# Mobile SDK Android Guide

# Mobile SDK Intro

Unlimint mobile SDK for Android (UnlimintSdk) helps you to:

Embed card data forms in the merchant's mobile app and securely collect and transmit the user's card data for:

- Card tokenization (without a payment) on the Unlimint side
- Making a mobile payment
- Maling a payment with card token

Android minSdkVersion 19

**Wise notice**
Unlimint mobile SDK just reminds you that the device is rooted. Please take a look at our security approach. You can accept working on Rooted devices or reject a payment.

For integration with MobileSdk you have to do the following steps:

1. Add Mobile Sdk .aar to your project and next dependencies to your Android project

   ```
   repositories {
       google()
       jcenter()

       maven { url "https://repos.unlimint.io/repository/mobile-sdk/" }
   }
   ```

   ```
   implementation 'com.unlimint.sdk:mobile-sdk:$version'
   ```

2. Call this method "MobileSdk.bindNewCardForResult(...)" for card binding or "MobileSdk.paymentForResult(...)" for making a mobile payment
3. For receiving the result of card binding or mobile payment you have to implement onActivityResult(...) method
4. After sending a card data (using the "MobileSdk.bindNewCardForResult(...)" or "MobileSdk.paymentForResult(...)" methods) and before 3DS verification procedure SDK receives "transaction_id" parameter,
5. Mobile application should send this parameter to your Server part.
6. Please, create an instance of Service and wait for the transaction_id

   ```
   LocalBroadcastManager.getInstance(this).registerReceiver(listener, IntentFilter(MobileSdk.TransactionData.TRANSACTION_ACTION))
   val transactionId = intent?.getStringExtra(MobileSdk.TransactionData.TRANSACTION_ID)
   ```

7. Add style to your styles.xml

   ```
   <style name="MobileSdkStyle" parent="Theme.MobileSdk.Light">
   ```

8. Add <activity> to your manifest.xml with your added theme

   For card binding

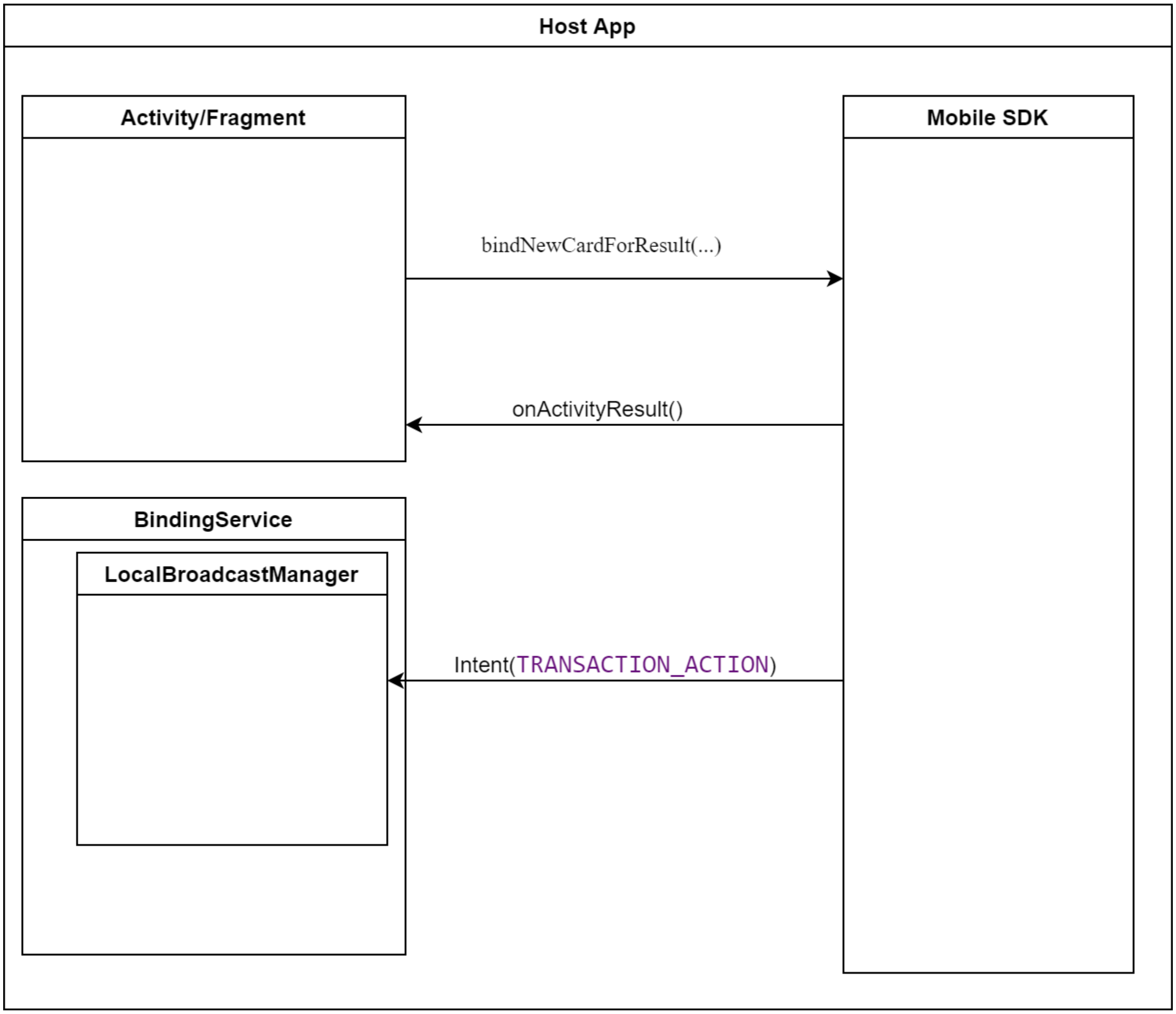
   ```
   <activity
   android:name=".sdk.scenario.binding.view.BindActivity"
   android:launchMode="singleInstance"
   android:screenOrientation="portrait"
   android:theme="@style/MobileSdkStyle"
   android:windowSoftInputMode="adjustResize" />
   ```

   For mobile payment

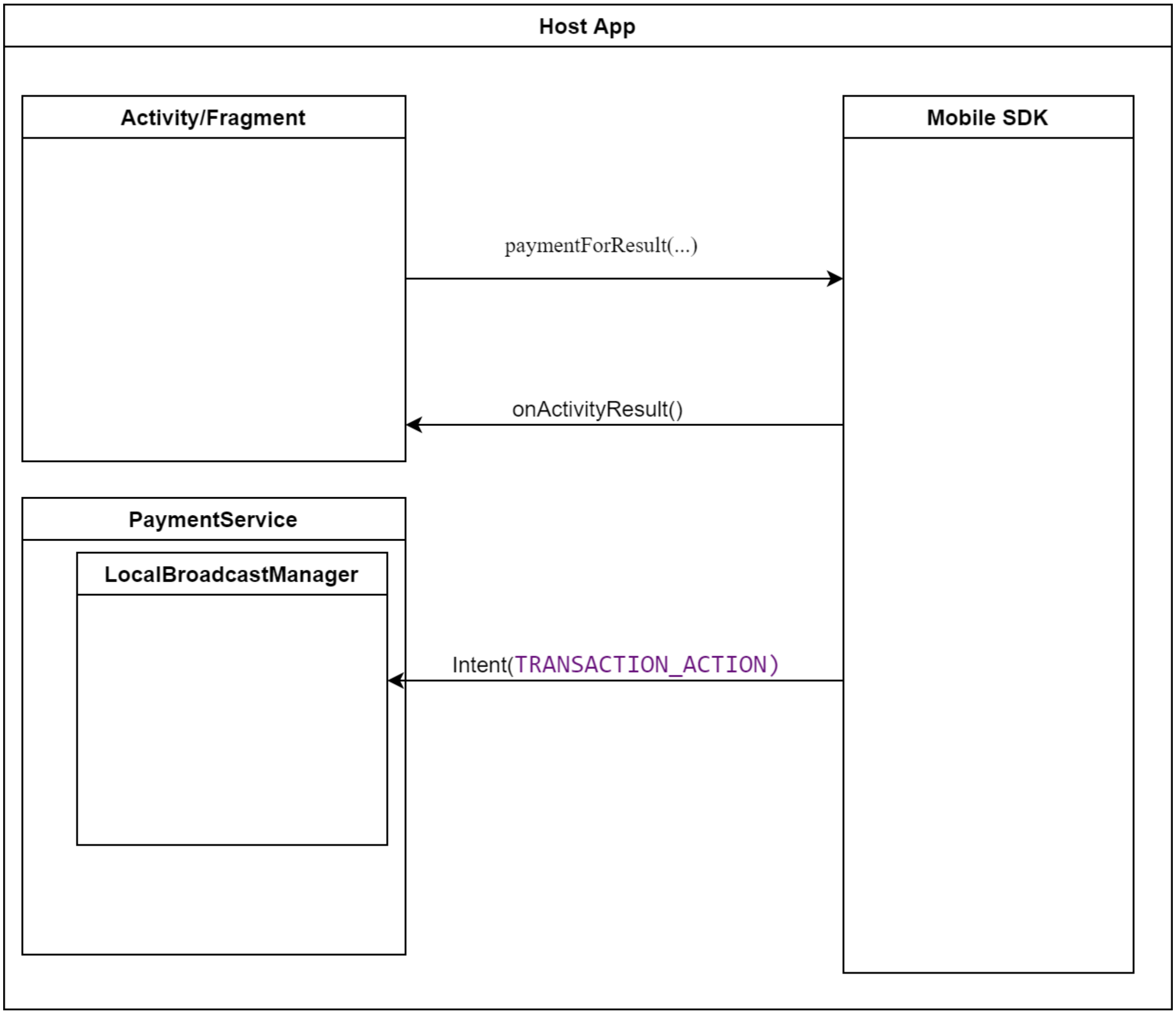
   ```
   <activity
   android:name=".sdk.scenario.payment.view.PaymentActivity"
   android:launchMode="singleInstance"
   android:screenOrientation="portrait"
   android:theme="@style/MobileSdkStyle"
   android:windowSoftInputMode="adjustResize" />
   ```

