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

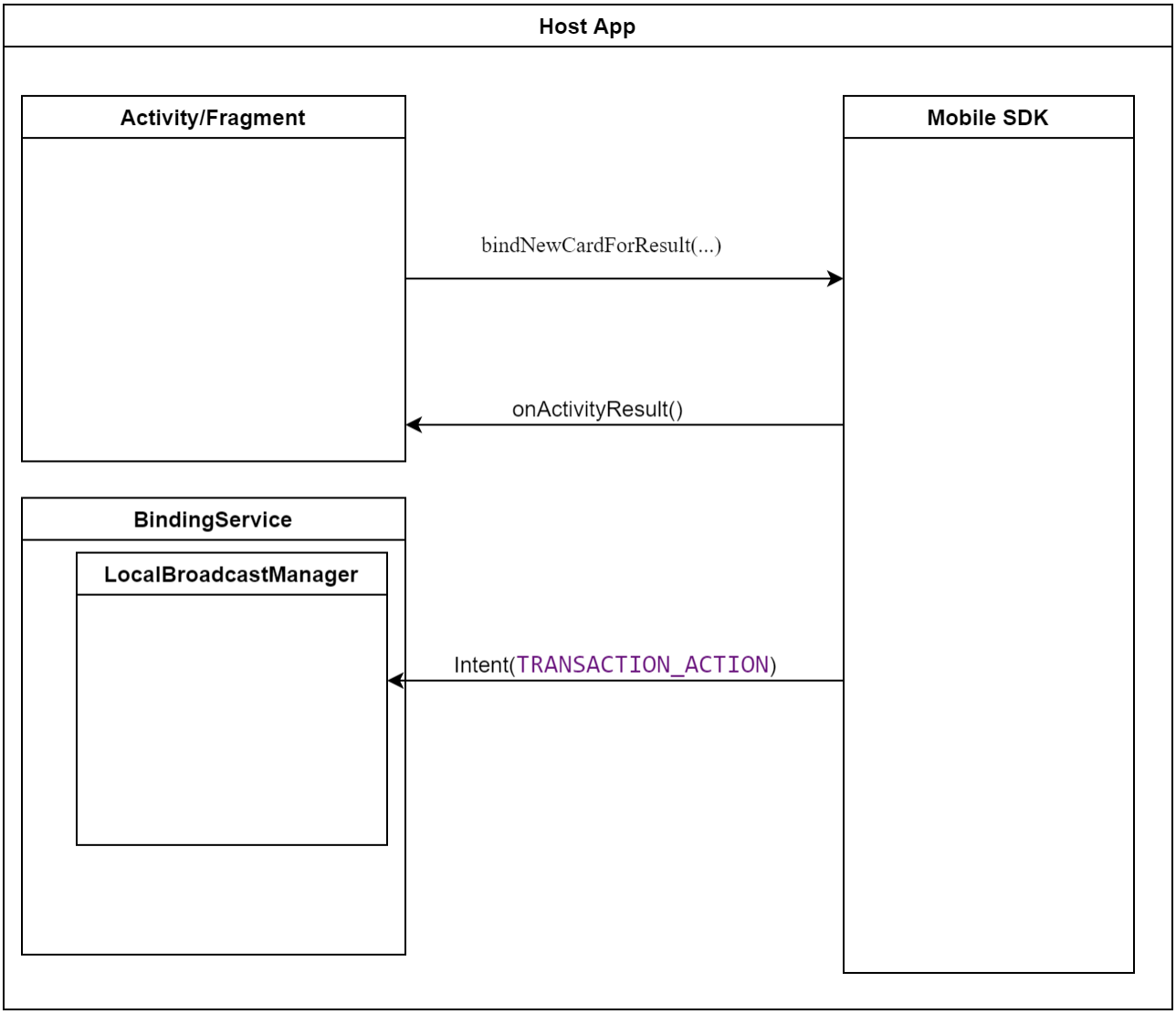
```
<activity
android:name=".sdk.scenario.binding.view.BindActivity"
android:launchMode="singleInstance"
android:screenOrientation="portrait"
android:theme="@style/MobileSdkStyle"
android:windowSoftInputMode="adjustResize" />
```

