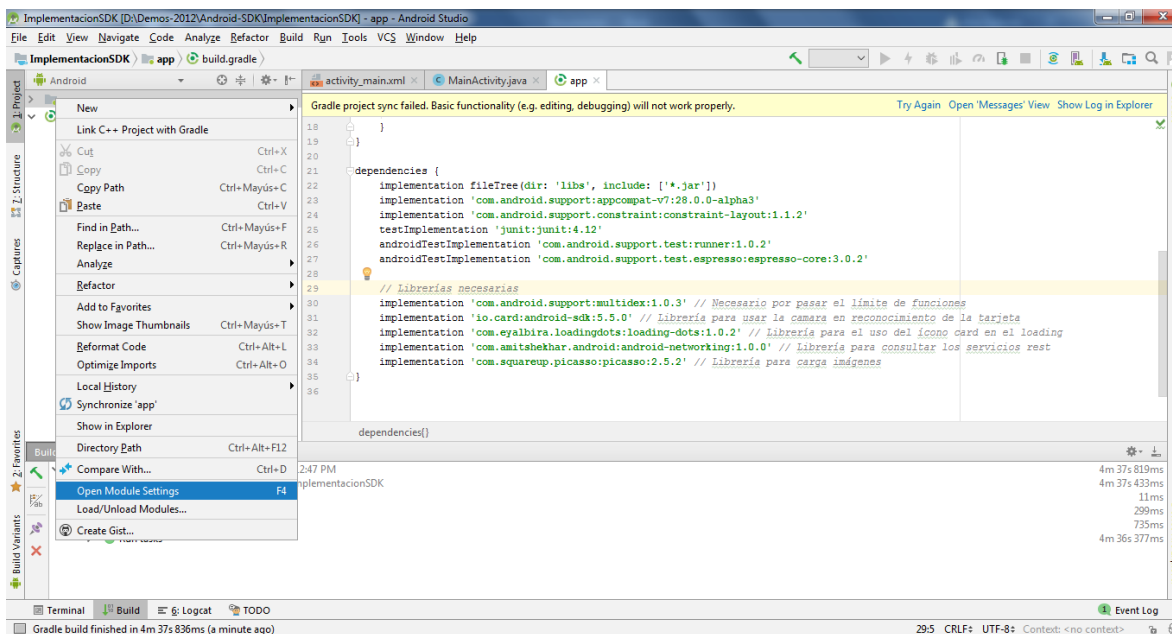
Previo a iniciar el formulario es necesario verificar el acceso, para esto se debe invocar al API de seguridad de la siguiente manera:

1: Dentro del archivo “build.gradle” en la sección “dependencies” agregar las siguientes referencias:

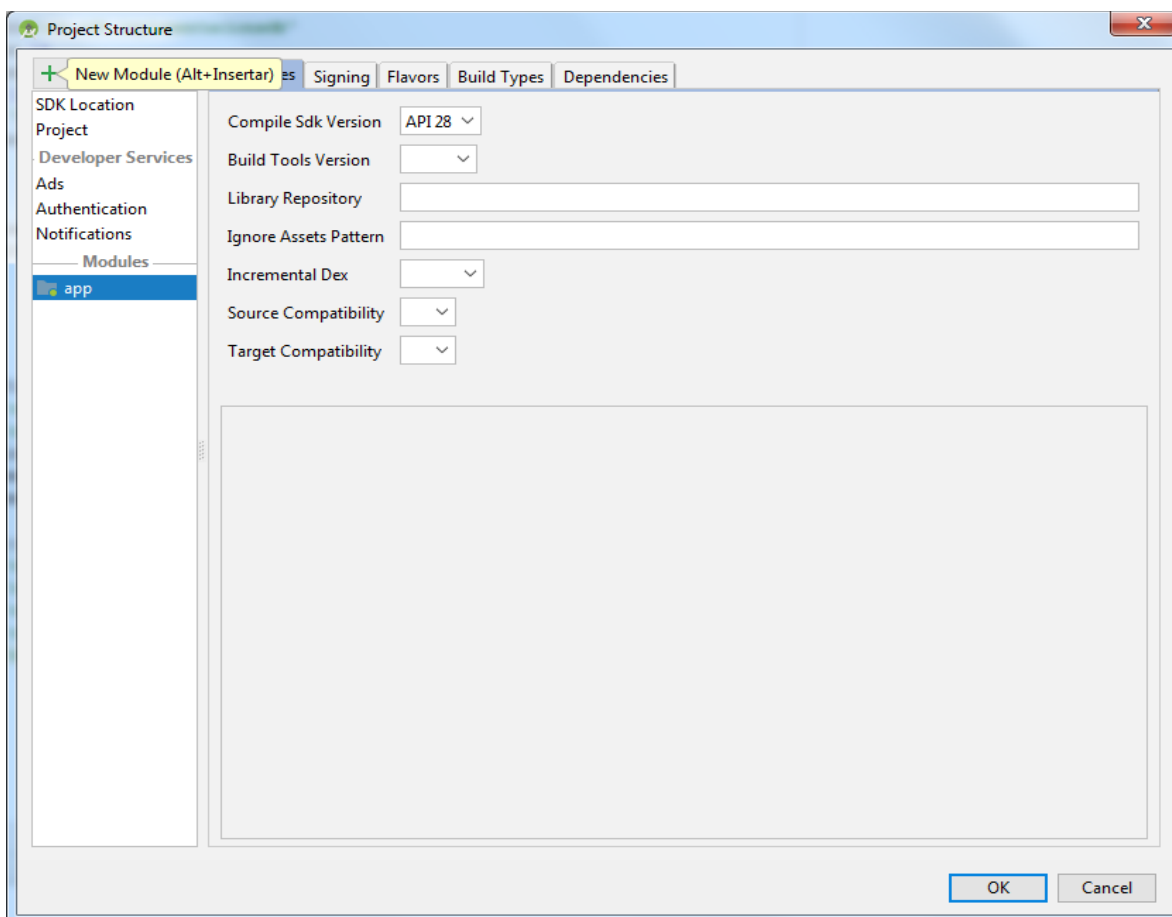
```
implementation 'com.android.support:multidex:1.0.3'
implementation 'io.card:android-sdk:5.5.0'
implementation 'com.eyalbira.loadingdots:loading-dots:1.0.2'
implementation 'com.amitshekhar.android:android-networking:1.0.0'
implementation 'com.squareup.picasso:picasso:2.5.2'
```



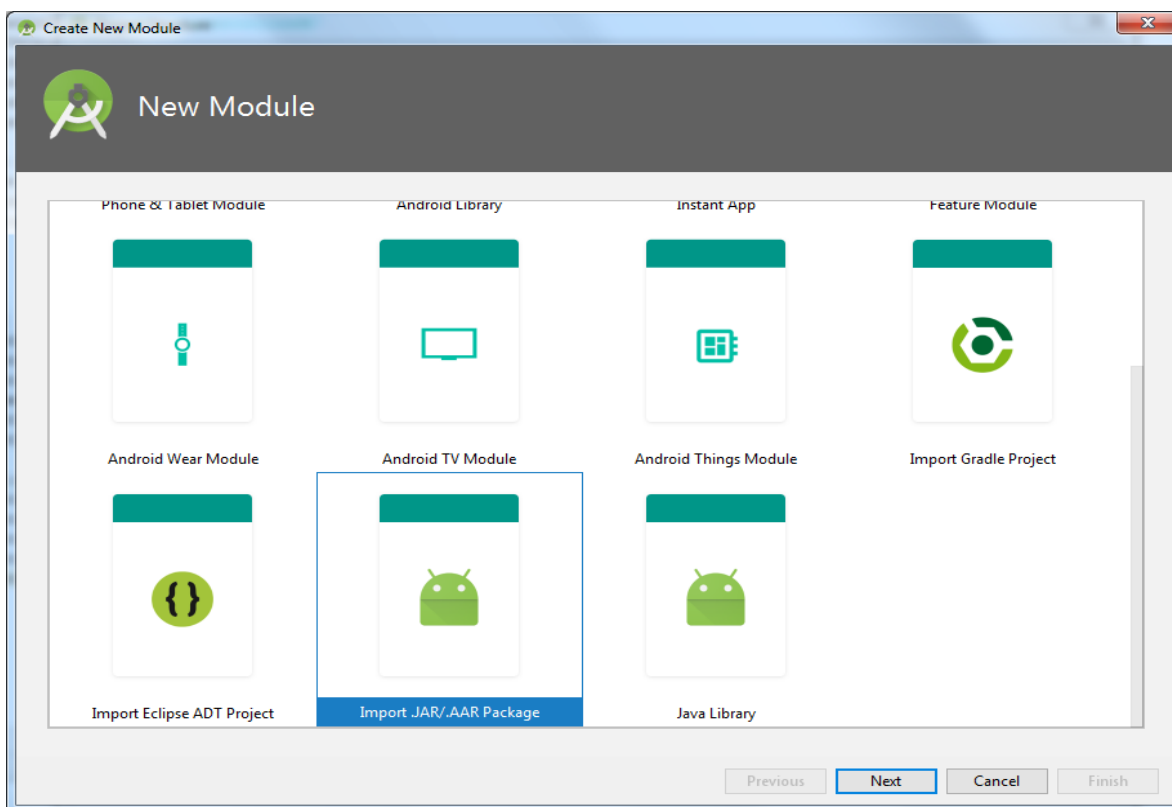
2: Clic derecho en el directorio “app” y seleccionar la opción “Open Module Settings”



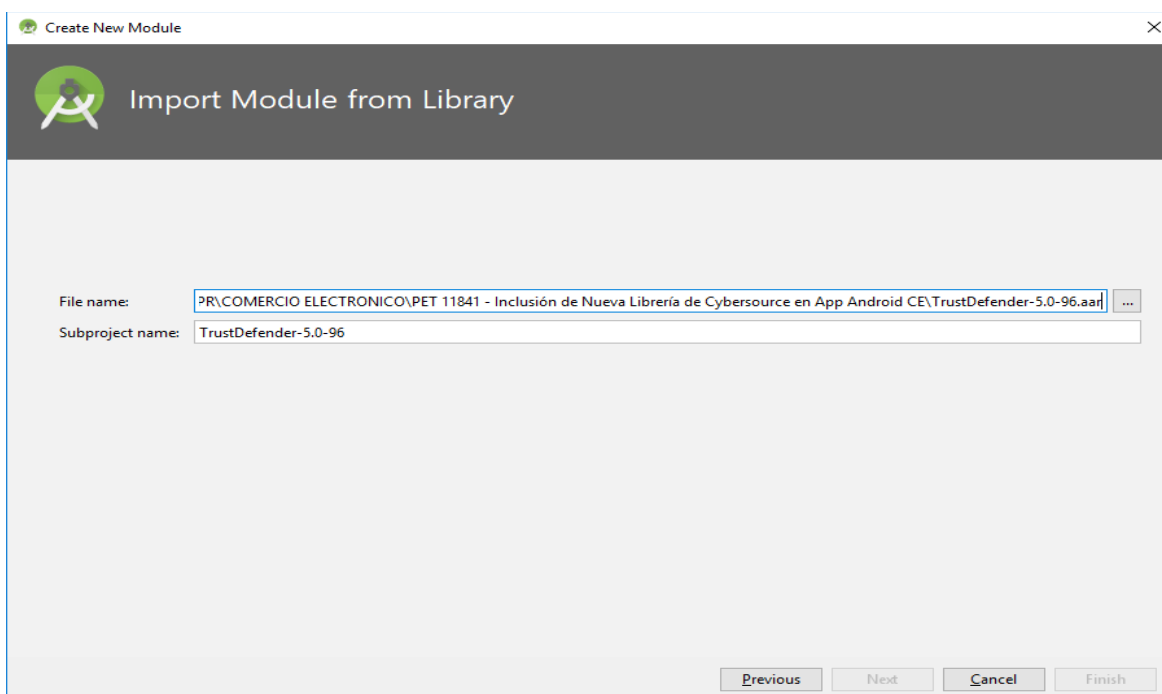
3: En la ventana que aparece, clic en el botón “+” (New Module)