# Mobile SDK API description

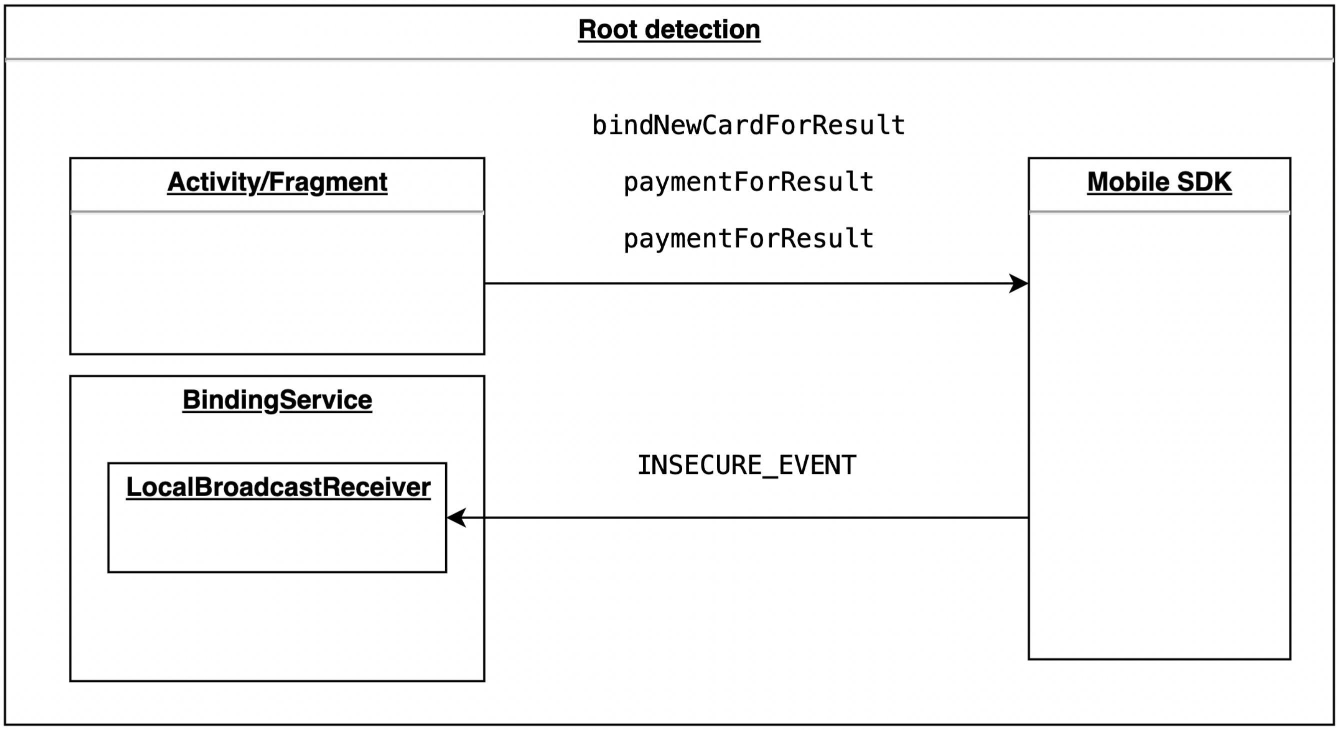
Card binding API description:



Mobile payment API description:



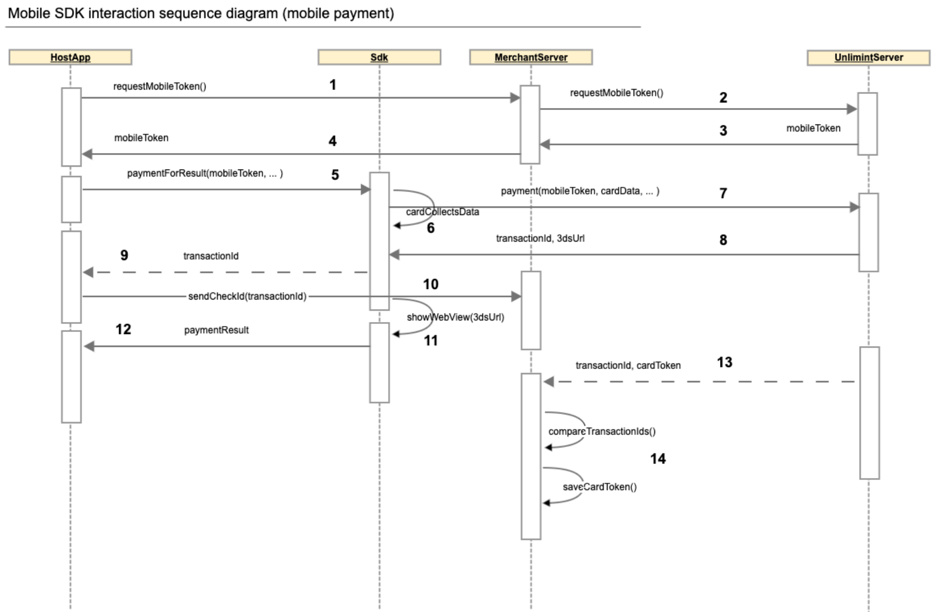
Root detection:



Close SDK:

SDK can be closed via **close()** method (for example, if **INSECURE_EVENT** is comming with **true**)

# Mobile API SDK interaction

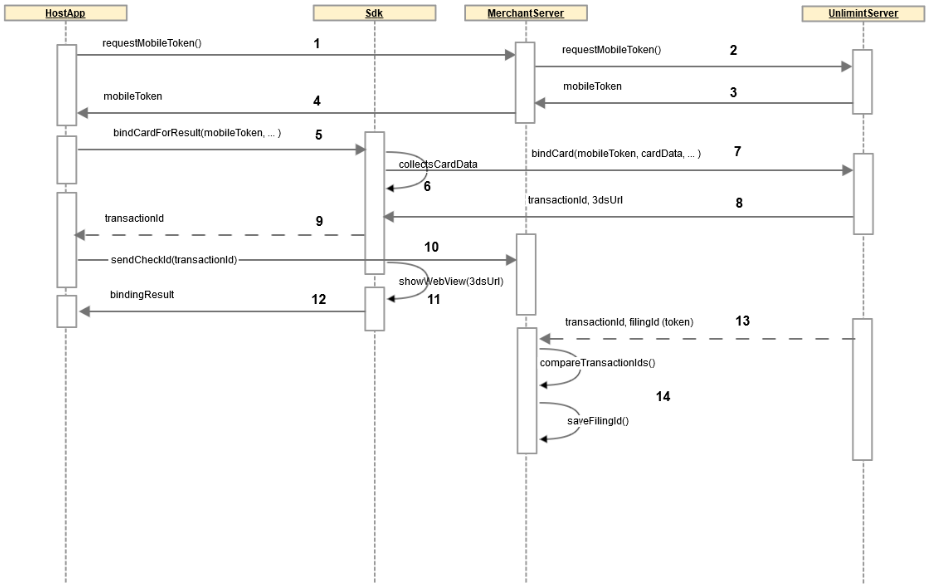


User scenario for mobile payment process:

| Step | Requirement text |
|---|---|
| 1 | Host (merchant) mobile application sends request for getting mobile token to merchant backend (requestMobileToken) |
| 2 | Merchant backend sends POST request (JSON) with valid access token for getting mobile token to API v3 endpoint https://cardpay.com/api/mobile/token<br>Header of request: valid access_token of merchant<br>Parameters of request:<br>request.id - request ID, should be unique for time period of 30 minutesrequest.time - request attempt date and time up to milliseconds in ISO 8601 format (milliseconds is optional part)(example of format - yyyy-MM-dd'T'HH:mm:ss.SSS'Z' |
| 3 | API v3 returns response to merchant backend with requested mobile token:<br>**-** mobile_token (string, unique identifier, 128 symbols)<br>**-** expires ( date and time of mobile token expiration in ISO 8601 format, example of format - yyyy-MM-dd'T'HH:mm:ss.SSS'Z')<br>Lifetime of mobile token is (see below):<br>'lifetime of mobile token' <= 5 min and < 'Access_token' ('Bearer' token) life time.<br>Granted LT of access token (for mobile token creation) should be less or equal 5 min but more or equeal than 4 min<br>4 min <= access token<= 5 min |
| 4 | Merchant backend sends mobile token to the host application |
| 5 | Host application calls paymentForResult(mobileToken) method of the mobile SDK |
| 6 | Customer fills in card data in card data form (in SDK) |
| 7 | Mobile SDK sends request with customer, card data, received mobile token (JSON) for making a payment to API v3 endpoint https://cardpay.com/api/mobile/* (here is presented masked endpoint) |
| 8 | API v3 sends a payment response with redirect URL and transaction id (for 3DS verification) to the mobile SDK |
| 9 | Mobile SDK returns transaction id to the host application, host application resend it to the merchant backend |
| 10 | Host application sends transaction id to the merchant server |

| | |
|---|---|
| 11 | Mobile SDK presents webview with 3dsUrl to the customer for 3-D Secure verification |
| 12 | 3-D Secure verification procedure passes and customer redirects to success or decline url (paymentResult) |
| 13 | Unlimint API v3 sends callback to the merchant backend (with transaction id and card token (if it was requested by customer) |
| 14 | Merchant backend compares received transaction id's from the host application and callback and saves a received card token for a future use (recommendations to do) |

Mobile SDK interaction sequence diagram (binding card)



User scenario for card binding process:

| Step | Requirement text |
|---|---|
| 1 | Host (merchant) mobile application sends request for getting mobile token to merchant backend (requestMobileToken) |
| 2 | Merchant backend sends POST request (JSON) with valid access token for getting mobile token to API v3 endpoint  https://cardpay.com/api/mobile/token<br>Header of request: valid access_token of merchant.<br>Parameters of request:<br>request.id - request ID, should be unique for time period of 30 minutesrequest.time - request attempt date and time up to milliseconds in ISO 8601 format (milliseconds is optional part)(example of format - yyyy-MM-dd'T'HH:mm:ss.SSS'Z' |
| 3 | API v3 returns response to merchant backend with requested mobile token:<br>**-** mobile_token (string, unique identifier, 128 symbols)<br>**-** expires ( date and time of mobile token expiration in ISO 8601 format, example of format - yyyy-MM-dd'T'HH:mm:ss.SSS'Z')\ |
| 4 | Merchant backend sends mobile token to the host application |
| 5 | Host application calls bindCardForResult(mobileToken) method of the mobile SDK |
| 6 | Customer fills in card data in card data form (in SDK) |
| 7 | Mobile SDK sends request with customer, card data, received mobile token (JSON) for card binding to API v3 endpoint  https://cardpay.com/api/mobile/* (here is presented masked endpoint) |
| 8 | API v3 sends a card binding response with redirect URL and transaction id (for 3DS verification) to the mobile SDK |
| 9 | Mobile SDK returns transaction id to the host application, host application resend it to the merchant backend |
| 10 | Host application sends transaction id to the merchant server |
| 11 | Mobile SDK presents webview with 3dsUrl to the customer for 3-D Secure verification |

| | |
|---|---|
| 12 | 3-D Secure verification procedure passes and customer redirects to success or decline url (bindingResult) |
| 13 | Unlimint API v3 sends callback to the merchant backend (with transaction id and filing id) |
| 14 | Merchant backend compares received transaction id's from the host application and callback and saves a received filing id for a future use (recommendations to do) |