For mobile payment

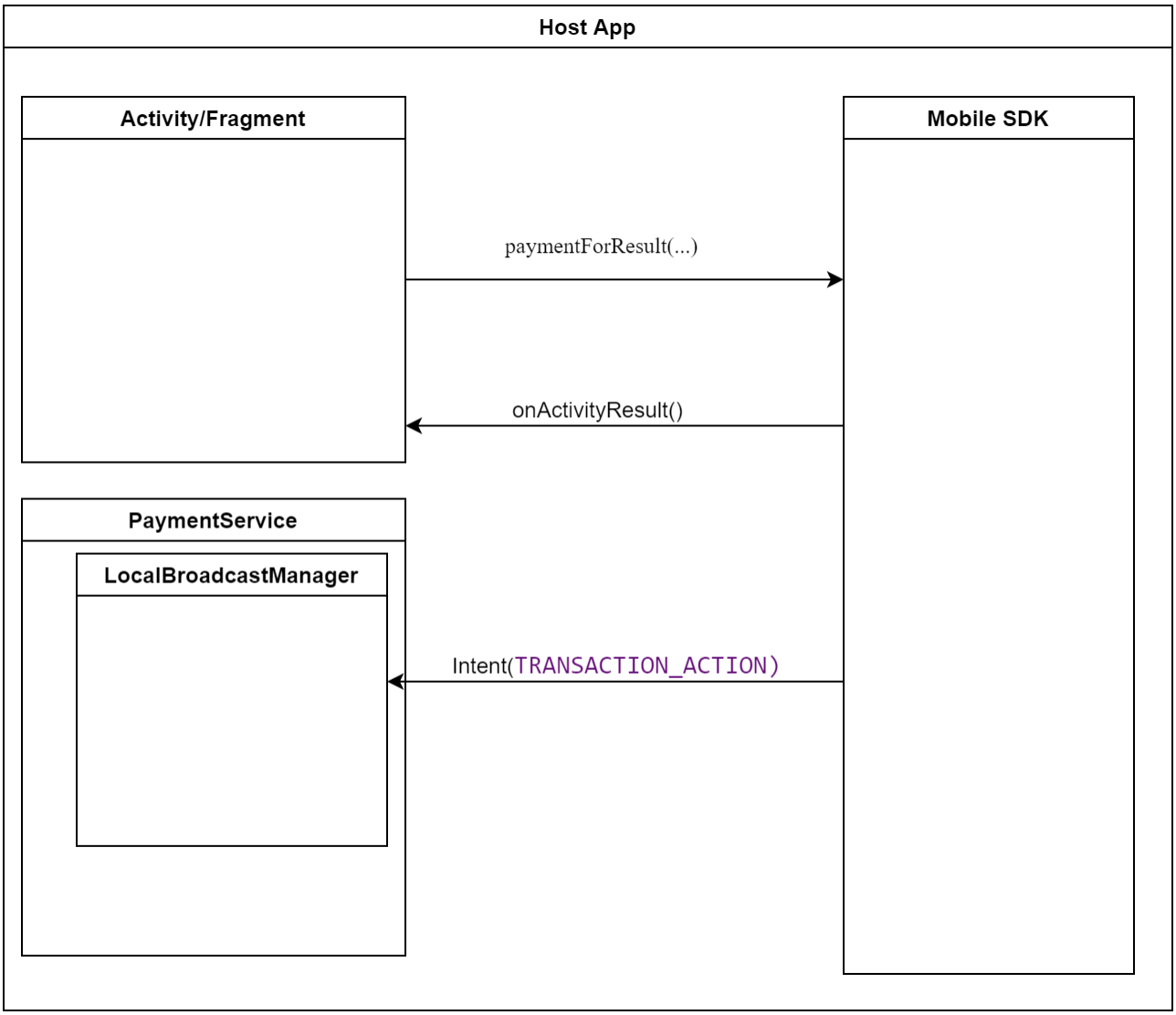
```
<activity
android:name=".sdk.scenario.payment.view.PaymentActivity"
android:launchMode="singleInstance"
android:screenOrientation="portrait"
android:theme="@style/MobileSdkStyle"
android:windowSoftInputMode="adjustResize" />
```

Mobile SDK API description

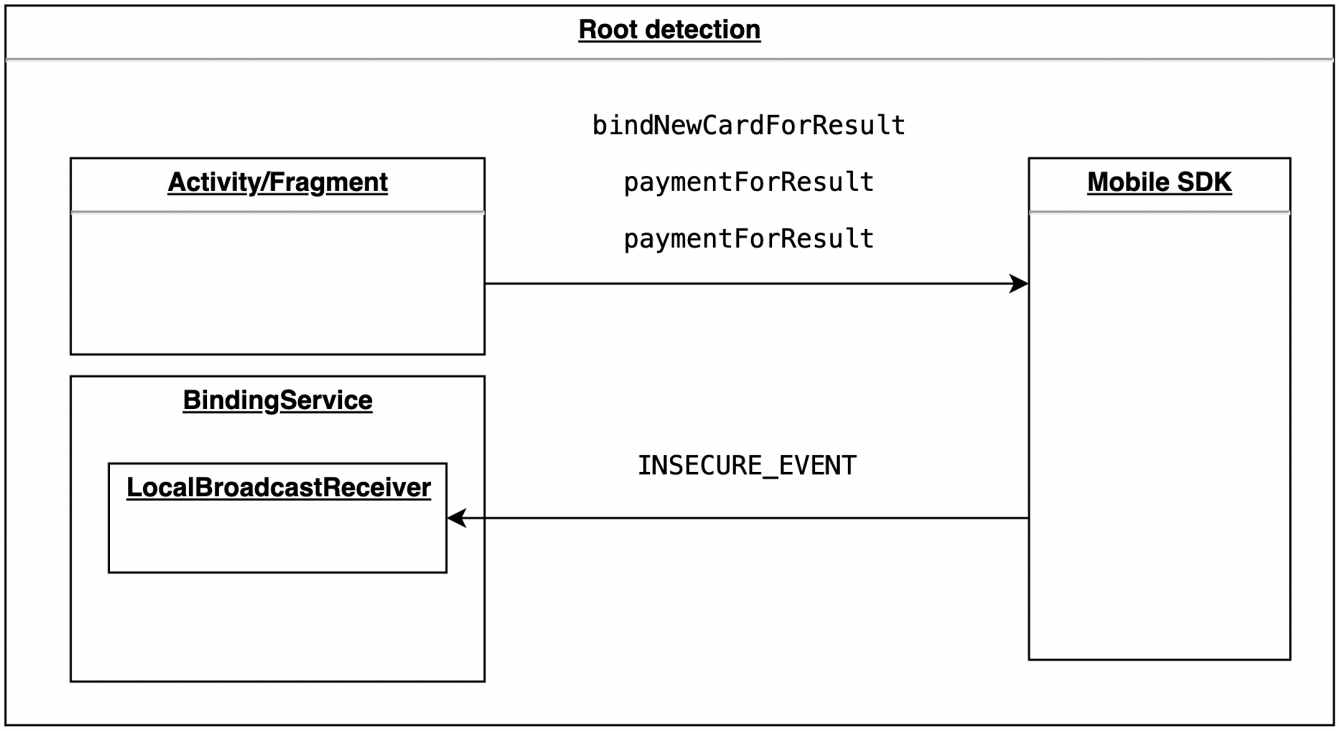
Card binding API description:



Mobile payment API description:



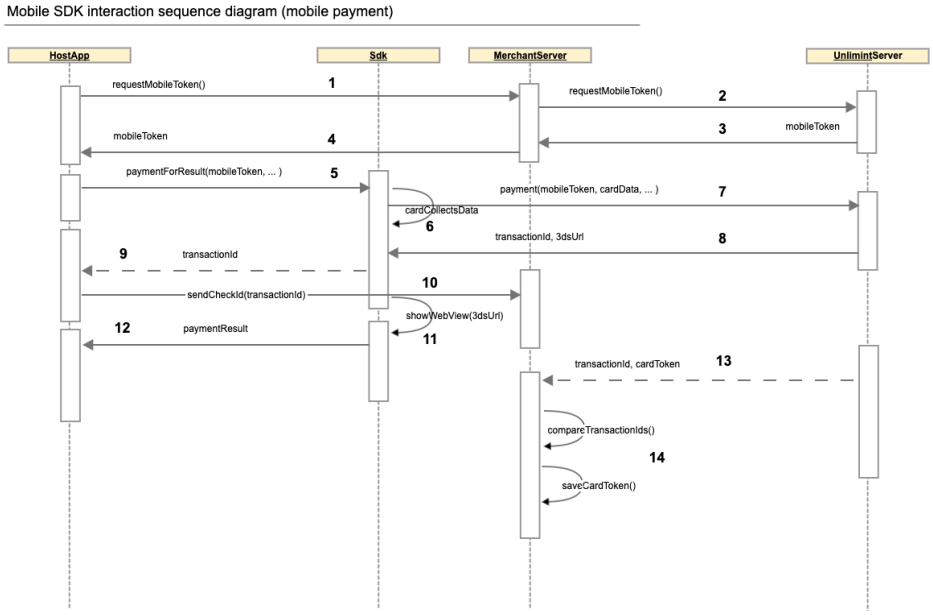
Root detection:



Close SDK:

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

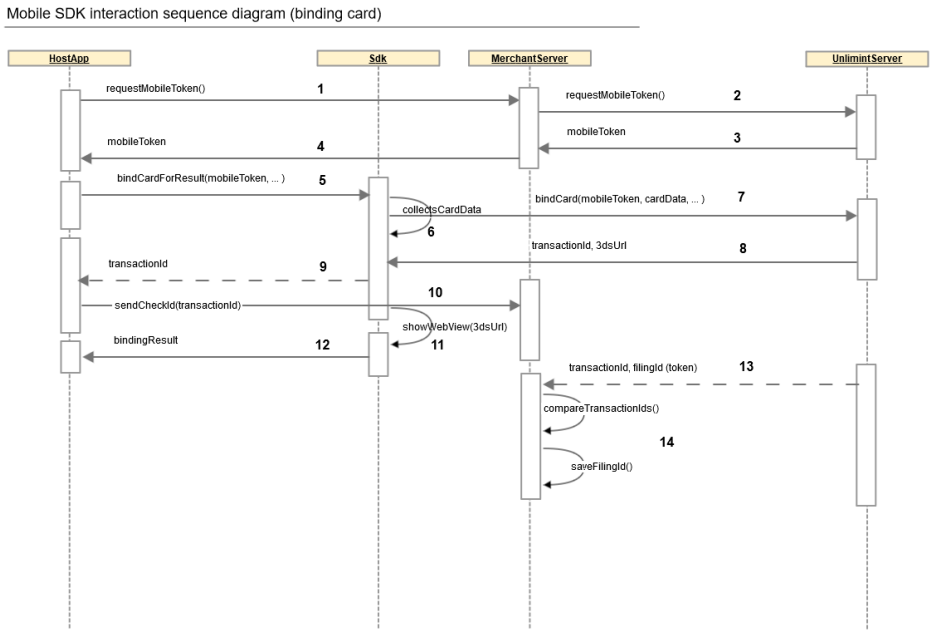
Mobile API SDK interaction



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 Header of request: valid access_token of merchant Parameters of request: 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: **-** mobile_token (string, unique identifier, 128 symbols) **-** expires (date and time of mobile token expiration in ISO 8601 format, example of format - yyyy-MM-dd'T'HH:mm:ss.SSS'Z') Lifetime of mobile token is (see below): 'lifetime of mobile token' <= 5 min and < 'Access_token' ('Bearer' token) life time. Granted LT of access token (for mobile token creation) should be less or equal 5 min but more or equeal than 4 min 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)