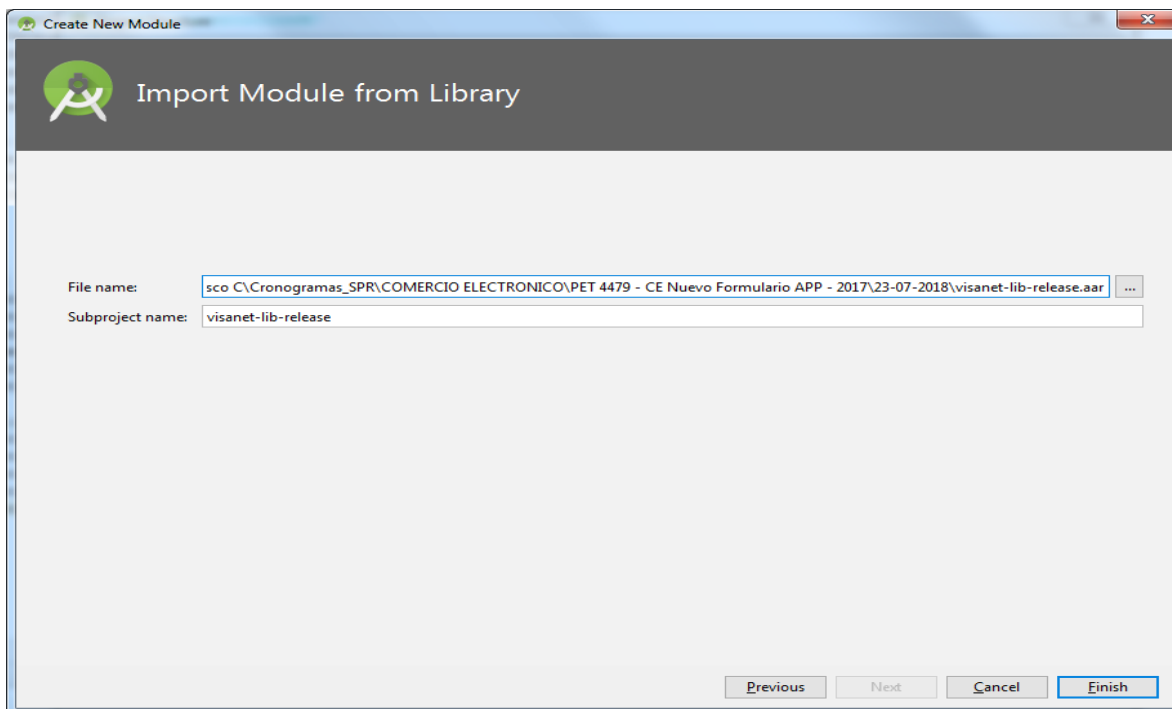
4: A continuación, seleccionar la opción “Import .JAR/ .ARR Package” y clic en el botón “Next”



5: Seleccionar la librería ***“TrustDefender-5.0-96.aar”*** a importar y finalmente dar clic en el botón ***“Finish”***



6: Seleccionar la librería ***“visanet-lib-release.aar”*** a importar y finalmente dar clic en el botón ***“Finish”***

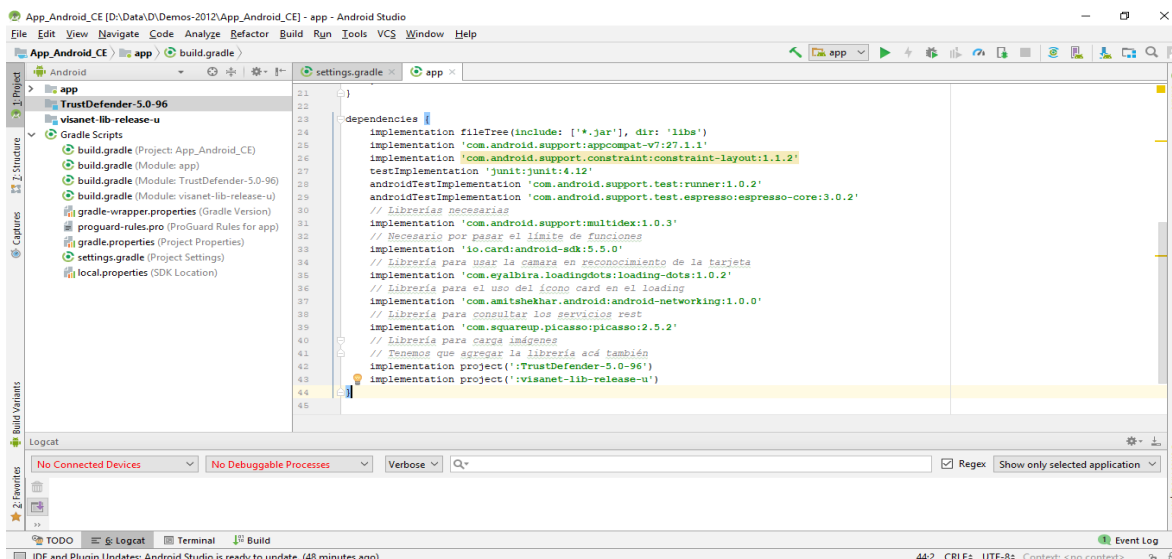


7: Verificar que las referencias a las librerías se hayan agregado a los archivos “**build.gradle**” y “**settings.gradle**” como se muestra a continuación. De no existir las referencias, estas se deben ingresar manualmente en cada archivo.

En archivo: **build.gradle**

```
implementation project(':TrustDefender-5.0-96')

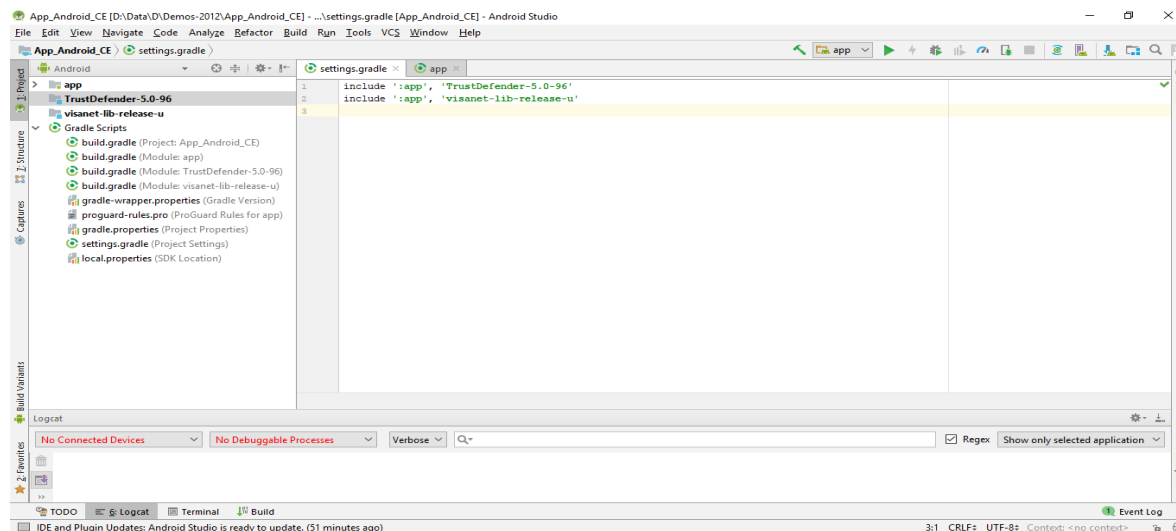
implementation project(':visanet-lib-release')
```



En archivo: **settings.gradle**

```
include ':app', 'TrustDefender-5.0-96'
```

```
include ':app', 'visanet-lib-release'
```



ii. Paso 2: Configurar la librería

Para inyectar el formulario de pago se requieren múltiples parámetros los que se explican a continuación:

Formulario de pago regular

```
Map<String, Object> data = new HashMap<>();

data.put(VisaNet.VISANET_CHANNEL,
Channel.MOBILE);

data.put(VisaNet.VISANET_COUNTABLE, true);

data.put(VisaNet.VISANET_USERNAME,
"integracion@gmail.com");

data.put(VisaNet.VISANET_PASSWORD,
"A55$HgT1");

data.put(VisaNet.VISANET_MERCHANT,
"341198210");

data.put(VisaNet.VISANET_PURCHASE_NUMBER,
"1790");

data.put(VisaNet.VISANET_AMOUNT, "15.22");
```

Formulario de pago con recordar tarjeta


```
Map<String, Object> data = new HashMap<> ();

data.put (VisaNet.VISANET_CHANNEL,
Channel.MOBILE);

data.put (VisaNet.VISANET_COUNTABLE, true);

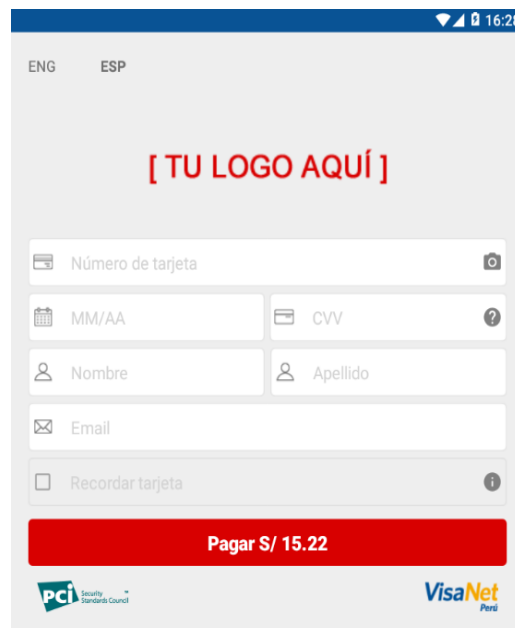
data.put (VisaNet.VISANET_USERNAME,
"integracion@gmail.com");

data.put (VisaNet.VISANET_PASSWORD,
"Av5$HgT1");

data.put (VisaNet.VISANET_MERCHANT,
"341198210");

data.put (VisaNet.VISANET_PURCHASE_NUMBER,
"1790");

data.put (VisaNet.VISANET_AMOUNT, "15.22");
```



Formulario de pago (recordar tarjeta + ocultando datos del cliente)

```
Map<String, Object> data = new HashMap<> ();

data.put (VisaNet.VISANET_CHANNEL,
Channel.MOBILE);

data.put (VisaNet.VISANET_COUNTABLE, true);

data.put (VisaNet.VISANET_USERNAME,
"integracion@gmail.com");

data.put (VisaNet.VISANET_PASSWORD, "Av5$HgT1");

data.put (VisaNet.VISANET_MERCHANT, "341198210");


data.put (VisaNet.VISANET_PURCHASE_NUMBER,
"1790");

data.put (VisaNet.VISANET_AMOUNT, "15.22");

data.put (VisaNet.VISANET_REGISTER_NAME, "Juan");

data.put (VisaNet.VISANET_REGISTER_LASTNAME,
"Perez");

data.put (VisaNet.VISANET_REGISTER_EMAIL,
"jperez@test.com");
```