# Mobile SDK customisation

This page shows you the different ways that you can customize the UI in SDK

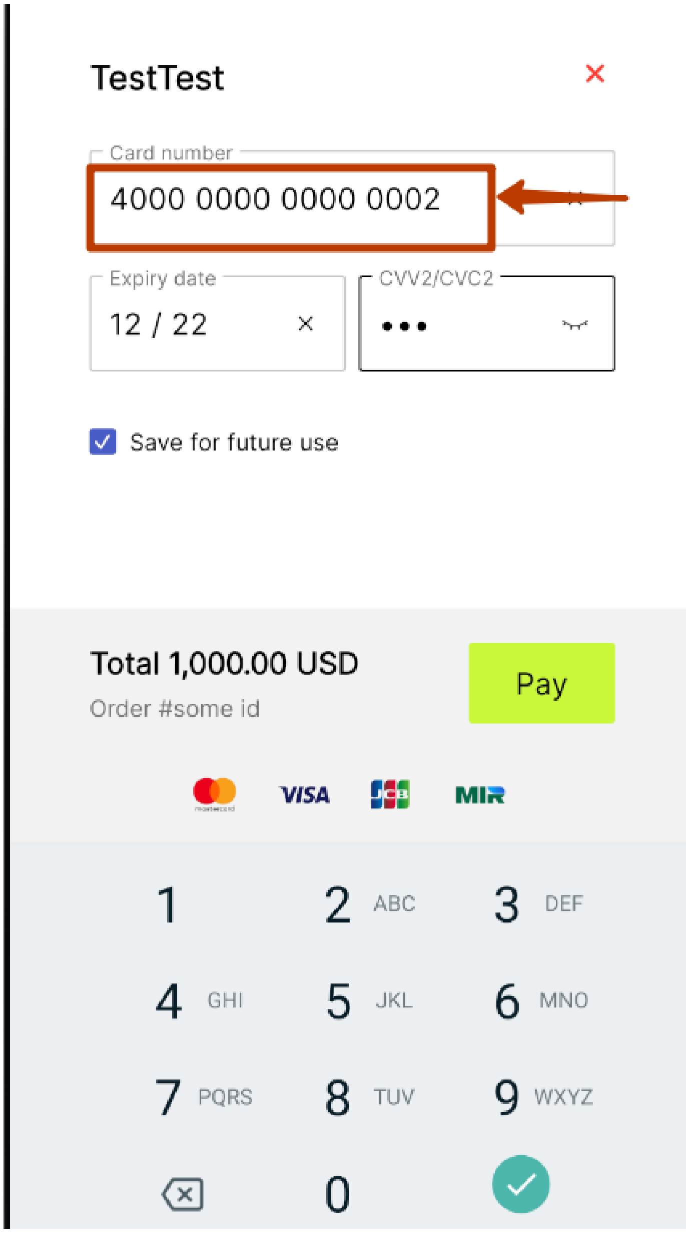
You can customize several UI styles and elements look:

- texts (hints and information), titles styles
- buttons style
- input layouts styles
- payment messages styles
- status bar, bottom area, background, text colors
- card binding elements customization
- title anf regular fonts

1. Texts (hints and information), titles styles

You can customize hint style as mentioned in image

<attr name="cp_sdk_hint_style" format="reference" />



TestTest

Card number: 4000 0000 0000 0002

Expiry date: 12 / 22

CVV2/CVC2: •••

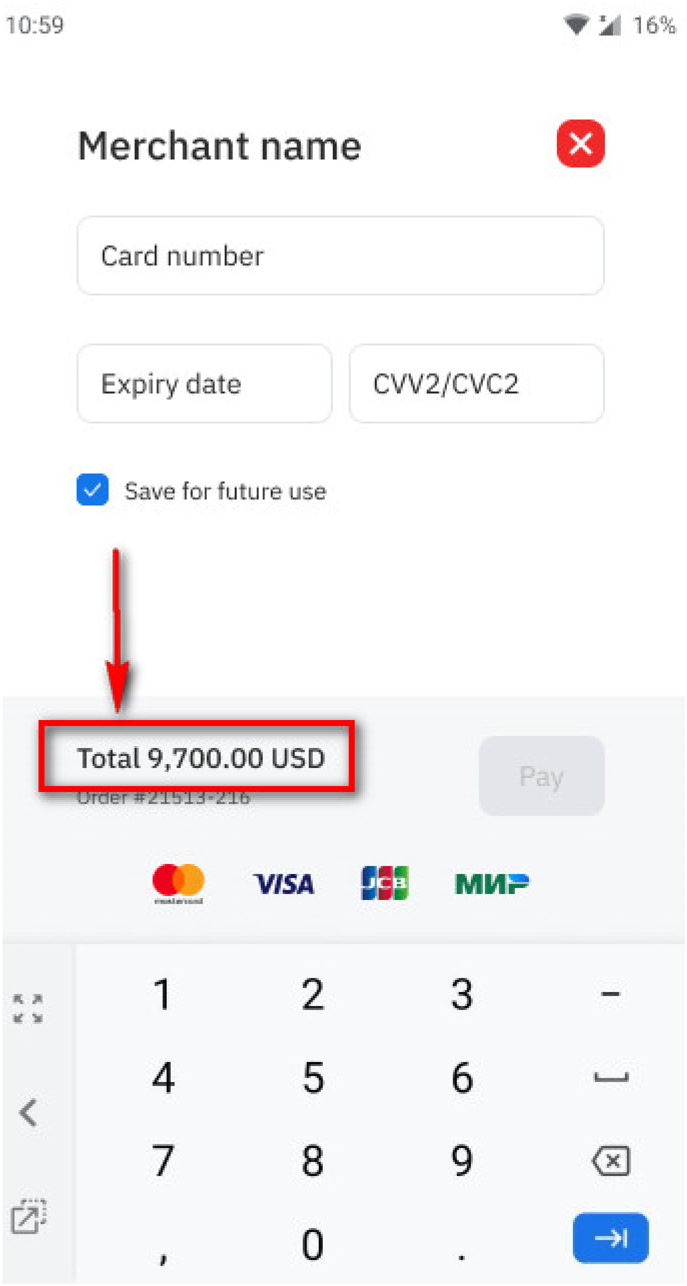
☑ Save for future use

Total 1,000.00 USD

Order #some id

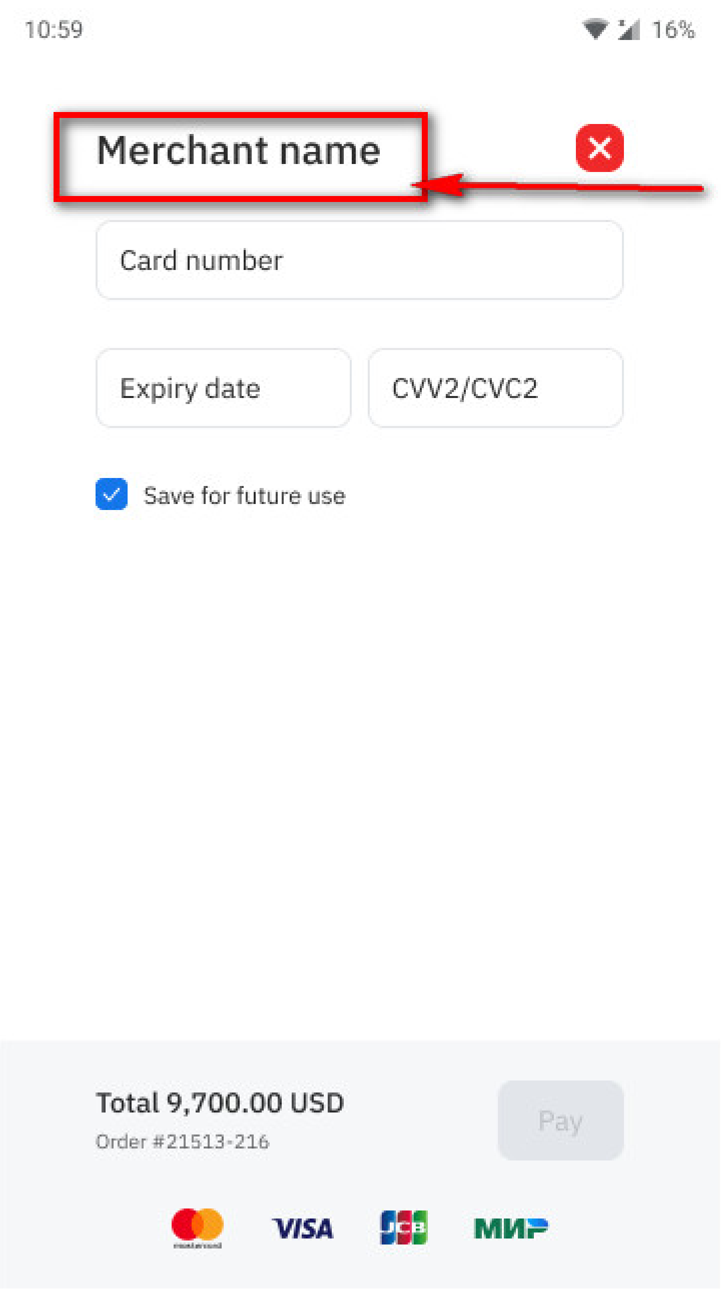
Also you can customize information and title styles for each info text or each title

<attr name="cp_sdk_info_style" format="reference" />



10:59
16%
Merchant name
Card number
Expiry date
CVV2/CVC2
☑ Save for future use
Total 9,700.00 USD
Order #21513-216
Pay
VISA
JCB
МИР
1
2
3
–
4
5
6
7
8
9
,
0
.

```
<attr name="cp_sdk_title_style" format="reference" />
```