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 Header of request: valid access_token of merchant. Parameters of request: 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: - mobile_token (string, unique identifier, 128 symbols) - 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)

Sdk ui castomization

To customize ui use **Theme.MobileSdk.Light** or **Theme.MobileSdk.Dark** theme

There are 4 types castomization is available:

1. Custom style (Title, Subtitle, Button, some views from Payment screens)
2. Custom text colors
3. Custom images

1. Custom style

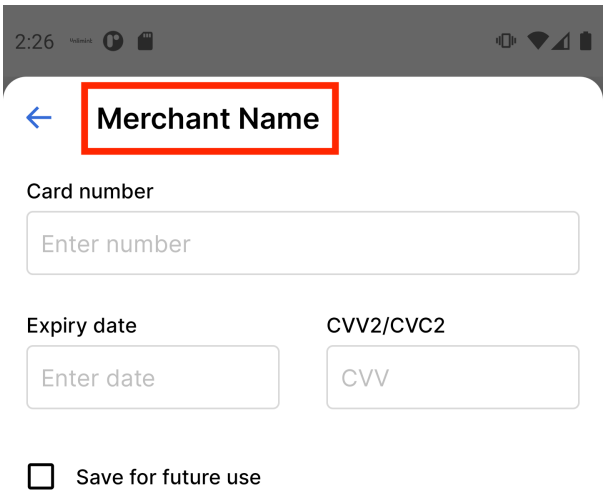
You can customize some view groups on sdk:

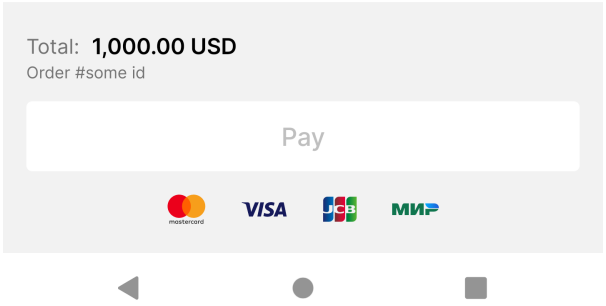
1. Title
2. Subtitle
3. Button
4. Views on screen with input payment credentials
5. Views on screen with payment result

> **Note**
>
> Below will be provided examples of custom styles. You can customize view by attributes in this style. Another attributes can be overridden by xml and them working capacity is not guaranteed