Formulario de pago con tarjeta recordada

```
Map<String, Object> data = new HashMap<>();

data.put(VisaNet.VISANET_CHANNEL,
Channel.MOBILE);

data.put(VisaNet.VISANET_COUNTABLE, true);

data.put(VisaNet.VISANET_USERNAME,
"integracion@gmail.com");

data.put(VisaNet.VISANET_PASSWORD,
"Av5$$HgT1");

data.put(VisaNet.VISANET_MERCHANT,
"341198210");

data.put(VisaNet.VISANET_PURCHASE_NUMBER,
"1790");

data.put(VisaNet.VISANET_AMOUNT, "15.22");

data.put(VisaNet.VISANET_USER_TOKEN,
"demo@gmail.com");
```



Formulario de pago mostrando cuotas

```
Map<String, Object> data = new
HashMap<>();

data.put(VisaNet.VISANET_CHANNEL,
Channel.MOBILE);

data.put(VisaNet.VISANET_COUNTABLE,
true);

data.put(VisaNet.VISANET_USERNAME,
"integracion@gmail.com");

data.put(VisaNet.VISANET_PASSWORD,
"Av5$$HgT1");

data.put(VisaNet.VISANET_MERCHANT,
"341198210");

data.put(VisaNet.VISANET_PURCHASE_NUMBER,
"1790");

data.put(VisaNet.VISANET_AMOUNT,
"15.22");
```



Formulario de pago con tarjeta foránea

```
Map<String, Object> data = new
HashMap<>();

data.put(VisaNet.VISANET_CHANNEL,
Channel.MOBILE);

data.put(VisaNet.VISANET_COUNTABLE, true);

data.put(VisaNet.VISANET_USERNAME,
"integracion@gmail.com");

data.put(VisaNet.VISANET_PASSWORD,
"Av5$HgT1");

data.put(VisaNet.VISANET_MERCHANT,
"341198210");

data.put(VisaNet.VISANET_PURCHASE_NUMBER,
"1790");

data.put(VisaNet.VISANET_AMOUNT, "15.22");
```

The screenshot shows a mobile payment interface for VisaNet Perú. At the top, there are language options 'ENG' and 'ESP'. Below them is a placeholder for a logo: '[TU LOGO AQUÍ]'. The form contains the following fields:

- Card number: 4242-42
- Expiration date: MM/AA
- CVV: CVV
- Name: Nombre
- Address: Apellido
- Email: Email
- Address of the card: Dirección de facturación de la tarjeta:
 - City: Ciudad
 - Country: País
- Remember card: Recordar tarjeta (checkbox)

 At the bottom, there is a large red button labeled 'Pagar S/ 15.22'. Logos for PCI Security Standards Council and VisaNet Perú are visible at the very bottom.

Formulario de pago ocultando en monto en “Pagar”

```
Map<String, Object> data = new HashMap<>();

data.put(VisaNet.VISANET_CHANNEL,
Channel.MOBILE);

data.put(VisaNet.VISANET_COUNTABLE, true);

data.put(VisaNet.VISANET_USERNAME,
"integracion@gmail.com");

data.put(VisaNet.VISANET_PASSWORD, "Av5$HgT1");

data.put(VisaNet.VISANET_MERCHANT, "341198210");

data.put(VisaNet.VISANET_PURCHASE_NUMBER,
"1790");

data.put(VisaNet.VISANET_AMOUNT, "15.22");

data.put(VisaNet.VISANET_SHOW_AMOUNT, false);
```

This screenshot is similar to the previous one, showing the same payment form. However, the red button at the bottom is labeled 'Pagar' instead of 'Pagar S/ 15.22', indicating that the amount is hidden. All other fields and UI elements remain the same.

Formulario de pago con recurrencia fija

```
Map<String, Object> data = new HashMap<>();

data.put(VisaNet.VISANET_CHANNEL, Channel.MOBILE);

data.put(VisaNet.VISANET_COUNTABLE, true);

data.put(VisaNet.VISANET_USERNAME,
"integracion@gmail.com");

data.put(VisaNet.VISANET_PASSWORD, "Av5$HgT1");

data.put(VisaNet.VISANET_MERCHANT, "341198210");

data.put(VisaNet.VISANET_PURCHASE_NUMBER, "1790");

data.put(VisaNet.VISANET_AMOUNT, "1.00");

data.put(VisaNet.VISANET_RECURRENCE, true);

data.put(VisaNet.VISANET_RECURRENCE_TYPE,
VisaNet.RecurrenceType.FIXED);

data.put(VisaNet.VISANET_RECURRENCE_FREQUENCY,
VisaNet.RecurrenceFrequency.MONTHLY);

data.put(VisaNet.VISANET_RECURRENCE_MAX_AMOUNT,
"12.23");

data.put(VisaNet.VISANET_RECURRENCE_AMOUNT,
"2.25");

data.put(VisaNet.VISANET_RECURRENCE_DOCUMENT_TYPE,
VisaNet.DocumentType.DNI);

data.put(VisaNet.VISANET_RECURRENCE_DOCUMENT_ID,
"40525847");

data.put(VisaNet.VISANET_RECURRENCE_PRODUCT_ID,
"321");

data.put(VisaNet.VISANET_RECURRENCE_BENEFICIARY_ID,
"ABCXY123");
```



The screenshot shows the VisaNet Perú mobile app interface for a recurring payment. At the top, there are language options (ENG, ESP) and a status bar showing 17:51. Below the language options is a large red box with the text "[TU LOGO AQUÍ]". The form contains several input fields: "Número de tarjeta" (card number), "MM/AA" (expiration date), "CVV" (card verification number), "Nombre" (first name), "Apellido" (last name), "Email", "Móvil" (mobile number), "S/ 2.25" (amount), "Mensual" (frequency), and "Establecer monto máximo" (set maximum amount). Below these fields is a "Regístrate a:" label and a red "Suscribirse" (subscribe) button. At the bottom, there are logos for PCI Security Standards Council and VisaNet Perú.

Formulario de pago con recurrencia variable

```
Map<String, Object> data = new HashMap<>();

data.put(VisaNet.VISANET_CHANNEL, Channel.MOBILE);

data.put(VisaNet.VISANET_COUNTABLE, true);

data.put(VisaNet.VISANET_USERNAME,
"integracion@gmail.com");

data.put(VisaNet.VISANET_PASSWORD, "Av5$HgT1");

data.put(VisaNet.VISANET_MERCHANT, "341198210");

data.put(VisaNet.VISANET_PURCHASE_NUMBER, "1790");

data.put(VisaNet.VISANET_AMOUNT, "1.00");

data.put(VisaNet.VISANET_RECURRENCE, true);

data.put(VisaNet.VISANET_RECURRENCE_TYPE,
VisaNet.RecurrenceType.VARIABLE);

data.put(VisaNet.VISANET_RECURRENCE_FREQUENCY,
VisaNet.RecurrenceFrequency.MONTHLY);

data.put(VisaNet.VISANET_RECURRENCE_MAX_AMOUNT,
"12.23");

data.put(VisaNet.VISANET_RECURRENCE_AMOUNT, "0.00");

data.put(VisaNet.VISANET_RECURRENCE_DOCUMENT_TYPE,
VisaNet.DocumentType.DNI);

data.put(VisaNet.VISANET_RECURRENCE_DOCUMENT_ID,
"40525847");

data.put(VisaNet.VISANET_RECURRENCE_PRODUCT_ID,
"321");

data.put(VisaNet.VISANET_RECURRENCE_BENEFICIARY_ID,
"ABCXY123");

data.put(VisaNet.VISANET_RECURRENCE_BENEFICIARY_NAME,
"Julio");

data.put(VisaNet.VISANET_RECURRENCE_BENEFICIARY_LASTNAME,
"Mondragon");
```



ENG ESP

[TU LOGO AQUÍ]

Número de tarjeta

MM/AA CVV

Nombre Apellido

Email

Móvil

☐ Establecer monto máximo

Regístrate a:

Suscribirse

PCI Security Standards Council VisaNet Perú

Formulario de pago con recurrencia fija + pago inicial

```
Map<String, Object> data = new HashMap<>();

data.put(VisaNet.VISANET_CHANNEL, Channel.MOBILE);

data.put(VisaNet.VISANET_COUNTABLE, true);

data.put(VisaNet.VISANET_USERNAME,
"integracion@gmail.com");

data.put(VisaNet.VISANET_PASSWORD, "Av5$5HgT1");

data.put(VisaNet.VISANET_MERCHANT, "341198210");

data.put(VisaNet.VISANET_PURCHASE_NUMBER, "1790");

data.put(VisaNet.VISANET_AMOUNT, "5.50");

data.put(VisaNet.VISANET_RECURRENCE, true);

data.put(VisaNet.VISANET_RECURRENCE_TYPE,
VisaNet.RecurrenceType.FIXED_INITIAL);

data.put(VisaNet.VISANET_RECURRENCE_FREQUENCY,
VisaNet.RecurrenceFrequency.MONTHLY);

data.put(VisaNet.VISANET_RECURRENCE_MAX_AMOUNT,
"12.23");

data.put(VisaNet.VISANET_RECURRENCE_AMOUNT, "2.00");

data.put(VisaNet.VISANET_RECURRENCE_DOCUMENT_TYPE,
VisaNet.DocumentType.DNI);

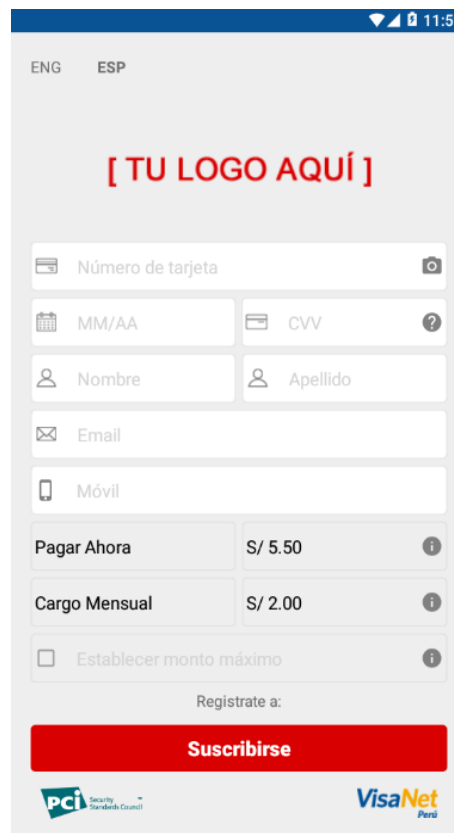
data.put(VisaNet.VISANET_RECURRENCE_DOCUMENT_ID,
"40525847");

data.put(VisaNet.VISANET_RECURRENCE_PRODUCT_ID,
"321");

data.put(VisaNet.VISANET_RECURRENCE_BENEFICIARY_ID,
"ABCXY123");

data.put(VisaNet.VISANET_RECURRENCE_BENEFICIARY_NAME,
"Julio");

data.put(VisaNet.VISANET_RECURRENCE_BENEFICIARY_LASTN
AME, "Mondagron");
```



The screenshot shows the VisaNet Perú mobile app interface. At the top, there are language options for ENG and ESP. Below this is a placeholder for a logo, labeled "[TU LOGO AQUÍ]". The form contains several input fields: "Número de tarjeta" (Card Number), "MM/AA" (Month/Year), "CVV", "Nombre" (First Name), "Apellido" (Last Name), "Email", and "Móvil" (Mobile). Below these fields, there are two rows for payment details: "Pagar Ahora" (Pay Now) with a value of S/ 5.50, and "Cargo Mensual" (Monthly Charge) with a value of S/ 2.00. There is also a checkbox labeled "Establecer monto máximo" (Set maximum amount). At the bottom, there is a red button labeled "Suscribirse" (Subscribe) and a "Regístrate a:" (Register to:) label. The VisaNet Perú logo and the PCI Security Standards Council logo are visible at the bottom of the screen.