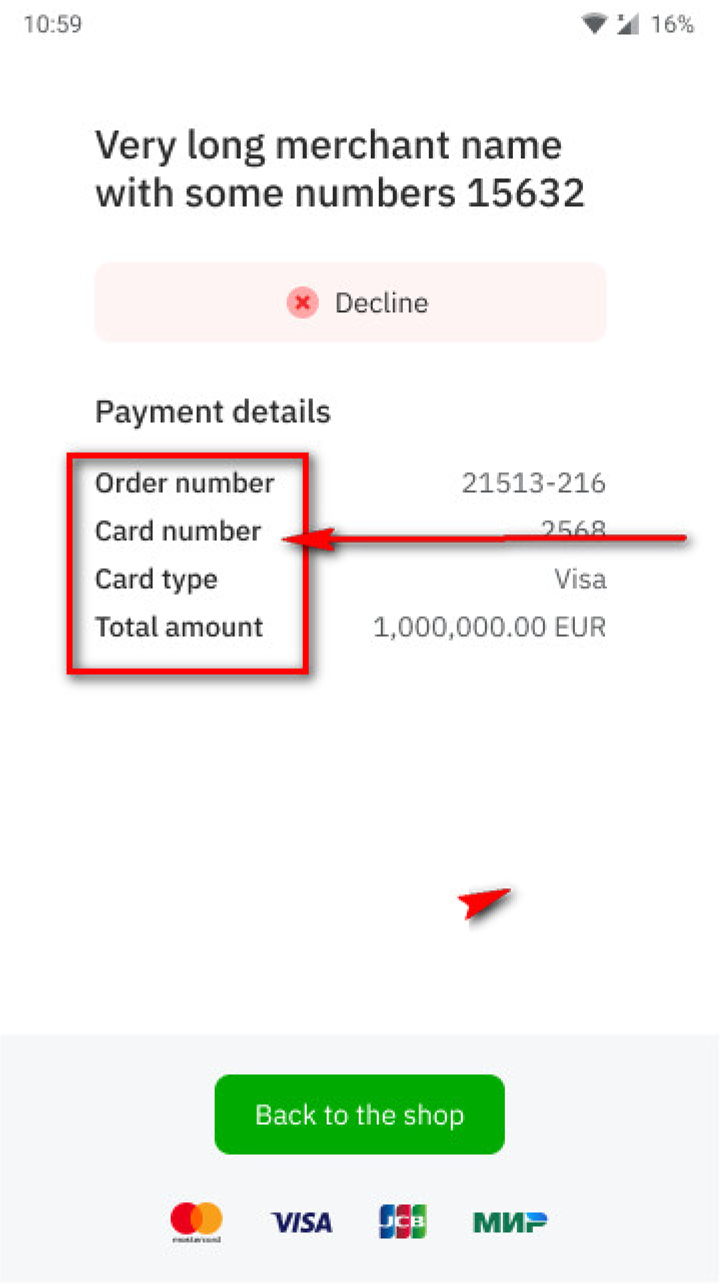
☑ Save for future use

Total 9,700.00 USD

Order #21513-216

You can customize list title or list item style

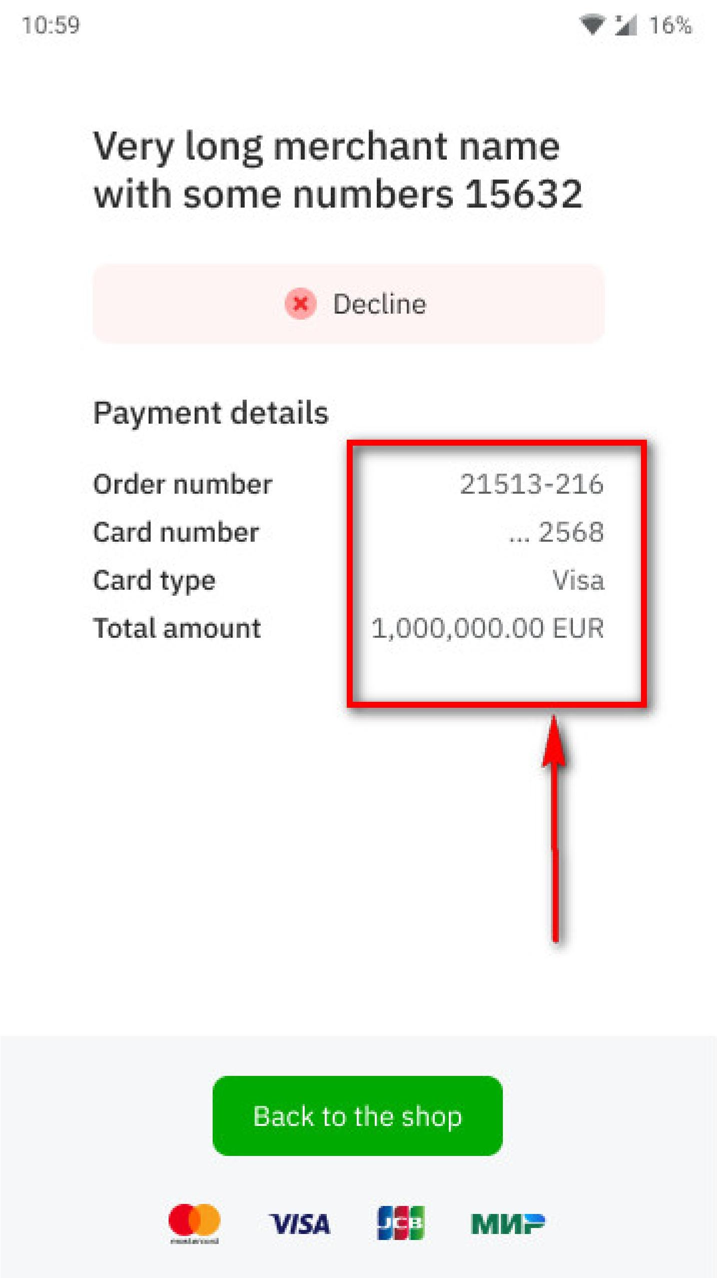
```
<attr name="cp_sdk_list_title_style" format="reference" />
<attr name="cp_sdk_list_item_style" format="reference" />
```



Very long merchant name with some numbers 15632

Payment details

| | |
|---|---|
| Order number | 21513-216 |
| Card number | 2568 |
| Card type | Visa |
| Total amount | 1,000,000.00 EUR |



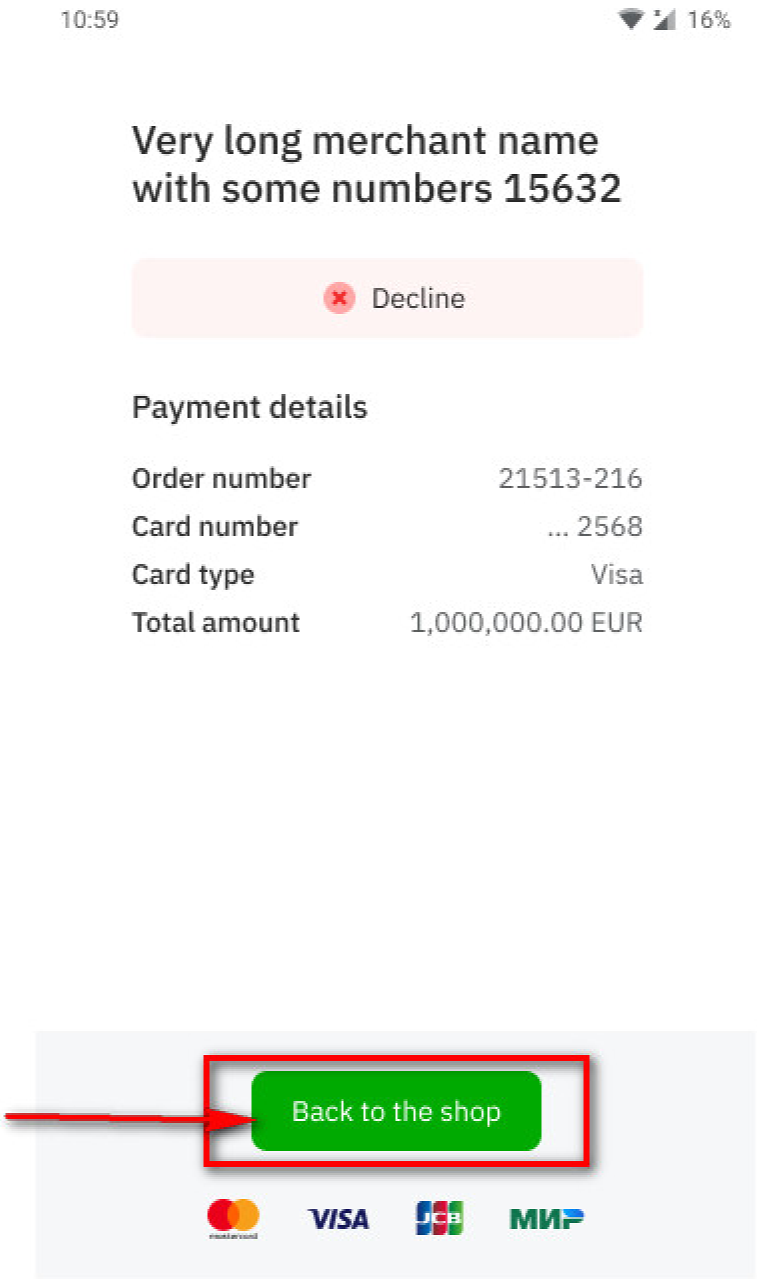
Very long merchant name with some numbers 15632

Payment details

| | |
|---|---|
| Order number | 21513-216 |
| Card number | ... 2568 |
| Card type | Visa |
| Total amount | 1,000,000.00 EUR |

2. Buttons styles: you can customize style of buttons

```
<attr name="cp_sdk_button_style" format="reference" />
```



Very long merchant name with some numbers 15632

Decline

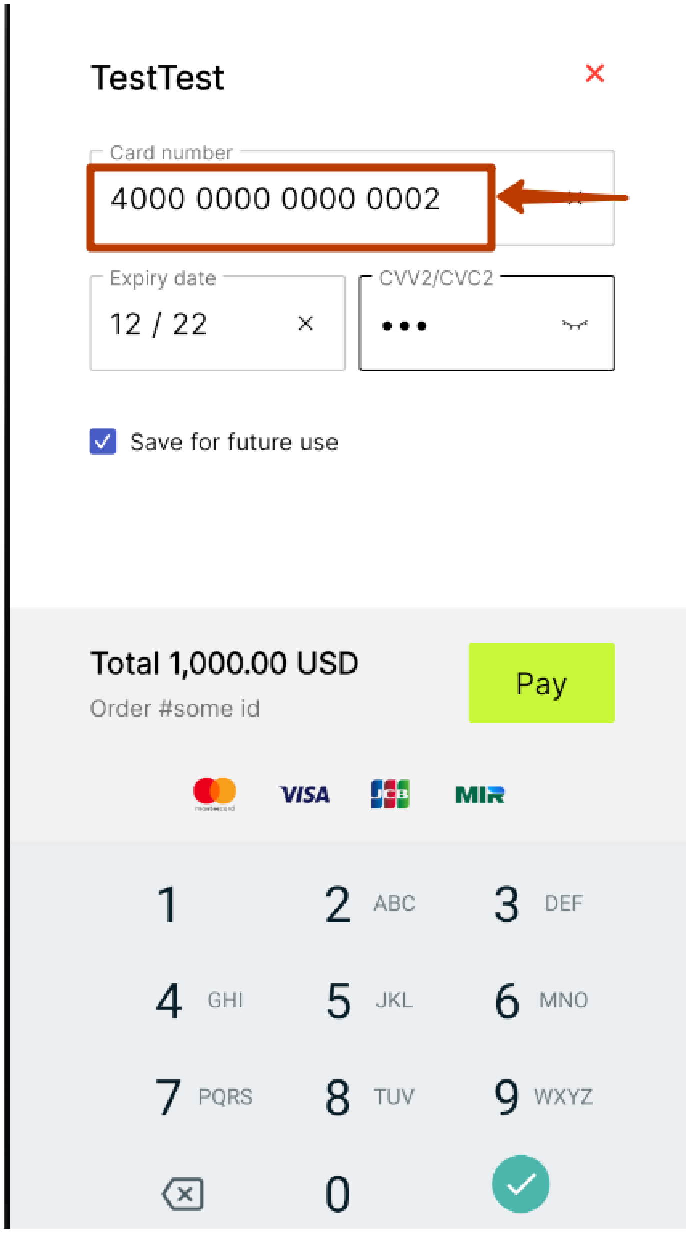
Payment details

| | |
|---|---|
| Order number | 21513-216 |
| Card number | ... 2568 |
| Card type | Visa |
| Total amount | 1,000,000.00 EUR |