1.1. Title

You can customize title style on each screen where Toolbar is exist





To change title style use this code on your theme.xml file

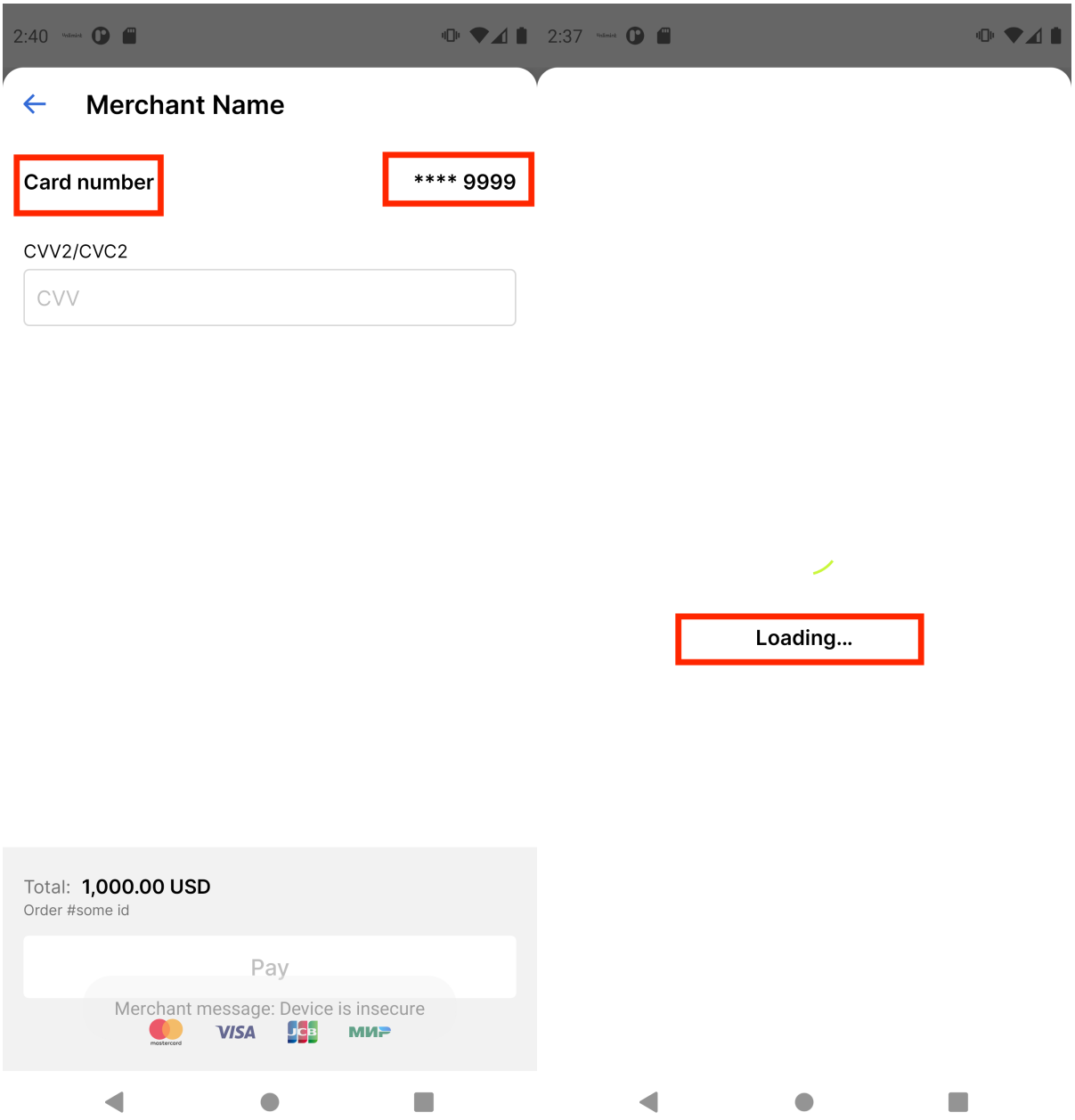
```
    <style name="myTitleStyle">
        <item name="android:fontFamily">@font/inter_400_regular</item>
        <item name="android:textColor">@color/color_payment_total_text</item>
        <item name="android:textSize">@dimen/subhead_14</item>
    </style>

    <style name="AppTheme" parent="Theme.MobileSdk.Light">
        <item name="sdkTitleStyle">@style/myTitleStyle</item>
    </style>
```