Formulario de pago con recurrencia variable + pago inicial

```
Map<String, Object> data = new HashMap<>();

data.put(VisaNet.VISANET_CHANNEL, Channel.MOBILE);

data.put(VisaNet.VISANET_COUNTABLE, true);

data.put(VisaNet.VISANET_USERNAME,
"integracion@gmail.com");

data.put(VisaNet.VISANET_PASSWORD, "Av5$HgT1");

data.put(VisaNet.VISANET_MERCHANT, "341198210");

data.put(VisaNet.VISANET_PURCHASE_NUMBER, "1790");

data.put(VisaNet.VISANET_AMOUNT, "5.50");

data.put(VisaNet.VISANET_RECURRENCE, true);

data.put(VisaNet.VISANET_RECURRENCE_TYPE,
VisaNet.RecurrenceType.VARIABLE_INITIAL);

data.put(VisaNet.VISANET_RECURRENCE_FREQUENCY,
VisaNet.RecurrenceFrequency.MONTHLY);

data.put(VisaNet.VISANET_RECURRENCE_MAX_AMOUNT,
"12.23");

data.put(VisaNet.VISANET_RECURRENCE_AMOUNT, "0.00");

data.put(VisaNet.VISANET_RECURRENCE_DOCUMENT_TYPE,
VisaNet.DocumentType.DNI);

data.put(VisaNet.VISANET_RECURRENCE_DOCUMENT_ID,
"40525847");

data.put(VisaNet.VISANET_RECURRENCE_PRODUCT_ID,
"321");

data.put(VisaNet.VISANET_RECURRENCE_BENEFICIARY_ID,
"ABCXY123");

data.put(VisaNet.VISANET_RECURRENCE_BENEFICIARY_NAME,
"Julio");

data.put(VisaNet.VISANET_RECURRENCE_BENEFICIARY_LASTNAME,
"AME", "Mondagron");
```



The screenshot shows the VisaNet Perú mobile app interface. At the top, there are language options for ENG and ESP. Below this is a placeholder for a logo, labeled "[TU LOGO AQUÍ]". The form contains several input fields: "Número de tarjeta" (Card number), "MM/AA" (Expiration date), "CVV", "Nombre" (First name), "Apellido" (Last name), "Email", and "Móvil" (Mobile number). There is a section for "Pagar Ahora" (Pay now) with a value of "S/ 5.50" and an option to "Establecer monto máximo" (Set maximum amount). A red button labeled "Suscribirse" (Subscribe) is prominently displayed. At the bottom, there are logos for PCI Security Standards Council and VisaNet Perú.

Formulario de pago con PagoEfectivo

Estos son los campos mínimos a mandar, adicionales al formulario de pago elegido arriba (esto para cuando el Pago Habiente seleccione Pago con tarjeta)

```
Map<String, Object> data = new HashMap<>();

data.put(VisaNet.VISANET_CHANNEL,
Channel.MOBILE);

data.put(VisaNet.VISANET_COUNTABLE, true);

data.put(VisaNet.VISANET_USERNAME,
"integracion@gmail.com");

data.put(VisaNet.VISANET_PASSWORD, "Av5$HgT1");

data.put(VisaNet.VISANET_MERCHANT, "341198210");

data.put(VisaNet.VISANET_PURCHASE_NUMBER,
"1790");

data.put(VisaNet.VISANET_AMOUNT, "12.90");
```



ENG ESP Cancelar

[TU LOGO AQUÍ]

¿Cómo realizar tu compra con PagoEfectivo?
Luego de haber dado click en pagar, recibirás un e-mail con el código CIP de PagoEfectivo que tendrá validez solo por 48 horas.

✓ Depósitos en efectivo vía PagoEfectivo - Paga en BBVA, BCP, Interbank, Scotiabank, BanBif, Western Union, Tambo+, Kasnet, Full Carga, Red Digital, Money Gram, Caja Arequipa, Disashop, en cualquier agente o agencia autorizada a nivel nacional a la cuenta de PagoEfectivo.

✓ Transferencias bancarias vía PagoEfectivo - Paga en BBVA, BCP, Interbank, Scotiabank, BanBif, Caja Arequipa, a través de la banca por internet o banca móvil en la opción pago de servicios.

BBVA Continental BCP Interbank Scotiabank
BanBif WesternUnion TAMBO+ KASNET
MoneyGram Arequipa Disashop Full Carga Red Digital

Ingrese correo para recibir el código:

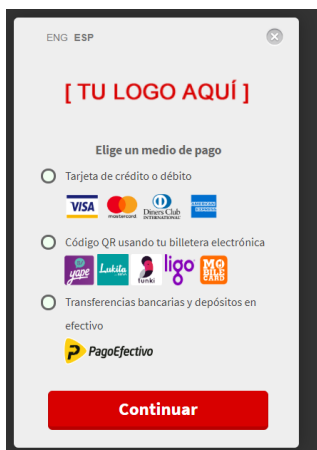
✉ Correo

Pagar S/ 12.59

 PagoEfectivo

Nota:

Si el comercio tiene habilitada la opción de PagoEfectivo se mostrará esta pantalla inicial para seleccionar el medio de pago antes de todos los formularios anteriores.



Configuraciones finales (Registro de MDDs opcionales + endpoint + personalización)

```
//Registro de MDDs complementarios (Ingreso opcional)
HashMap<String, String> MDDdata = new HashMap<String, String>();
MDDdata.put("19", "LIM");
MDDdata.put("20", "AQP");
MDDdata.put("21", "AFKI345");
MDDdata.put("94", "ABC123DEF");

data.put(VisaNet.VISANET_MDD, MDDdata);

//Especificación de endpoint (Ingreso obligatorio)
data.put(VisaNet.VISANET_END_POINT_URL, "https://apitestenv.vnforapps.com/");

//Personalización (Ingreso opcional)
VisaNetViewAuthorizationCustom custom = new VisaNetViewAuthorizationCustom();

//Personalización 1: Configuración del nombre del comercio en el formulario de pago
custom.setLogoTextMerchant(true);
custom.setLogoTextMerchantText(nomComercio.getText().toString());
custom.setLogoTextMerchantTextColor(R.color.visanet_black);
custom.setLogoTextMerchantTextSize(20);

//Personalización 2: Configuración del color del botón pagar en el formulario de pago
custom.setButtonColorMerchant(R.color.colorRed);
custom.setInputCustom(true);
```

iii. Paso 3: Concretar transacción

Dependiendo del método de pago (tarjeta o PagoEfectivo), habrán dos caminos para concretar la transacción.

a) Método de pago - Tarjeta

1) Invocar la librería

El paso final de la integración es invocar al método de **Autorización**, esta invocación se debe realizar de la siguiente manera:

```
try {
    VisaNet.authorization(Activity.this, data, custom);
}
catch (Exception e) {
    Log.i(TAG, "onClick: " + e.getMessage());
}
```

En donde “**data**” es todo lo configurado según los diferentes escenarios explicados y “**custom**” son todas las personalizaciones establecidas.

2) Respuesta de la librería

La respuesta al proceso de autorización se obtiene en el método “**onActivityResult**” de la siguiente manera:

```
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    super.onActivityResult(requestCode, resultCode, data);

    if (requestCode == VisaNet.VISANET_AUTHORIZATION) {
        if (data != null) {
            if (resultCode == RESULT_OK) {
                String jsonString = data.getExtras().getString("keySuccess");
                Toast toast1 = Toast.makeText(getApplicationContext(), jsonString, Toast.LENGTH_LONG);
                toast1.show();
            } else {
                String jsonString = data.getExtras().getString("keyError");
                jsonString = jsonString != null ? jsonString : "";
                Toast toast1 = Toast.makeText(getApplicationContext(), jsonString, Toast.LENGTH_LONG);
                toast1.show();
            }
        }
        else {
            Toast toast1 = Toast.makeText(getApplicationContext(), "Cancel...", Toast.LENGTH_LONG);
            toast1.show();
        }
    }
}
```

b) Método de pago – PagoEfectivo

1) Respuesta CIP

El formulario de pagos de PagoEfectivo, devolverá lo siguiente:

Parámetro	Descripción
cip	Es el código CIP generado por PagoEfectivo.
cipURL	URL generada por PagoEfectivo con los detalles e información para realizar el pago.
expiryDate	Fecha límite en la que expira el código CIP.

2) Actualización del estado de la venta

Para este caso el paso final de la integración es el actualizar el estado de la venta y confirmar si se realizó el pago por esta o expiró el tiempo de reserva. Para ello es necesario como

comercio se exponga un API callback, la cual de ser invocada de la siguiente manera:

Endpoint para conexión con la API Callback:

Ambiente	URL API Callback
Desarrollo	https://ambiente.comercio.com/api.pagoefectivocallback/v1/callback
Producción	https://ambiente.comercio.com/api.pagoefectivocallback/v1/callback

Request

POST https://ambiente.comercio.com/api.pagoefectivocallback

HEADER

Content-Type: application/json

BODY

```
{
  "operationNumber": "{ operationNumber }",
  "cip": "{ cip }",
  "status": "{ status }",
  "amount": { amount }
}
```

Los parámetros necesarios para la invocación del API callback se explican a continuación:

(*) Campos mínimos obligatorios para la solicitud de autorización

Parámetro	Descripción
operationNumber (*)	Número de compra o código de transacción
Cip (*)	Código CIP.
status (*)	Estado de la venta. Puede ser "Paid" Pagado o "Expired" Expirado
amount (*)	Monto registrado de la venta

Successful Response

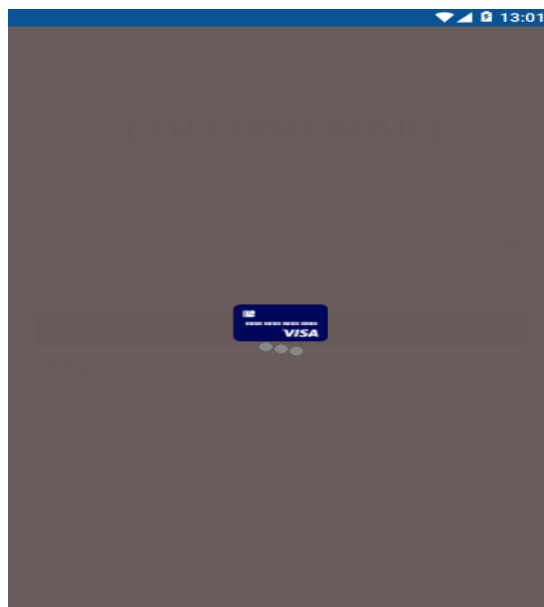
STATUS CODE 200 OK

HEADERS

Content-Type: application/json

iv. Comportamiento de la librería

- El formulario de pago presenta el siguiente ícono de carga (loading) al ser invocado:



- El formulario de pago ocupa toda la pantalla del dispositivo:

- Si faltase completar algún campo del formulario de pago, al momento de dar clic en “Pagar”, el formulario resaltará el campo en rojo:

v. Pruebas, Certificación y Producción

Para realizar las pruebas del formulario de pago, el comercio necesitará:

- Código de comercio: código que se le asignó al comercio para pruebas.
- Llaves de autenticación: usuario y contraseña.