3. Customization of input layouts styles

You can customize input layout style, fonts, elements look

```
<attr name="cp_sdk_text_input_layout_style" format="reference" />
```



TestTest

Card number
4000 0000 0000 0002

Expiry date
12 / 22

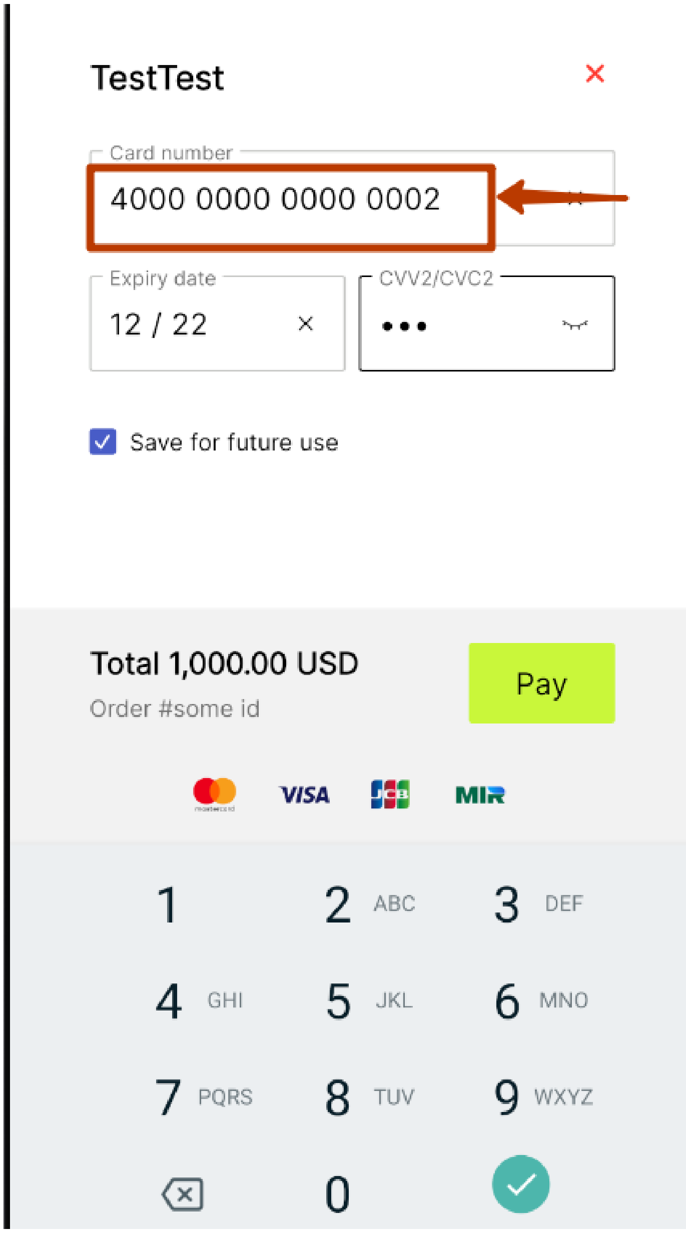
CVV2/CVC2
•••

☑ Save for future use

Total 1,000.00 USD
Order #some id

You can customize input layout edit text

```
<attr name="cp_sdk_text_input_edittext_style" format="reference" />
```



TestTest

Card number: 4000 0000 0000 0002

Expiry date: 12 / 22

CVV2/CVC2: •••

☑ Save for future use

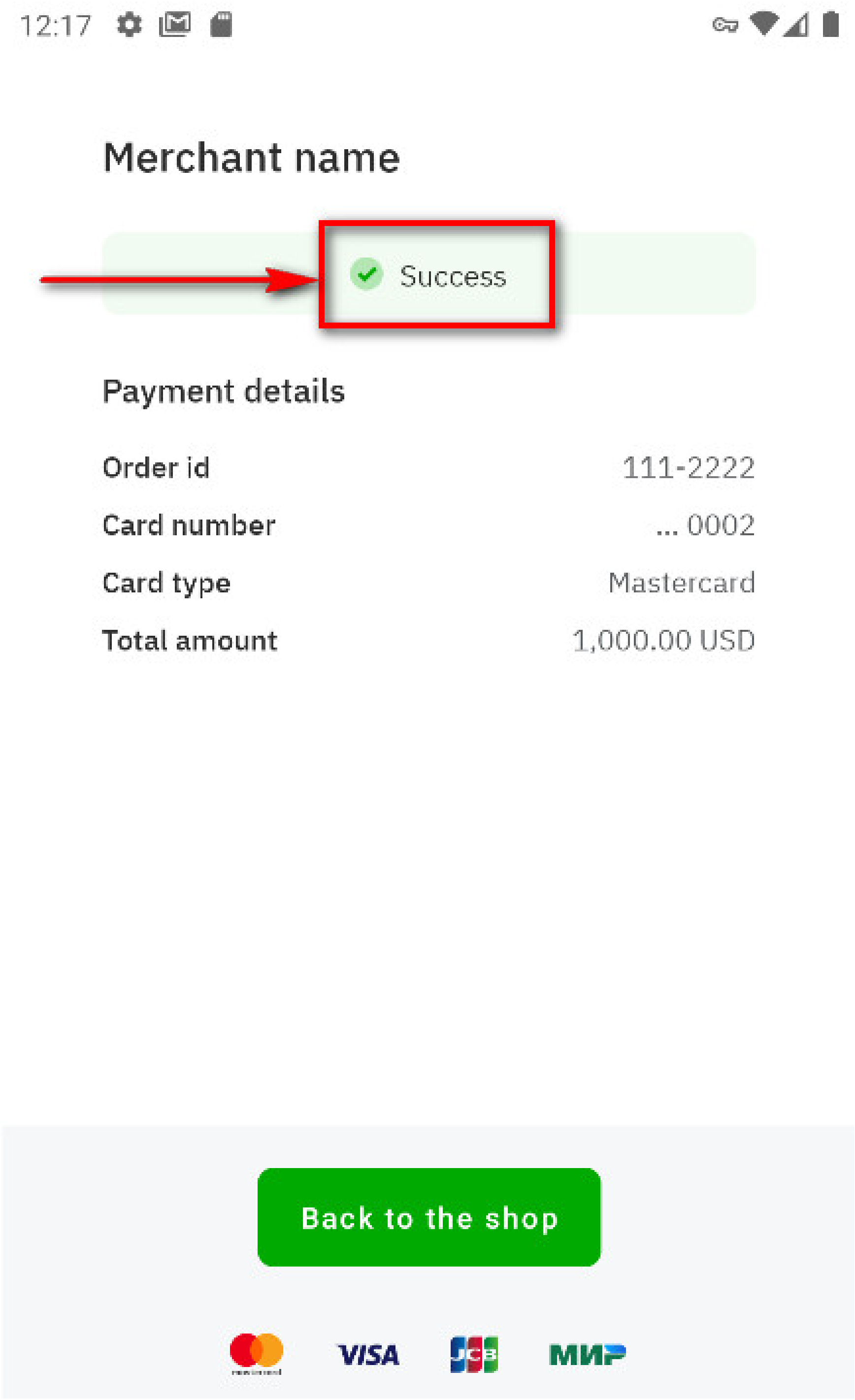
Total 1,000.00 USD

Order #some id

4. Payment messages styles

You can customize payment messages success and decline styles

```
<attr name="cp_sdk_payment_message_success_style" format="reference" />
```



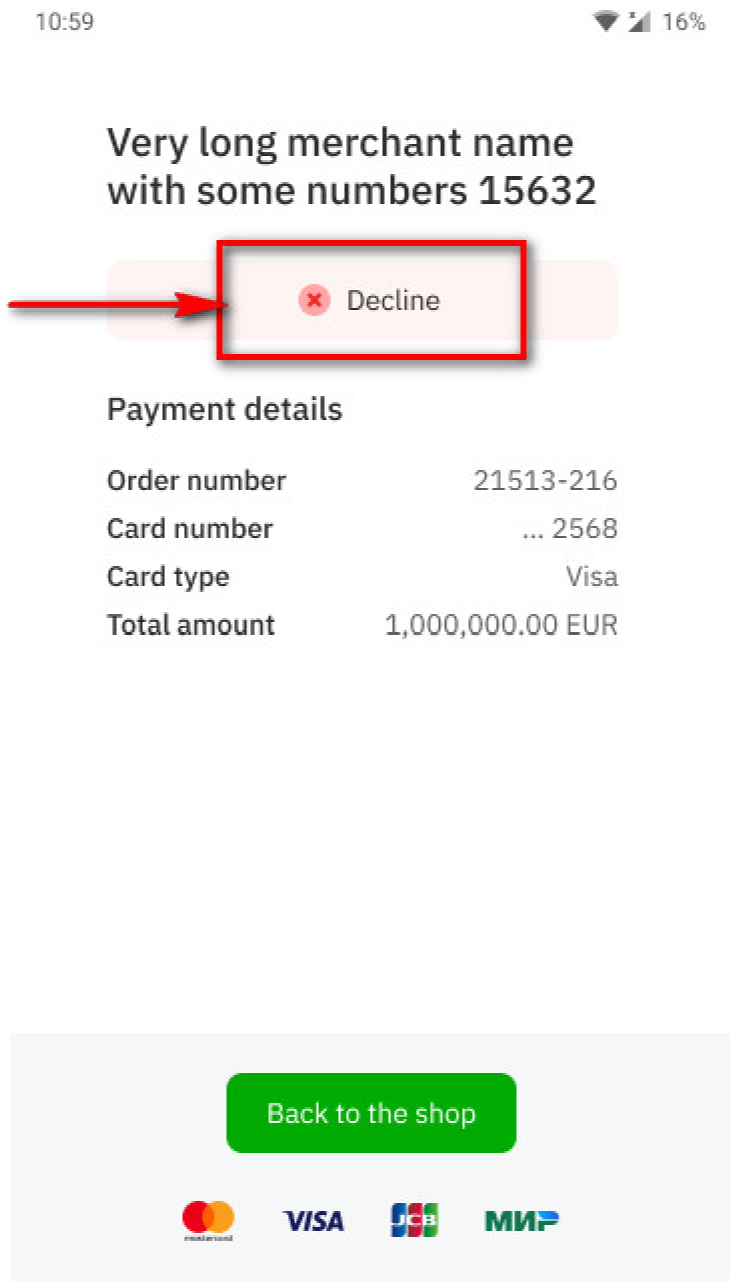
Merchant name

✓ Success

Payment details

| | |
|---|---|
| Order id | 111-2222 |
| Card number | ... 0002 |
| Card type | Mastercard |
| Total amount | 1,000.00 USD |

```
<attr name="cp_sdk_payment_message_decline_style" format="reference" />
```