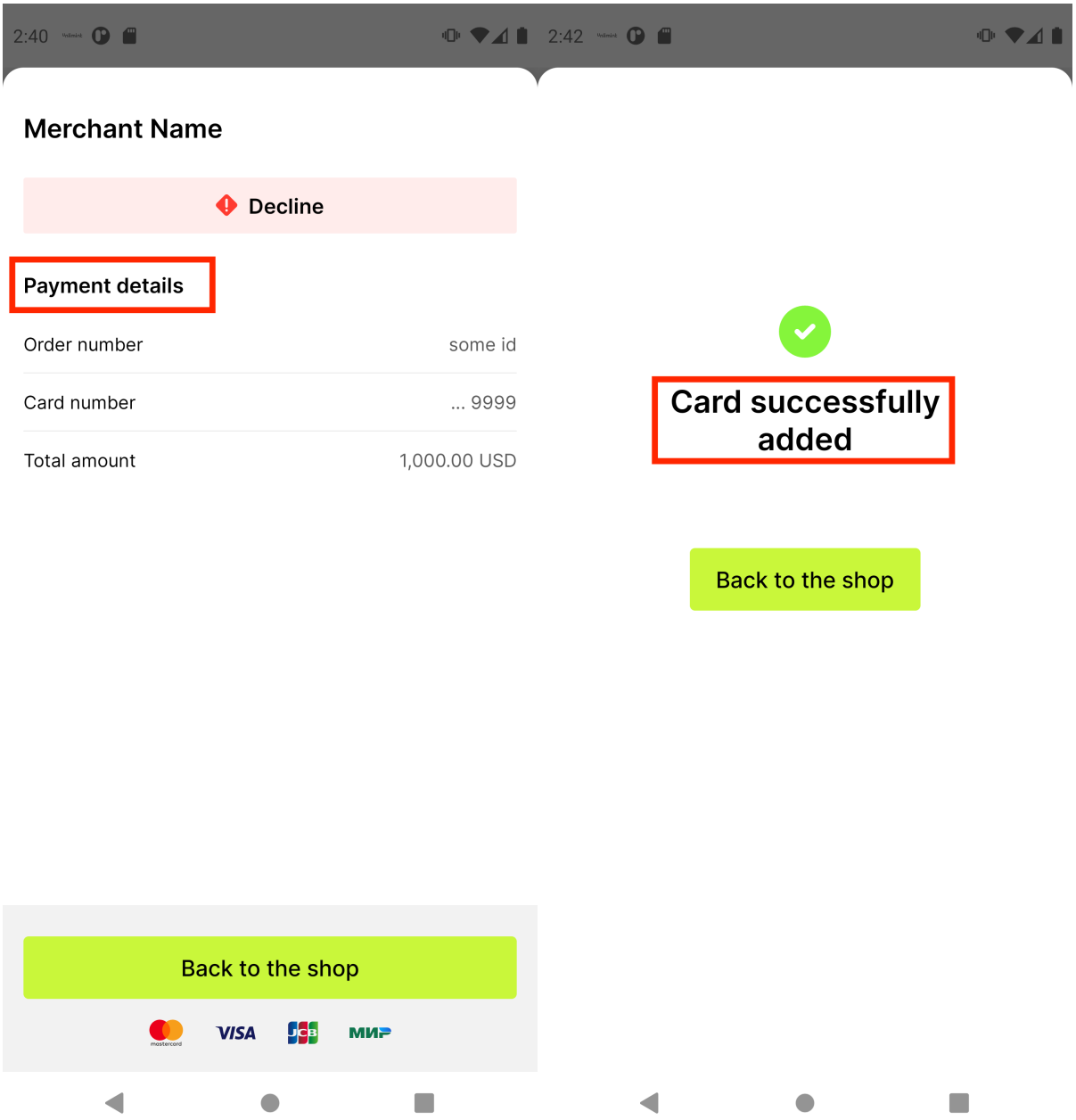
1.2. Subtitle

Some views have subtitle style. By changing this style you can change this views:

1. Views on payment by token (first picture)

2. Text on loading screen (second picture)
3. Text below status message on screen with payment result (third picture)
4. Text with binding result (fourth picture)





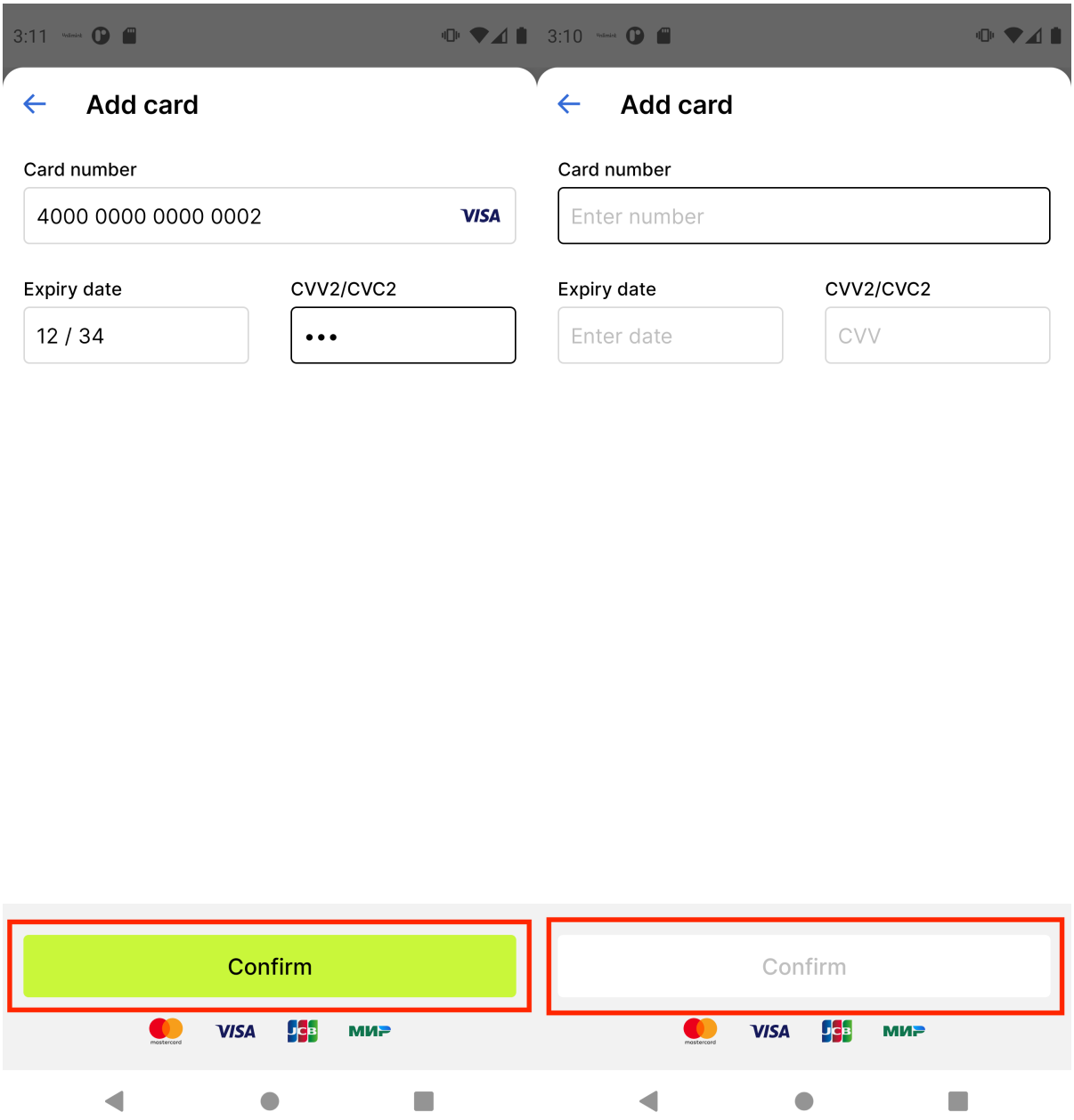
You cannot change these views separately, only **together**