Las pruebas y certificación consisten en lo siguiente:

- Pruebas de compra para validar los diferentes escenarios de la transacción, para lo cual el comercio deberá tener cargado algunos productos y sus montos respectivos.

El comercio tiene los siguientes escenarios de pruebas disponibles:

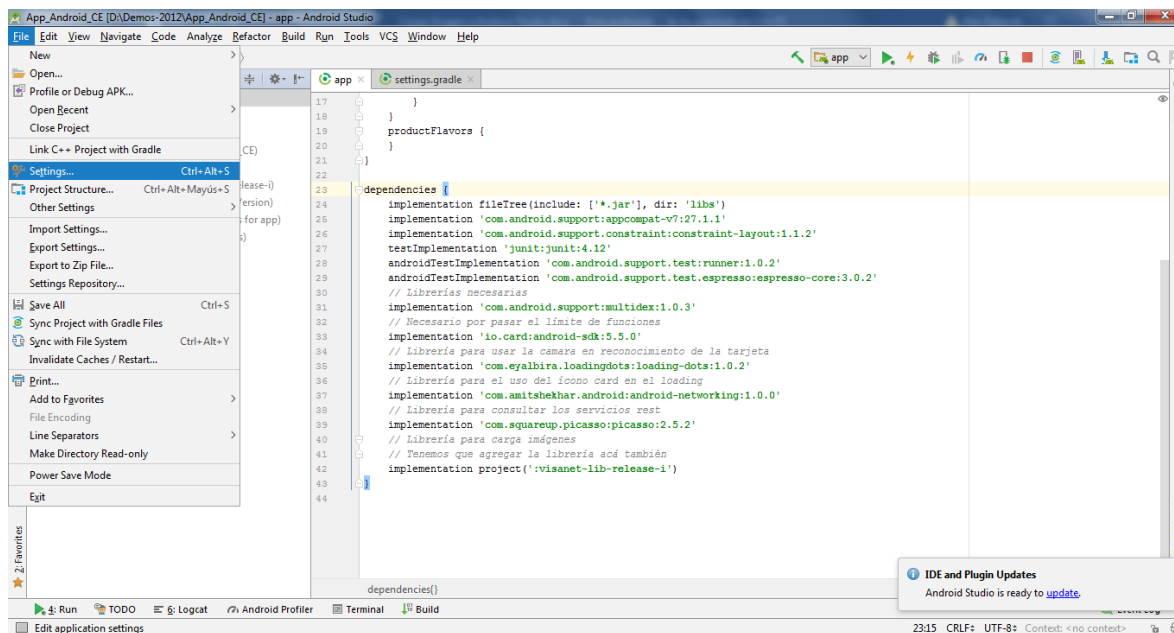
Escenario	Tarjeta de prueba
Autorizado	<u>Nacional</u>
	4919 – 1481 – 0785 – 9067
	4919 – 1481 – 0785 – 9067
	4500 – 3400 – 9000 – 0016
	<u>Foránea</u>
Denegado	4072 – 2102 – 9053 – 6663
	4444 – 3333 – 2222 – 1111
	4111 – 1111 – 1111 – 1111

Si las validaciones fueron exitosas, el equipo de integración coordina el pase a producción de la aplicación del comercio.

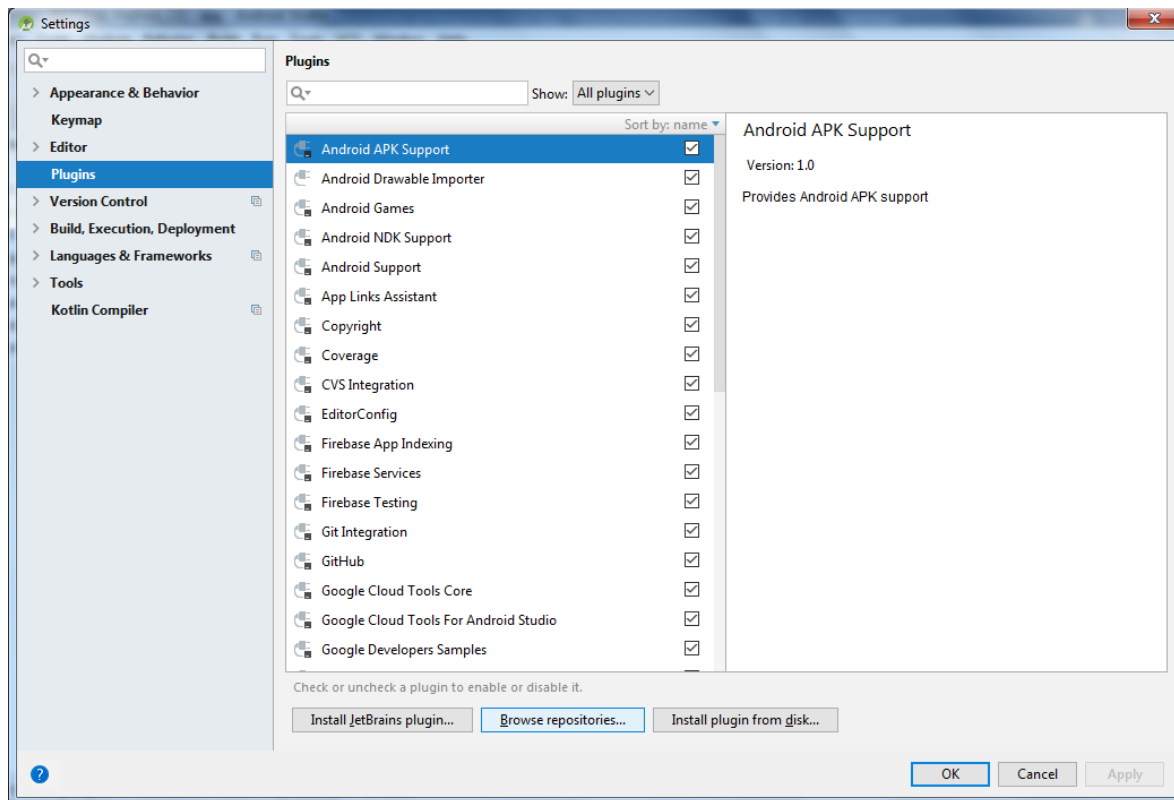
6. Anexos

a. Carga de logo del comercio en la librería

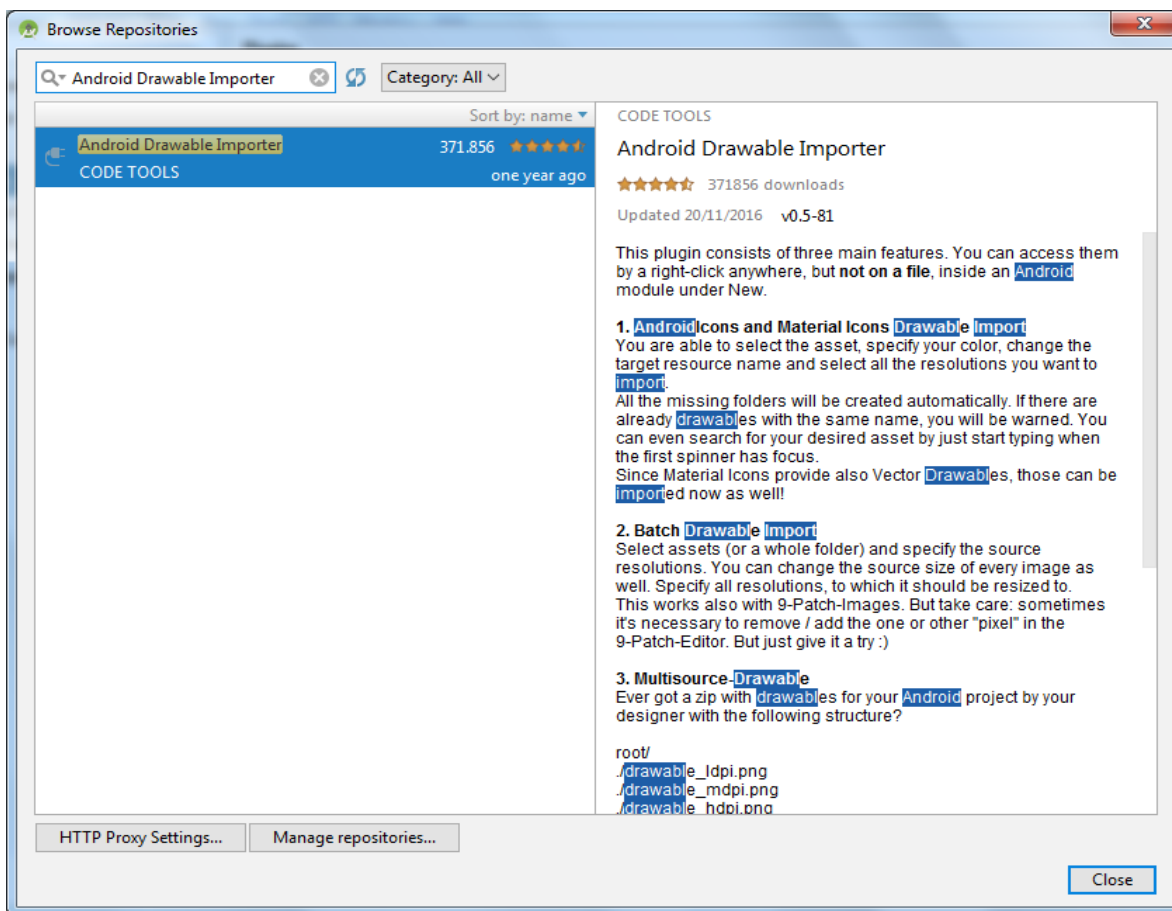
Estando en el proyecto Android, dar clic derecho y seleccionar la opción “File → Settings...”



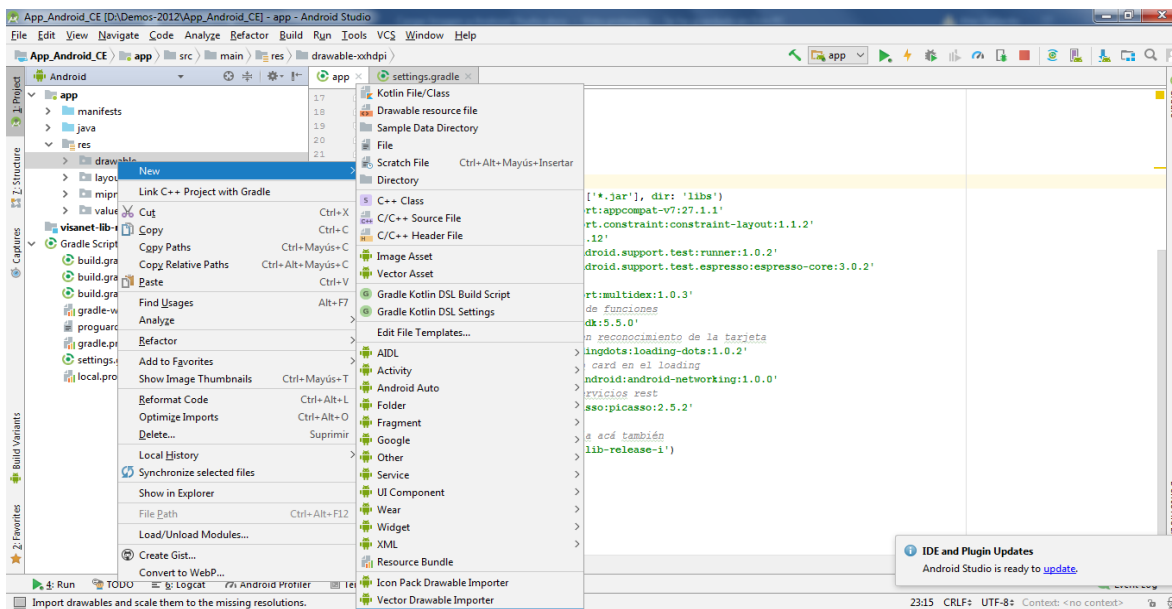
De la ventana que aparece seleccionar la opción “Plugins → Browse repositories...”