Very long merchant name with some numbers 15632

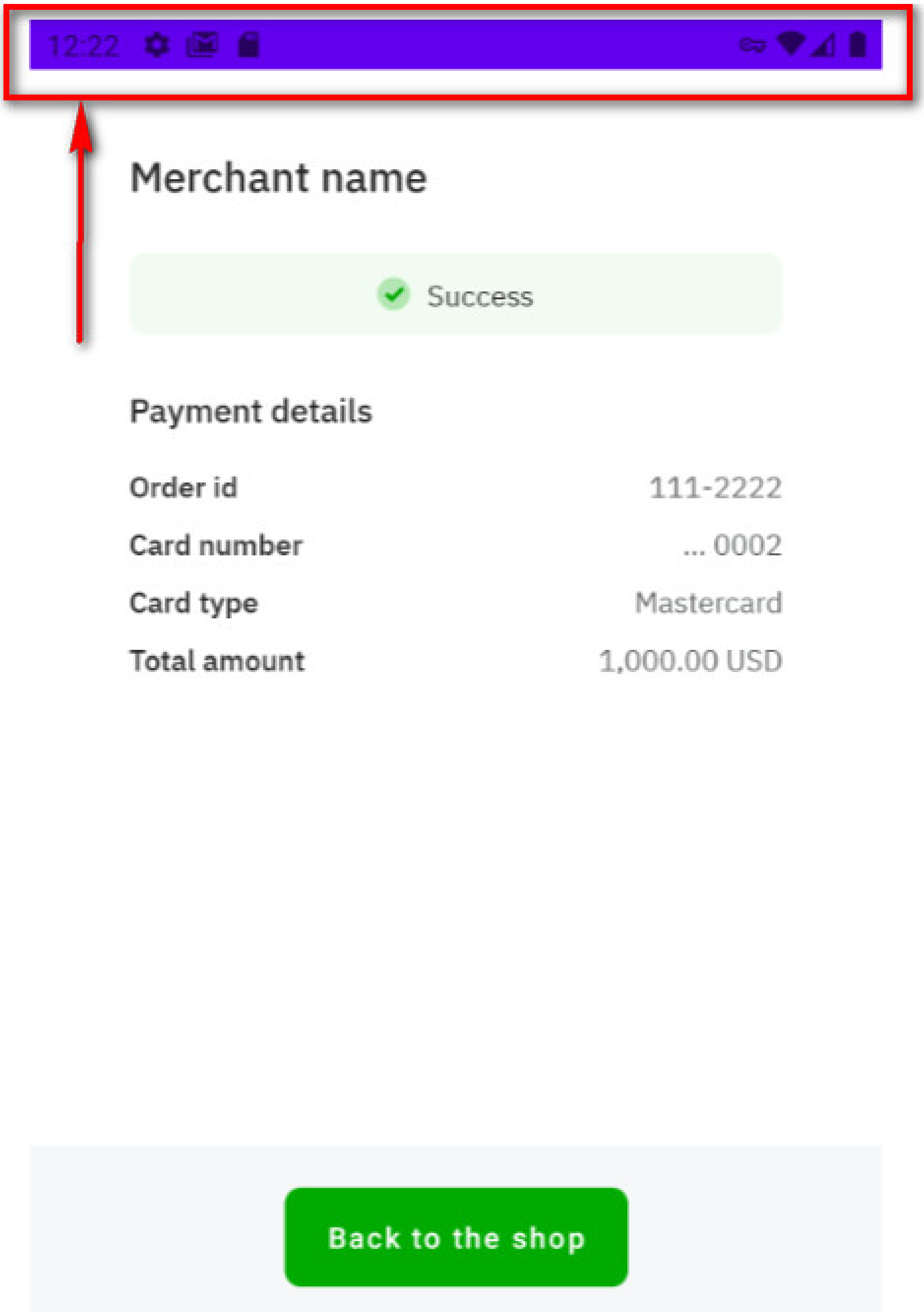
Payment details

| | |
|---|---|
| Order number | 21513-216 |
| Card number | ... 2568 |
| Card type | Visa |
| Total amount | 1,000,000.00 EUR |

5. Status bar, bottom area, background, text colors

You can customize status bar color

```
<attr name="cp_sdk_status_bar_color" format="color" />
```



Merchant name

✓ Success

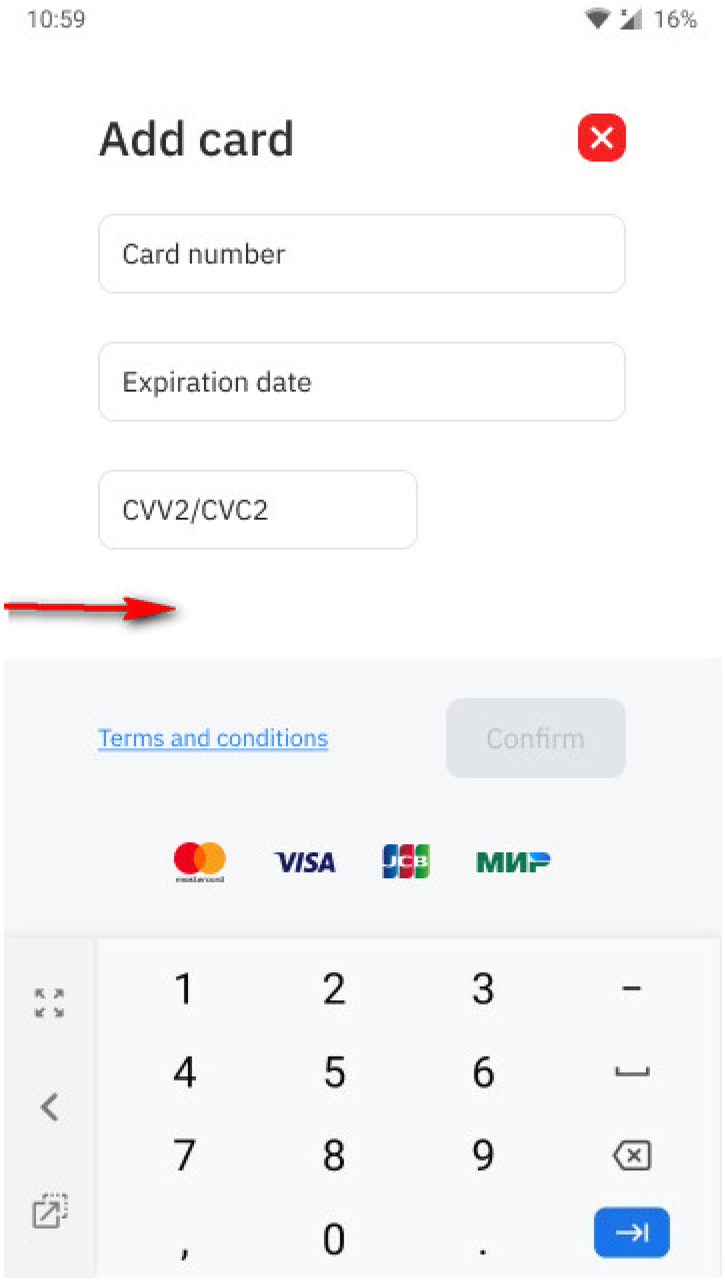
Payment details

| | |
|---|---|
| Order id | 111-2222 |
| Card number | ... 0002 |
| Card type | Mastercard |
| Total amount | 1,000.00 USD |

Also you can customize bottom and backgroud area colors

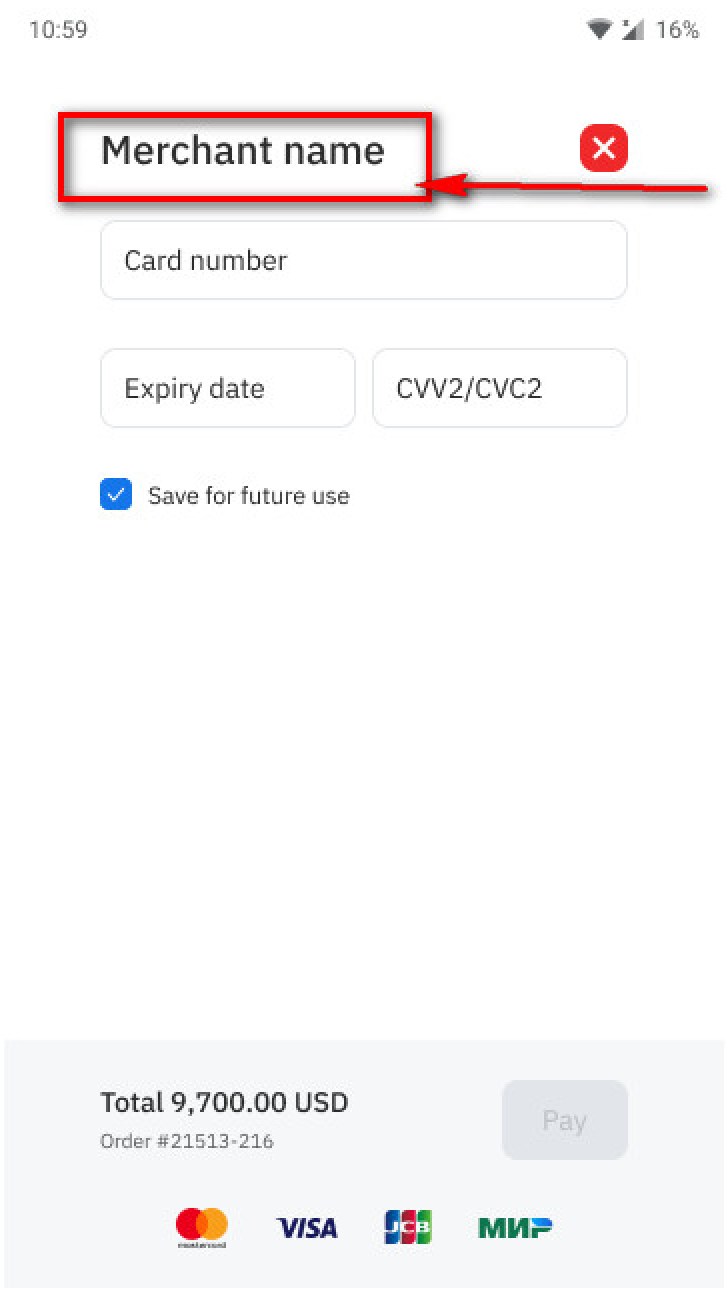
```
<attr name="cp_sdk_bottom_area_color" format="color" />


<attr name="cp_sdk_background" format="color" />
```



And you can customize title and text colors in general

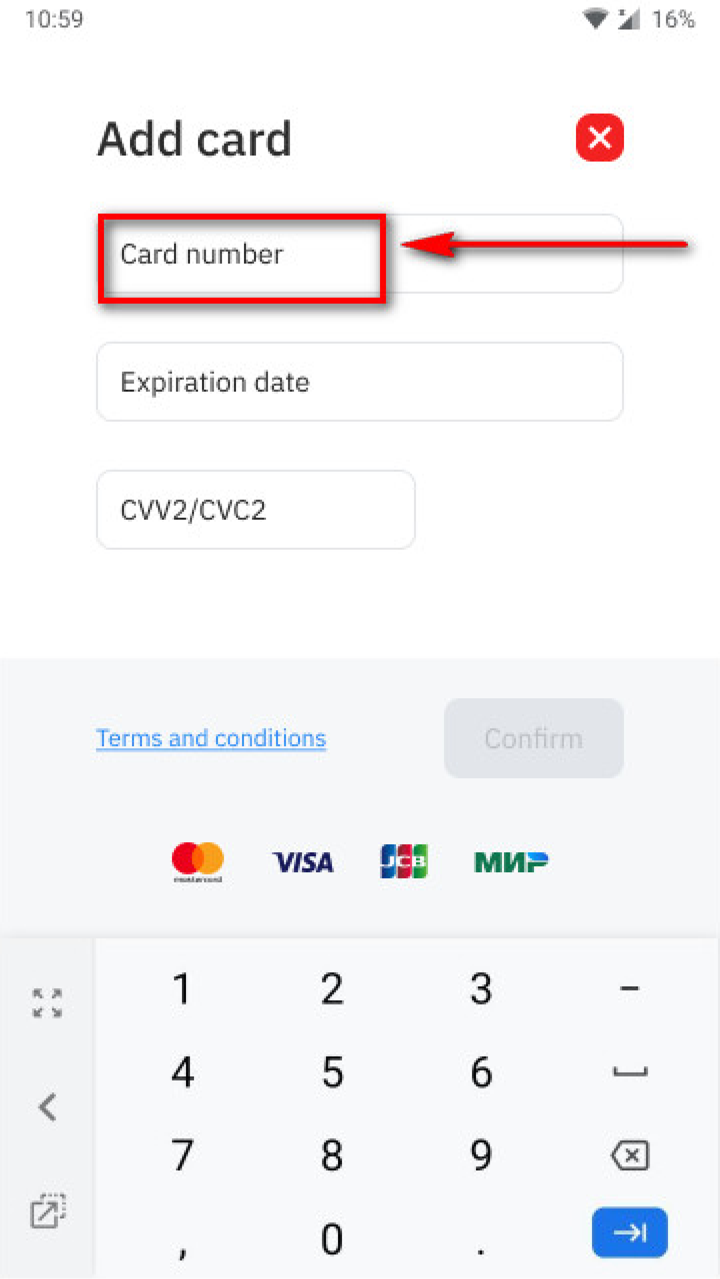
```
<attr name="cp_sdk_title_color" format="color" />
```



Total 9,700.00 USD

Order #21513-216

```
<attr name="cp_sdk_text_color" format="color" />
```



10:59

16%

Add card

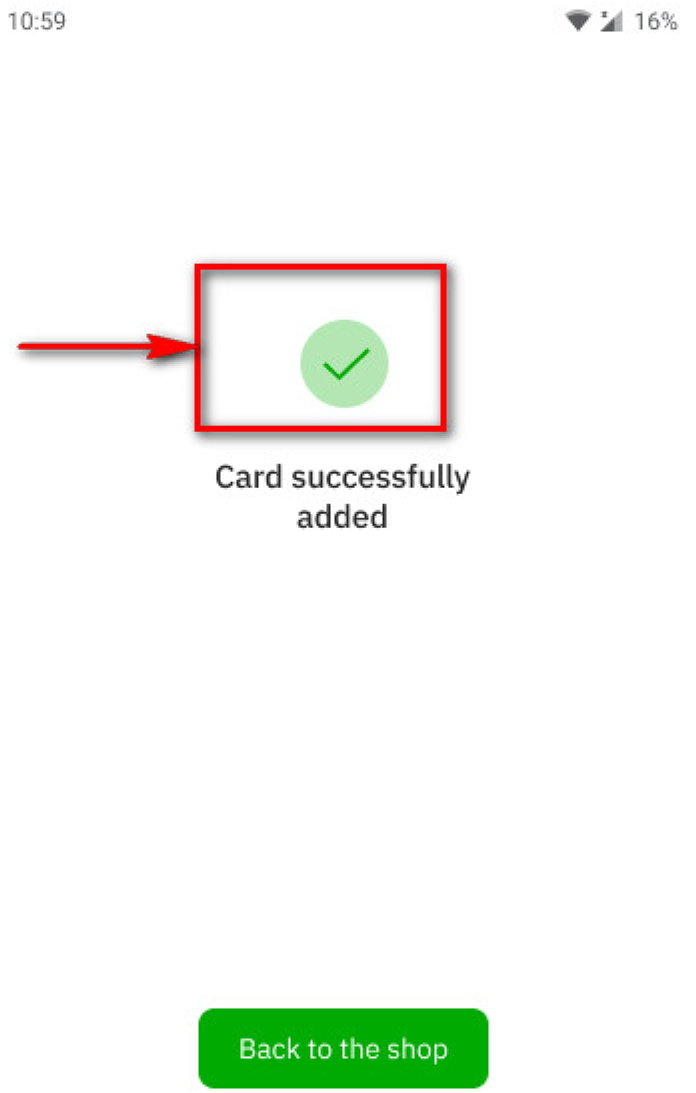
Card number

Expiration date

CVV2/CVC2

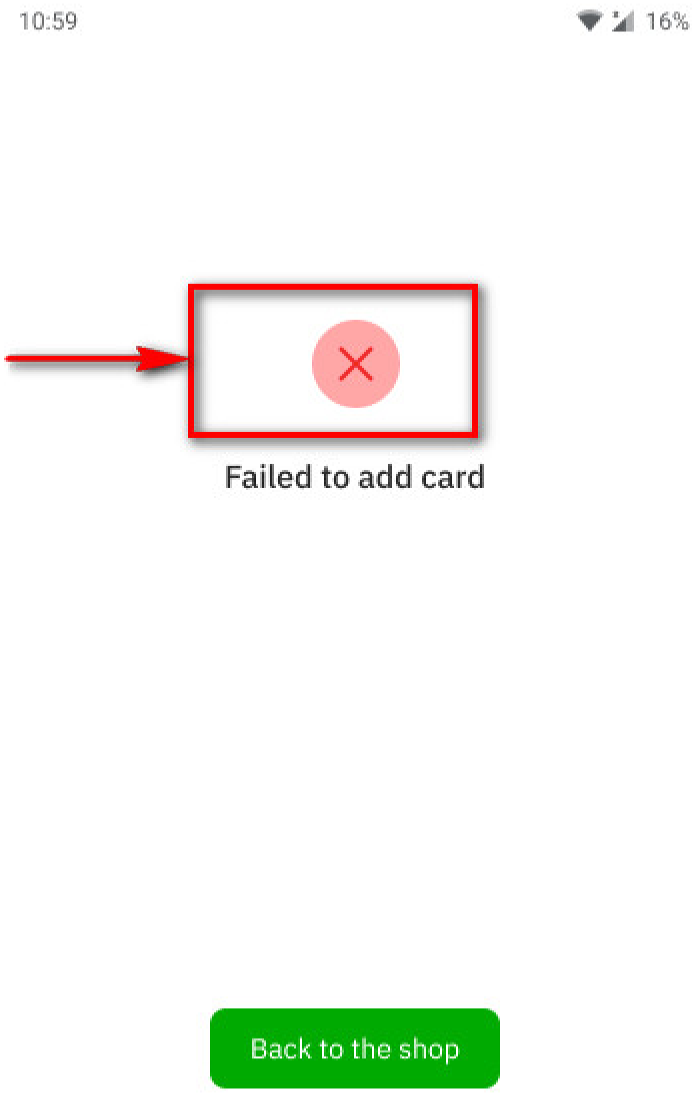
6. Card binding elements customization

```
<attr name="cp_sdk_binding_success_drawable" format="reference" />
```



Card successfully added

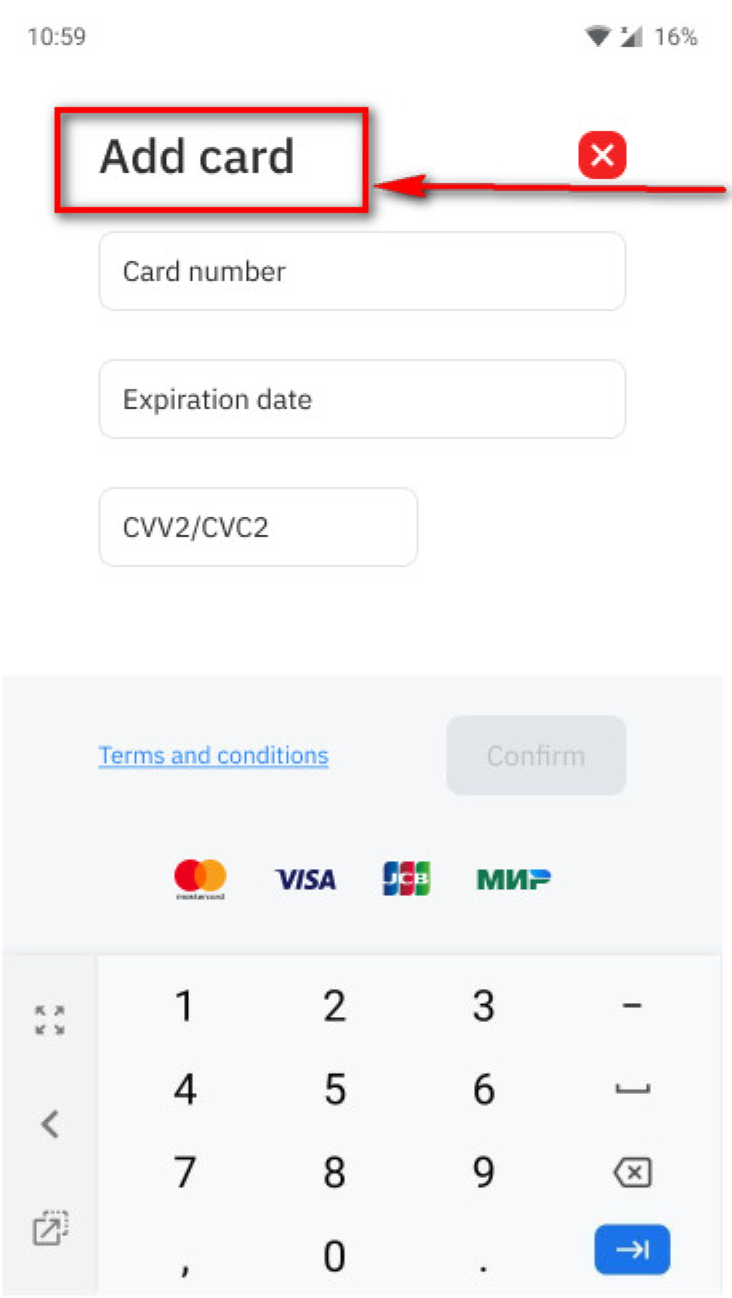
```
<attr name="cp_sdk_binding_fail_drawable" format="reference" />
```