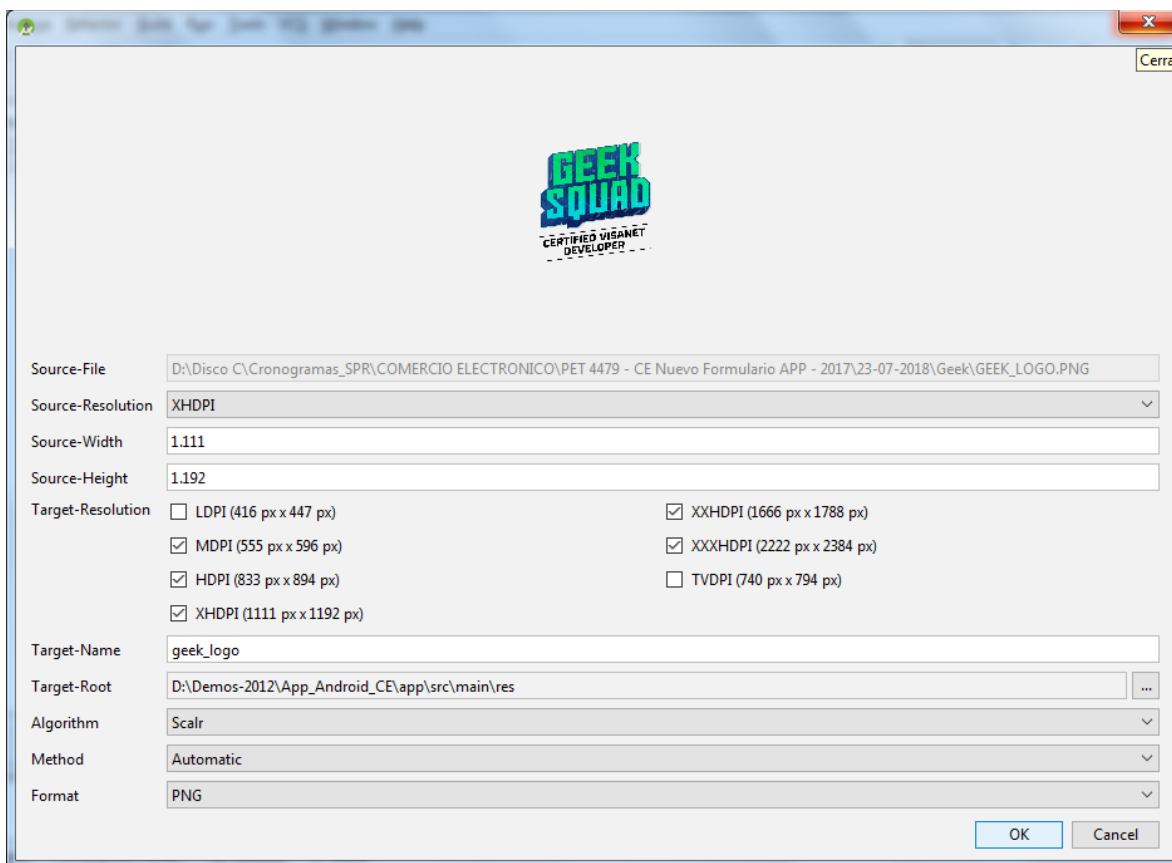
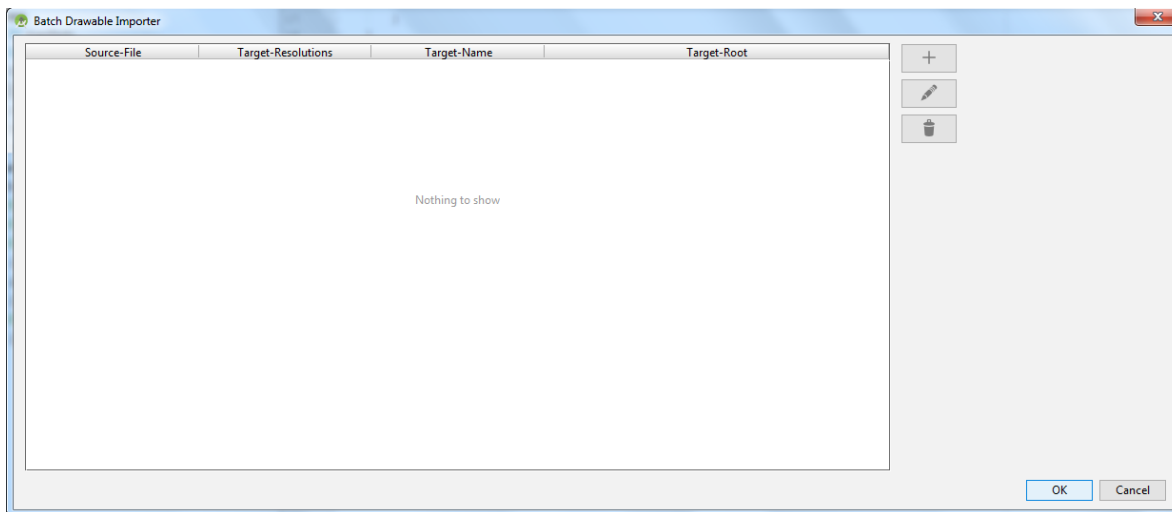
Se debe agregar el plugin “**Android Drawable Importer**” para importar imágenes al proyecto



Una vez instalado el plugin mencionado seleccionar la carpeta “**drawable**” del proyecto y dar clic derecho para finalmente escoger la opción “**New → Batch Drawable Import**”



Arrastrar el logo del comercio en la ventana que aparece, es acá en donde se especifica la calidad de imagen, nombre y tamaño



Para llamar al logo del comercio en el código fuente es de la siguiente manera:

R.drawable.unnamed (En donde **unnamed** = nombre de la imagen). Ejemplo:

```
VisaNetViewAuthorizationCustom custom = new VisaNetViewAuthorizationCustom();
custom.setLogoTextMerchant(false);
custom.setLogoImage(R.drawable.logoComercio);
```


b. Códigos de acción (denegaciones)

Códigos retornados desde el proceso de autorización para transacciones denegadas:

Código de acción	Descripción de apoyo para el comercio	Descripción a mostrar al cliente
101	Operación Denegada. Tarjeta Vencida.	Operación Denegada. Tarjeta Vencida. Verifique los datos en su tarjeta e ingréselos correctamente.
102	Operación Denegada. Contactar con la entidad emisora.	Operación Denegada. Contactar con entidad emisora de su tarjeta.
104	Operación Denegada. Operación no permitida para esta tarjeta.	Operación Denegada. Operación no permitida para esta tarjeta. Contactar con la entidad emisora de su tarjeta.
106	Operación Denegada. Exceso de intentos de ingreso de clave secreta.	Operación Denegada. Intentos de clave secreta excedidos. Contactar con la entidad emisora de su tarjeta.
107	Operación Denegada. Contactar con la entidad emisora.	Operación Denegada. Contactar con la entidad emisora de su tarjeta.
108	Operación Denegada. Exceso de actividad.	Operación Denegada. Contactar con la entidad emisora de su tarjeta.
109	Operación Denegada. Identificación inválida de establecimiento.	Operación Denegada. Contactar con el comercio.
110	Operación Denegada. Operación no permitida para esta tarjeta.	Operación Denegada. Operación no permitida para esta tarjeta. Contactar con la entidad emisora de su tarjeta.
111	Operación Denegada. El monto de la transacción supera el valor máximo permitido para operaciones virtuales	Operación Denegada. Contactar con el comercio.
112	Operación Denegada. Se requiere clave secreta.	Operación Denegada. Se requiere clave secreta.
116	Operación Denegada. Fondos insuficientes.	Operación Denegada. Fondos insuficientes. Contactar con entidad emisora de su tarjeta
117	Operación Denegada. Clave secreta incorrecta.	Operación Denegada. Clave secreta incorrecta.
118	Operación Denegada. Tarjeta inválida.	Operación Denegada. Tarjeta Inválida. Contactar con entidad emisora de su tarjeta.
119	Operación Denegada. Exceso de intentos de ingreso de clave secreta.	Operación Denegada. Intentos de clave secreta excedidos. Contactar con entidad emisora de su tarjeta.
121	Operación Denegada.	Operación Denegada.
126	Operación Denegada. Clave secreta inválida.	Operación Denegada. Clave secreta inválida.
129	Operación Denegada. Tarjeta no operativa.	Operación Denegada. Código de seguridad inválido. Contactar con entidad emisora de su tarjeta
180	Operación Denegada. Tarjeta inválida.	Operación Denegada. Tarjeta Inválida. Contactar con entidad emisora de su tarjeta.
181	Operación Denegada. Tarjeta con restricciones de Débito.	Operación Denegada. Tarjeta con restricciones de débito. Contactar con entidad emisora de su tarjeta.
182	Operación Denegada. Tarjeta con restricciones de Crédito.	Operación Denegada. Tarjeta con restricciones de crédito. Contactar con entidad emisora de su tarjeta.
183	Operación Denegada. Error de sistema.	Operación Denegada. Problemas de comunicación. Intente más tarde.
190	Operación Denegada. Contactar con entidad emisora.	Operación Denegada. Contactar con entidad emisora de su tarjeta.
191	Operación Denegada. Contactar con entidad emisora.	Operación Denegada. Contactar con entidad emisora de su tarjeta.
192	Operación Denegada. Contactar con entidad emisora.	Operación Denegada. Contactar con entidad emisora de su tarjeta.
199	Operación Denegada.	Operación Denegada.
201	Operación Denegada. Tarjeta vencida.	Operación Denegada. Tarjeta vencida. Contactar con entidad emisora de su tarjeta.
202	Operación Denegada. Contactar con entidad emisora.	Operación Denegada. Contactar con entidad emisora de su tarjeta
204	Operación Denegada. Operación no permitida para esta tarjeta.	Operación Denegada. Operación no permitida para esta tarjeta. Contactar con entidad emisora de su tarjeta.
206	Operación Denegada. Exceso de intentos de ingreso de clave secreta.	Operación Denegada. Intentos de clave secreta excedidos. Contactar con la entidad emisora de su tarjeta.
207	Operación Denegada. Contactar con entidad emisora.	Operación Denegada. Contactar con entidad emisora de su tarjeta.
208	Operación Denegada. Tarjeta perdida.	Operación Denegada. Contactar con entidad emisora de su tarjeta.
209	Operación Denegada. Tarjeta robada.	Operación Denegada. Contactar con entidad emisora de su tarjeta