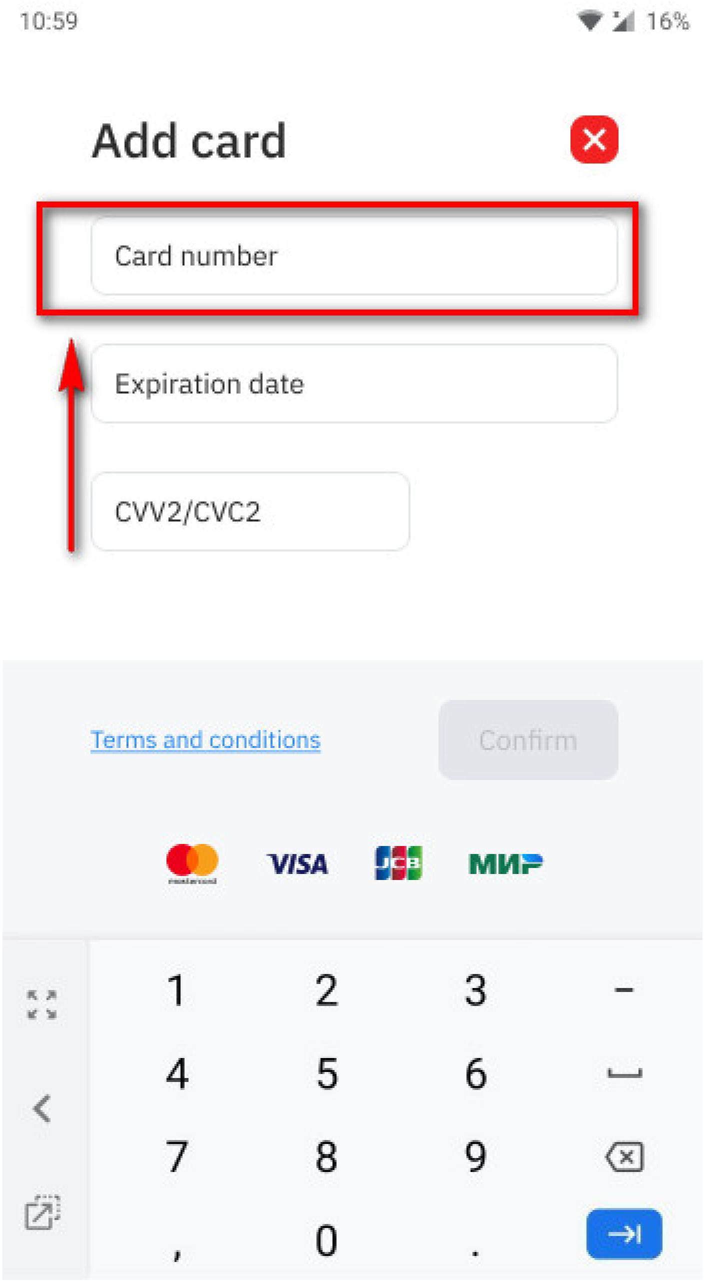
Failed to add card

7. Title and regular fonts customization

```
<attr name="cp_sdk_title_font" format="reference" />
```


10:59
16%
Add card
Card number
Expiration date
CVV2/CVC2
Terms and conditions
Confirm
mastercard
VISA
JCB
МИР
1
2
3
–
4
5
6
7
8
9
,
0
.

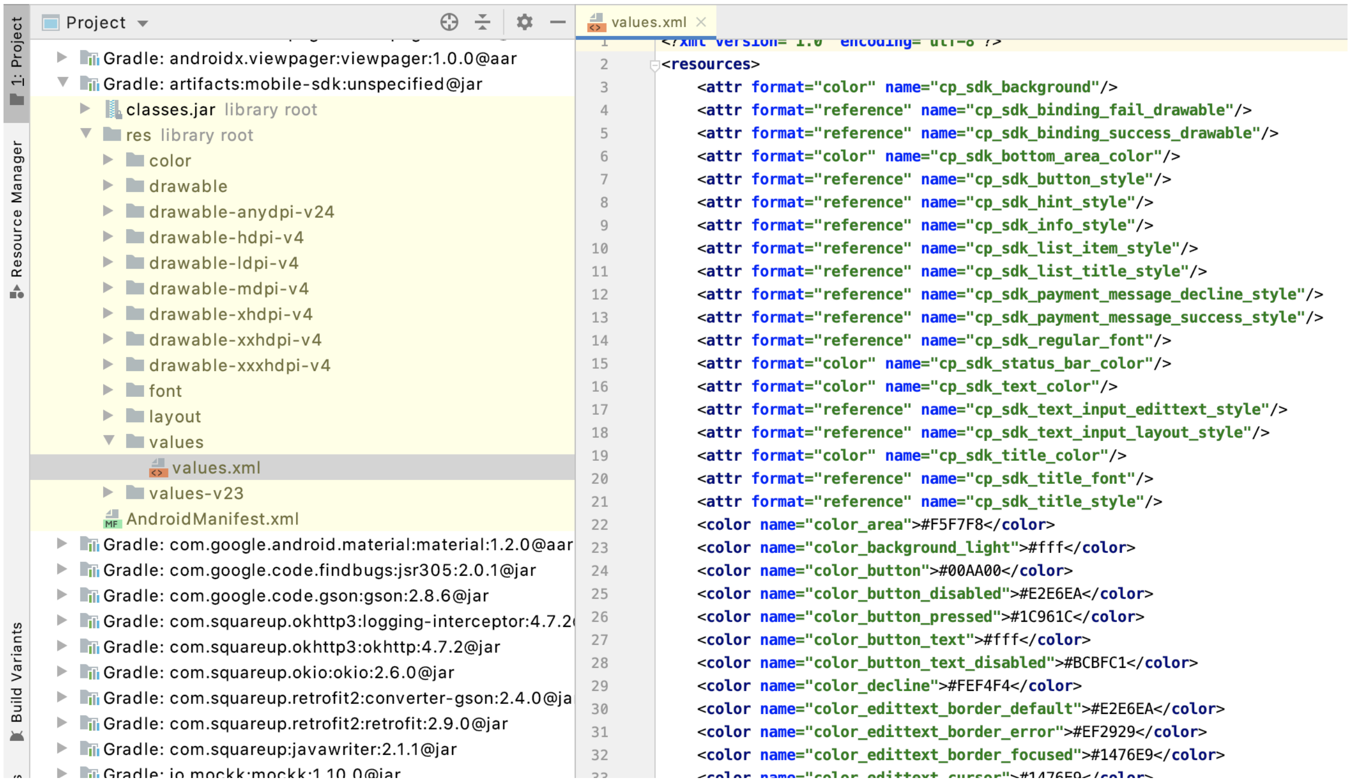
```
<attr name="cp_sdk_regular_font" format="reference" />
```



### Change the look of each component

If you want to customize the components beyond what is specified in the theme, find the component that you want to customize and look at the **layout** and **values** directories of the Support SDK. These directories contain all of the source for the SDK layouts and styles. To find these directories:

1. Include our SDK in your **build.gradle** file
2. Build your project.
3. View the project structure in Project view in Android Studio. This means using the file tree pane on the left side and selecting **Project** from the top drop-down menu, which defaults to **Android**.



```
▶ Gradle: androidx.viewpager:viewpager:1.0.0@aar
▼ Gradle: artifacts:mobile-sdk:unspecified@jar
    ▶ classes.jar library root
    ▼ res library root
        ▶ color
        ▶ drawable
        ▶ drawable-anydpi-v24
        ▶ drawable-hdpi-v4
        ▶ drawable-ldpi-v4
        ▶ drawable-mdpi-v4
        ▶ drawable-xhdpi-v4
        ▶ drawable-xxhdpi-v4
        ▶ drawable-xxxhdpi-v4
        ▶ font
        ▶ layout
        ▼ values
              values.xml
        ▶ values-v23
      AndroidManifest.xml
▶ Gradle: com.google.android.material:material:1.2.0@aar
▶ Gradle: com.google.code.findbugs:jsr305:2.0.1@jar
▶ Gradle: com.google.code.gson:gson:2.8.6@jar
▶ Gradle: com.squareup.okhttp3:logging-interceptor:4.7.2
▶ Gradle: com.squareup.okhttp3:okhttp:4.7.2@jar
▶ Gradle: com.squareup.okio:okio:2.6.0@jar
▶ Gradle: com.squareup.retrofit2:converter-gson:2.4.0@ja
▶ Gradle: com.squareup.retrofit2:retrofit:2.9.0@jar
▶ Gradle: com.squareup:javawriter:2.1.1@jar
▶ Gradle: io.mockk:mockk:1.10.0@jar
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <attr format="color" name="cp_sdk_background"/>
    <attr format="reference" name="cp_sdk_binding_fail_drawable"/>
    <attr format="reference" name="cp_sdk_binding_success_drawable"/>
    <attr format="color" name="cp_sdk_bottom_area_color"/>
    <attr format="reference" name="cp_sdk_button_style"/>
    <attr format="reference" name="cp_sdk_hint_style"/>
    <attr format="reference" name="cp_sdk_info_style"/>
    <attr format="reference" name="cp_sdk_list_item_style"/>
    <attr format="reference" name="cp_sdk_list_title_style"/>
    <attr format="reference" name="cp_sdk_payment_message_decline_style"/>
    <attr format="reference" name="cp_sdk_payment_message_success_style"/>
    <attr format="reference" name="cp_sdk_regular_font"/>
    <attr format="color" name="cp_sdk_status_bar_color"/>
    <attr format="color" name="cp_sdk_text_color"/>
    <attr format="reference" name="cp_sdk_text_input_edittext_style"/>
    <attr format="reference" name="cp_sdk_text_input_layout_style"/>
    <attr format="color" name="cp_sdk_title_color"/>
    <attr format="reference" name="cp_sdk_title_font"/>
    <attr format="reference" name="cp_sdk_title_style"/>
    <color name="color_area">#F5F7F8</color>
    <color name="color_background_light">#fff</color>
    <color name="color_button">#00AA00</color>
    <color name="color_button_disabled">#E2E6EA</color>
    <color name="color_button_pressed">#1C961C</color>
    <color name="color_button_text">#fff</color>
    <color name="color_button_text_disabled">#BCBFC1</color>
    <color name="color_decline">#FEF4F4</color>
    <color name="color_edittext_border_default">#E2E6EA</color>
    <color name="color_edittext_border_error">#EF2929</color>
    <color name="color_edittext_border_focused">#1476E9</color>
    <color name="color_edittext_cursor">#1476E9</color>
```