263	Operación Denegada. Error en el envío de parámetros.	Operación Denegada. Contactar con el comercio.
264	Operación Denegada. Entidad emisora no está disponible para realizar la autenticación.	Operación Denegada. Entidad emisora de la tarjeta no está disponible para realizar la autenticación.
265	Operación Denegada. Clave secreta del tarjetahabiente incorrecta.	Operación Denegada. Clave secreta del tarjetahabiente incorrecta. Contactar con entidad emisora de su tarjeta.
266	Operación Denegada. Tarjeta vencida.	Operación Denegada. Tarjeta Vencida. Contactar con entidad emisora de su tarjeta.
280	Operación Denegada. Clave errónea.	Operación Denegada. Clave secreta errónea. Contactar con entidad emisora de su tarjeta.
290	Operación Denegada. Contactar con entidad emisora.	Operación Denegada. Contactar con entidad emisora de su tarjeta.
300	Operación Denegada. Número de pedido del comercio duplicado. Favor no atender.	Operación Denegada. Número de pedido del comercio duplicado. Favor no atender.
306	Operación Denegada. Contactar con entidad emisora.	Operación Denegada. Contactar con entidad emisora de su tarjeta.
401	Operación Denegada. Tienda inhabilitada.	Operación Denegada. Contactar con el comercio.
402	Operación Denegada.	Operación Denegada.
403	Operación Denegada. Tarjeta no autenticada	Operación Denegada. Tarjeta no autenticada.
404	Operación Denegada. El monto de la transacción supera el valor máximo permitido.	Operación Denegada. Contactar con el comercio.
405	Operación Denegada. La tarjeta ha superado la cantidad máxima de transacciones en el día.	Operación Denegada. Contactar con el comercio.
406	Operación Denegada. La tienda ha superado la cantidad máxima de transacciones en el día.	Operación Denegada. Contactar con el comercio.
407	Operación Denegada. El monto de la transacción no llega al mínimo permitido.	Operación Denegada. Contactar con el comercio.
408	Operación Denegada. CVV2 no coincide.	Operación Denegada. Código de seguridad no coincide. Contactar con entidad emisora de su tarjeta
409	Operación Denegada. CVV2 no procesado por entidad emisora.	Operación Denegada. Código de seguridad no procesado por la entidad emisora de la tarjeta
410	Operación Denegada. CVV2 no procesado por no ingresado.	Operación Denegada. Código de seguridad no ingresado.
411	Operación Denegada. CVV2 no procesado por entidad emisora.	Operación Denegada. Código de seguridad no procesado por la entidad emisora de la tarjeta
412	Operación Denegada. CVV2 no reconocido por entidad emisora.	Operación Denegada. Código de seguridad no reconocido por la entidad emisora de la tarjeta
413	Operación Denegada. Contactar con entidad emisora.	Operación Denegada. Contactar con entidad emisora de su tarjeta.
414	Operación Denegada.	Operación Denegada.
415	Operación Denegada.	Operación Denegada.
416	Operación Denegada.	Operación Denegada.
417	Operación Denegada.	Operación Denegada.
418	Operación Denegada.	Operación Denegada.
419	Operación Denegada.	Operación Denegada.
420	Operación Denegada. Tarjeta no es VISA.	Operación Denegada. Tarjeta no es VISA.
421	Operación Denegada. Contactar con entidad emisora.	Operación Denegada. Contactar con entidad emisora de su tarjeta.
422	Operación Denegada. El comercio no está configurado para usar este medio de pago.	Operación Denegada. El comercio no está configurado para usar este medio de pago. Contactar con el comercio.
423	Operación Denegada. Se canceló el proceso de pago.	Operación Denegada. Se canceló el proceso de pago.
424	Operación Denegada. Contactar con entidad emisora.	Operación Denegada.
666	Operación Denegada. Problemas de comunicación. Intente más tarde.	Operación Denegada. Problemas de comunicación. Intente más tarde.
667	Operación Denegada. Transacción sin autenticación. Inicio del Proceso de Pago	Operación Denegada. Transacción sin respuesta de Verified by Visa.
668	Operación Denegada.	Operación Denegada. Contactar con el comercio.
669	Operación Denegada.	Operación Denegada. Contactar con el comercio.
670	Operación Denegada. Módulo antifraude.	Operación Denegada. Contactar con el comercio.
672	Operación Denegada. Transacción sin respuesta de Antifraude.	Operación Denegada. Módulo antifraude.

673	Operación Denegada. Transacción sin respuesta del Autorizador.	Operación Denegada. Contactar con el comercio.
674	Operación Denegada. Sesión no válida.	Operación Denegada. Contactar con el comercio.
675	Inicialización de transacción	Inicialización de transacción
676	Operación Denegada. No activa la opción Revisar Enviar al Autorizador.	Operación Denegada. Contactar con el comercio.
677	Operación Denegada. Respuesta Antifraude con parámetros nos válidos.	Operación Denegada. Contactar con el comercio.
678	Operación Denegada. Valor ECI no válido.	Operación Denegada. Contactar con el comercio.
682	Operación Denegada. Intento de Pago fuera del tiempo permitido.	Operación Denegada. Operación Denegada.
683	Operación Denegada. Registro incorrecto de sesión.	Operación Denegada. Registro incorrecto de sesión.
684	Operación Denegada Registro Incorrecto Antifraude	Operación Denegada Registro Incorrecto Antifraude
685	Operación Denegada Registro Incorrecto Autorizador	Operación Denegada Registro Incorrecto Autorizador
904	Operación Denegada. Formato de mensaje erróneo.	Operación Denegada.
909	Operación Denegada. Error de sistema.	Operación Denegada. Problemas de comunicación. Intente más tarde.
910	Operación Denegada. Error de sistema.	Operación Denegada.
912	Operación Denegada. Entidad emisora no disponible.	Operación Denegada. Entidad emisora de la tarjeta no disponible
913	Operación Denegada. Transmisión duplicada.	Operación Denegada.
916	Operación Denegada. Contactar con entidad emisora.	Operación Denegada.
928	Operación Denegada. Contactar con entidad emisora.	Operación Denegada.
940	Operación Denegada. Transacción anulada previamente.	Operación Denegada.
941	Operación Denegada. Transacción ya anulada previamente.	Operación Denegada.
942	Operación Denegada.	Operación Denegada.
943	Operación Denegada. Datos originales distintos.	Operación Denegada.
945	Operación Denegada. Referencia repetida.	Operación Denegada.
946	Operación Denegada. Operación de anulación en proceso.	Operación Denegada. Operación de anulación en proceso.
947	Operación Denegada. Comunicación duplicada.	Operación Denegada. Problemas de comunicación. Intente más tarde.
948	Operación Denegada. Contactar con entidad emisora.	Operación Denegada.
949	Operación Denegada. Contactar con entidad emisora.	Operación Denegada.
965	Operación Denegada. Contactar con entidad emisora.	Operación Denegada.

c. Diccionario de atributos enviados a la librería

Parámetro	Descripción
VisaNet.VISANET_CHANNEL	Canal de registro. Puede ser de tres tipos: VisaNet.Channel.MOBILE VisaNet.Channel.CALLCENTER VisaNet.Channel.RECURRENT
VisaNet.VISANET_COUNTABLE	Este campo indica si la venta a realizar tendrá liquidación automática o manual. Acepta los siguientes valores: true – Para liquidación automática false – Para liquidación manual Revisar anexo “e”
VisaNet.VISANET_USERNAME	Usuario para acceso a la librería
VisaNet.VISANET_PASSWORD	Contraseña del usuario para acceso a la librería
VisaNet.VISANET_MERCHANT	Código de comercio
VisaNet.VISANET_PURCHASE_NUMBER	Número de pedido
VisaNet.VISANET_AMOUNT	Monto de la transacción
VisaNet.VISANET_REGISTER_NAME	Nombre del tarjeta habiente
VisaNet.VISANET_REGISTER_LASTNAME	Apellido del tarjeta habiente
VisaNet.VISANET_REGISTER_EMAIL	Correo electrónico del tarjeta habiente
VisaNet.VISANET_USER_TOKEN	Token del usuario
VisaNet.VISANET_SHOW_AMOUNT	Indicador para ver monto en el botón “pagar”
VisaNet.VISANET_MDD	Colección de MDDs adicionales
VisaNet.VISANET_END_POINT_URL	Endpoint del entorno (Integración/Producción)
VisaNet.VISANET_RECURRENCE	Indicador de recurrencia. Puede ser de dos tipos: true false
VisaNet.VISANET_RECURRENCE_TYPE	Tipo de recurrencia. Puede ser de cuatros tipos: VisaNet.RecurrenceType.FIXED VisaNet.RecurrenceType.VARIABLE VisaNet.RecurrenceType.FIXED_INITIAL VisaNet.RecurrenceType.VARIABLE_INITIAL
VisaNet.VISANET_RECURRENCE_FREQUENCY	Tipo de frecuencia. Puede ser de cuatro tipos: VisaNet.RecurrenceFrequency.MONTHLY VisaNet.RecurrenceFrequency.QUARTERLY VisaNet.RecurrenceFrequency.BIANNUAL VisaNet.RecurrenceFrequency.ANNUAL
VisaNet.VISANET_RECURRENCE_MAX_AMOUNT	Monto máximo para los cargos
VisaNet.VISANET_RECURRENCE_AMOUNT	Monto del cargo
VisaNet.VISANET_RECURRENCE_DOCUMENT_TYPE	Tipo de documento. Puede ser de tres tipos: VisaNet.DocumentType.DNI VisaNet.DocumentType.CARNET_EXTRANJERIA VisaNet.DocumentType.PASAPORTE
VisaNet.VISANET_RECURRENCE_DOCUMENT_ID	Número de document
VisaNet.VISANET_RECURRENCE_PRODUCT_ID	Código de producto
VisaNet.VISANET_RECURRENCE_BENEFICIARY_ID	Id del beneficiario
VisaNet.VISANET_RECURRENCE_BENEFICIARY_NAME	Nombre del beneficiario
VisaNet.VISANET_RECURRENCE_BENEFICIARY_LASTNAME	Apellido del beneficiario

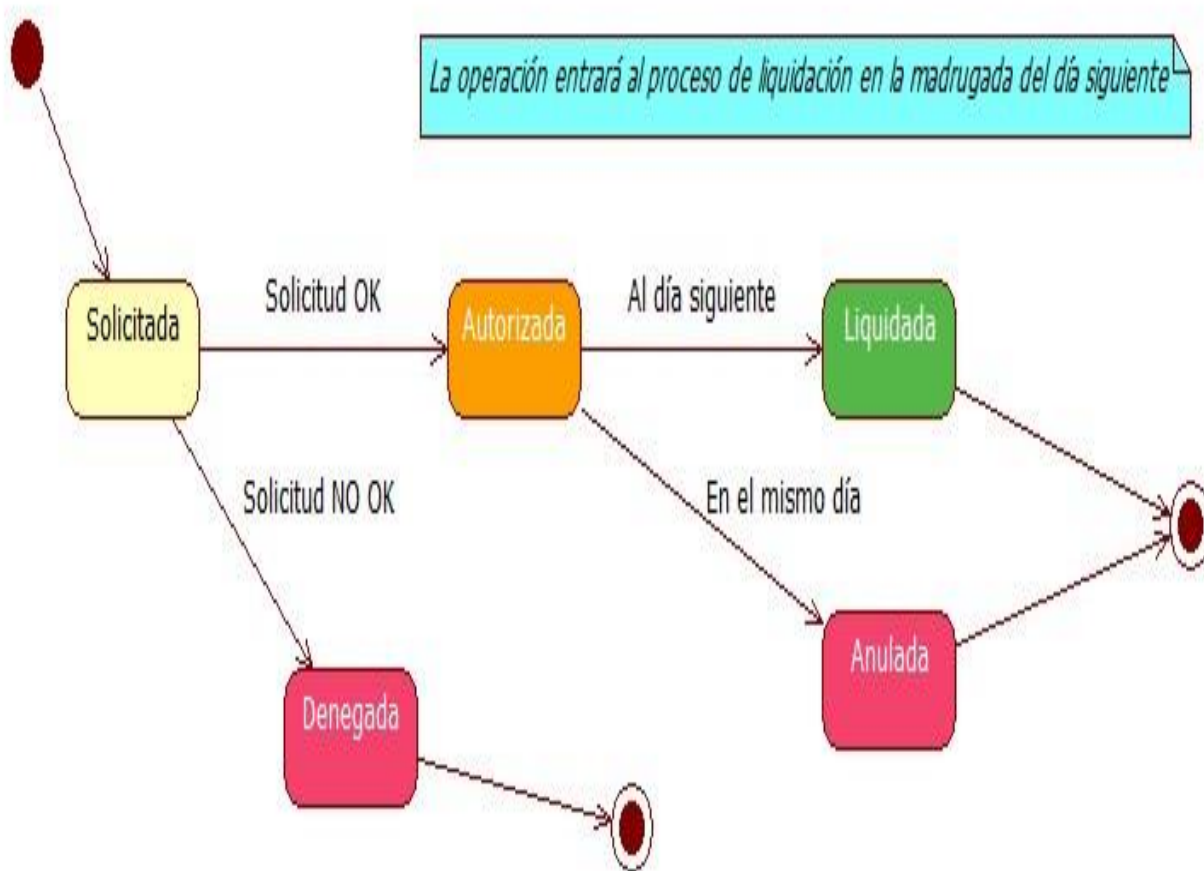
d. Diccionario de atributos devueltos por la librería

Parámetro	Descripción
header	Objeto cabecera
ecoreTransactionUUID	Código de la transacción dentro de las API de VisaNet
ecoreTransactionDate	Fecha y hora de la transacción. Formato: millis
millis	Tiempo de duración de la transacción
order	Objeto orden
tokenId	Token retornado del formulario de pago
purchaseNumber	Número de pedido generado por el comercio
productId	Código de producto registrado en el sistema pagos recurrentes VisaNet
amount	Importe de la transacción Formato ####.## (dos decimales separados por punto) Ejemplo: 1000.00
currency	Moneda de la transacción
authorizedAmount	Importe autorizado
authorizationCode	Código de autorización
actionCode	Código de denegación y aprobación. El Código de aprobación: 000
traceNumber	Número de orden de la transacción generada en los sistemas de VisaNet
transactionDate	Fecha y hora de la transacción. Formato UNIX TimeStamp
transactionId	ID único de la transacción del sistema VisaNet
token	Objeto tarjeta
tokenId	Número de tarjeta tokenizada
expireOn	Fecha de caducidad del token, Formato: yyMMddHHmmss
dataMap	Objeto datos complementarios
CURRENCY	Moneda de la transacción
TRANSACTION_DATE	Fecha y hora de la transacción. Formato UNIX TimeStamp
TERMINAL	Id de Terminal
ACTION_CODE	Código de denegación y aprobación. El Código de aprobación: 000
TRACE_NUMBER	Número de orden de la transacción generada en los sistemas de VisaNet
ECI_DESCRIPTION	Descripción del ECI (Electronic Commerce Indicator)
ECI	Código de ECI (Electronic Commerce Indicator)
BRAND	Marca de la tarjeta usada en el pago. Puede ser: visa, amex, mastercard, dinersclub
CARD	Número de tarjeta Visa. Está enmascarado. Se puede visualizar el BIN y los 4 últimos dígitos como si siguiente ejemplo: 491914*****9067
MERCHANT	Código de Comercio, creado al momento de la afiliación al producto Comercio Electrónico Pago Web
STATUS	Estado de la transacción
ADQUIRENTE	Código de adquirente
ACTION_DESCRIPTION	Descripción del código de acción, permite identificar el motivo de rechazo de una operación
QUOTA_AMOUNT	Importe aproximado del valor de cuota, este valor es devuelto por el emisor
ID_UNICO	Identificador único de la transacción generado por el sistema transaccional de VisaNet. Este permite rastrear la operación en los diferentes sistemas de VisaNet. Además, puede ser utilizado para procesos de conciliación automáticas
AMOUNT	Importe autorizado
PROCESS_CODE	Código de proceso
QUOTA_NUMBER	Número de cuotas
VAULT_BLOCK	Token generado por el tarjetahabiente al marcar la opción de recordar tarjeta
RECURRENCE_STATUS	Estado de registro en recurrentes
RECURRENCE_MESSAGE	Mensaje de respuesta de recurrentes
TRANSACTION_ID	ID único de la transacción del sistema VisaNet
AUTHORIZATION_CODE	Código de autorización
QUOTA_DEFERRED	Indica si el pago en cuotas debe procesarse con pagos en diferido

	0 = el pago no se procesa en diferido 1, 2 = el pago se procesa en diferido
SIGNATURE	UUID generado para el proceso de venta, el cual es utilizado en la reversa de la operación. Ejemplo: f9303580-8402-438f-bf52-f48c3cdc90ff
EXPIRATION_MONTH	Mes de vencimiento de la tarjeta. Ejemplo: 09
EXPIRATION_YEAR	Año de vencimiento de la tarjeta. Ejemplo: 2026

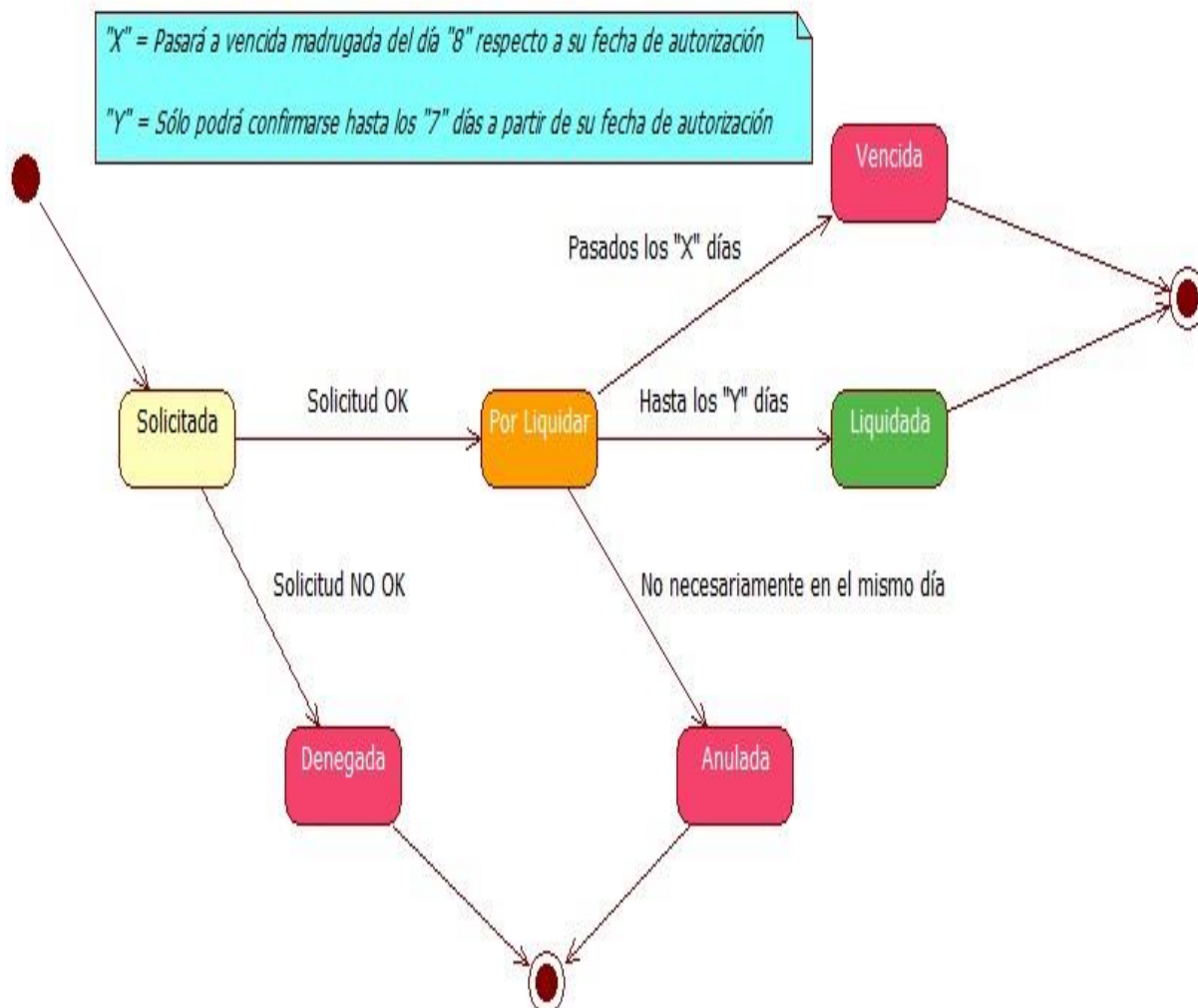
e. Transición de estados para una venta

Una venta si es <<**Contable**>> tiene la siguiente transición de estados:



Una venta <<**Contable**>> al ser aprobada pasa al estado "**Autorizada**", caso contrario pasa al estado "**Denegada**". La venta en estado "**Autorizada**" sólo puede ser "**Anulada**" el mismo día que su aprobación. En la madrugada del día siguiente la venta en estado "**Autorizada**" pasará en automático al estado "**Liquidada**".

Una venta si es <<**No Contable**>> tiene la siguiente transición de estados:



Una venta <<**No Contable**>> al ser aprobada pasa al estado "**Por Liquidar**", caso contrario pasa al estado "**Denegada**". La venta en estado "**Por Liquidar**" puede ser "**Anulada**" hasta que no sea confirmada. Si la venta es confirmada pasa al estado "**Liquidada**". Si la venta no es confirmada en los 7 días de haber sido aprobada, en la madrugada siguiente (Día 8) pasará al estado "**Vencida**".

f. API de autorización de venta

Tipo	Especificación
PDF	API de Autorización de Venta.pdf
SWAGGER	https://app.swaggerhub.com/apis-docs/VisaNetPeru/api.visanet.authorization/1.0.0

g. API de anulación de venta

Tipo	Especificación
PDF	API de Anulación de Venta.pdf
SWAGGER	https://app.swaggerhub.com/apis-docs/VisaNetPeru/api.visanet.void/1.0.0

h. API de confirmación de venta

Tipo	Especificación
PDF	API de Confirmación de Venta.pdf
SWAGGER	https://app.swaggerhub.com/apis-docs/VisaNetPeru/api.visanet.confirmation/1.0.0

i. API de consulta de venta

Tipo	Especificación
PDF	API de Consulta de Venta.pdf
SWAGGER	https://app.swaggerhub.com/apis-docs/VisaNetPeru/api.visanet.retrieve/1.0.0

j. API de antifraude para la venta

Tipo	Especificación
PDF	API de Antifraude para la Venta.pdf
SWAGGER	https://app.swaggerhub.com/apis-docs/VisaNetPeru/api.visanet.antifraud/1.0.0