> **Note**
>
> **android:textSize** on screen with binding result (last picture) cannot be changed

```
<style name="mySubtitleStyle">
    <item name="android:fontFamily">@font/inter_600_semi_bold</item>
    <item name="android:textColor">@color/color_payment_total_value</item>
    <item name="android:textSize">@dimen/text_15</item>
    <item name="android:textStyle">bold</item>
</style>

    <style name="AppTheme" parent="Theme.MobileSdk.Light">
    <item name="sdkSubtitleStyle">@style/mySubtitleStyle</item>
</style>
```

1.3. Button

All buttons on ui have the same style. By changing this style, you can change the style of all buttons



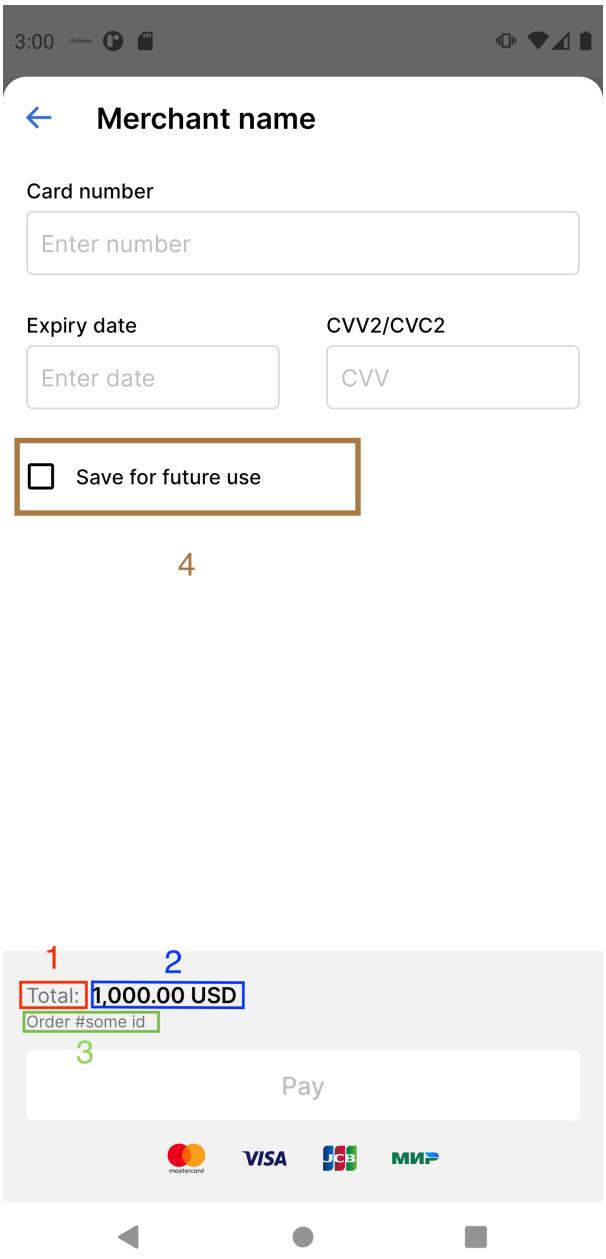
Enabled and disabled colors modifyed by **Button.setEnabled(boolean)** Android SDK method

```
<style name="myButtonStyle" parent="@style/Widget.MaterialComponents.Button">
    <item name="android:background">@drawable/button_ripple</item>
    <item name="android:textColor">@drawable/button_text_selector</item>
    <item name="android:paddingStart">20dp</item>
    <item name="android:paddingEnd">20dp</item>
    <item name="android:paddingTop">10dp</item>
    <item name="android:paddingBottom">10dp</item>
    <item name="android:textSize">@dimen/unlimint_title_17</item>
    <item name="android:textStyle">normal</item>
    <item name="android:fontFamily">@font/inter_500_medium</item>
    <item name="android:textAllCaps">false</item>
</style>

<style name="AppTheme" parent="Theme.MobileSdk.Light">
    <item name="sdkButtonStyle">@style/myButtonStyle</item>
</style>
```

1.4. Views on screen with input payment credentials

On screen with input payment credentials you can customize 4 views:

1. Total title
2. Total text
3. Order text
4. Checkbox (available only on payment with bank card)

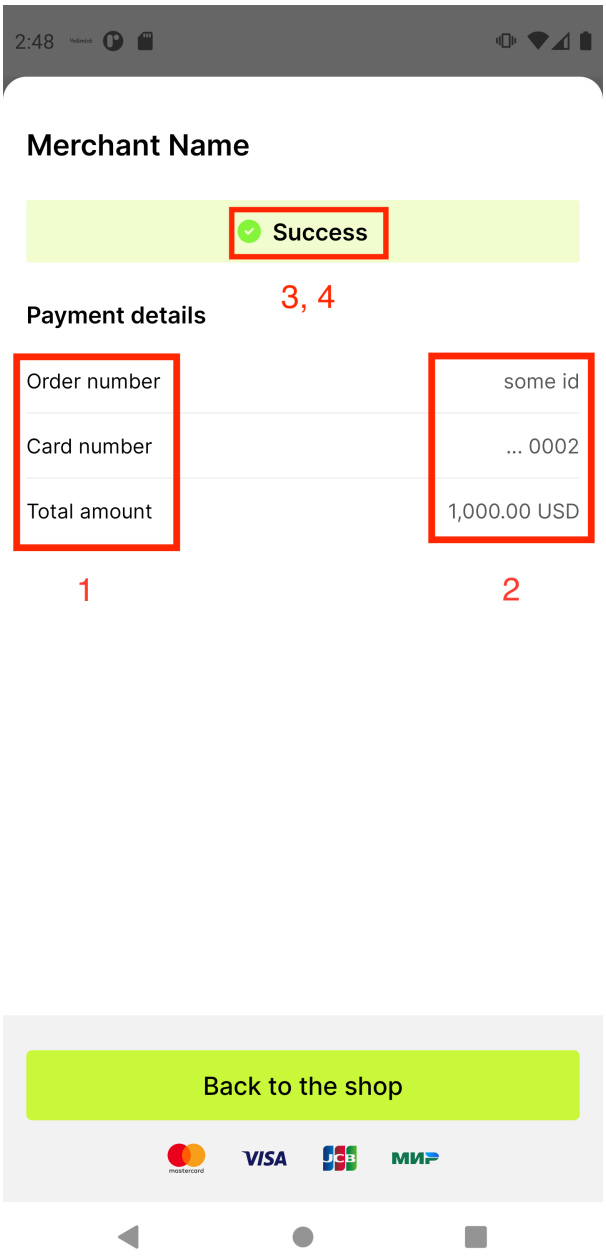

3:00
Merchant name
Card number
Enter number
Expiry date
CVV2/CVC2
Enter date
CVV
Save for future use
4
1
2
Total:
1,000.00 USD
Order #some id
3
Pay
VISA
МИР

```
<style name="myTotalTextStyle">
    <item name="android:fontFamily">@font/inter_400_regular</item>
    <item name="android:textColor">@color/color_payment_total_text</item>
    <item name="android:textSize">@dimen/subhead_14</item>
</style>

<style name="myTotalValueStyle">
    <item name="android:fontFamily">@font/inter_600_semi_bold</item>
    <item name="android:textColor">@color/color_payment_total_value</item>
    <item name="android:textSize">@dimen/text_15</item>
    <item name="android:textStyle">bold</item>
</style>

<style name="myOrderStyle">
    <item name="android:fontFamily">@font/inter_400_regular</item>
    <item name="android:textColor">@color/color_payment_order</item>
    <item name="android:textSize">@dimen/caption_11</item>

<style name="myCheckboxStyle">
    <item name="android:paddingStart">@dimen/space_x1</item>
    <item name="colorControlActivated">@color/color_checkbox</item>
    <item name="android:textSize">@dimen/text_15</item>
    <item name="android:textColor">@color/color_subtitle_text</item>
    <item name="android:fontFamily">@font/inter_500_medium</item>
</style>

<style name="AppTheme" parent="Theme.MobileSdk.Light">
    <item name="sdkTotalTextStyle">@style/myTotalTextStyle</item>  ---- 1 TextView
    <item name="sdkTotalValueStyle">@style/myTotalValueStyle</item> --- 2 TextView
    <item name="sdkOrderStyle">@style/myOrderStyle</item> ------------- 3 TextView
    <item name="sdkCheckboxStyle">@style/myCheckboxStyle</item> ------- 4 CheckBox
</style>
```

1.5. Views on screen with payment result

On screen with payment result you can change 4 views:

1. Titles of payment details
2. Values of payment details
3. Success payment text and icon
4. Decline payment text and icon


2:48
Merchant Name
Success
3, 4
Payment details
Order number
some id
Card number
... 0002
Total amount
1,000.00 USD
1
2
Back to the shop

```
    <style name="myPaymentListTitleStyle">
        <item name="android:fontFamily">@font/inter_400_regular</item>
        <item name="android:textSize">@dimen/subhead_14</item>
        <item name="android:textColor">@color/color_payment_list_title</item>
    </style>

    <style name="myPaymentListItemStyle">
        <item name="android:fontFamily">@font/inter_400_regular</item>
        <item name="android:textSize">@dimen/subhead_14</item>
        <item name="android:textColor">@color/color_payment_list_item</item>
    </style>

    <style name="myPaymentMessageSuccessStyle">
        <item name="android:textColor">@color/color_payment_message</item>
        <item name="android:fontFamily">@font/inter_600_semi_bold</item>
        <item name="android:textSize">@dimen/headline_16</item>
        <item name="android:textStyle">bold</item>
        <item name="android:drawableStart">@drawable/ic_success_small</item>
    </style>

    <style name="myPaymentMessageDeclineStyle">
        <item name="android:textColor">@color/color_payment_message</item>
        <item name="android:fontFamily">@font/inter_600_semi_bold</item>
        <item name="android:textSize">@dimen/headline_16</item>
        <item name="android:textStyle">bold</item>
        <item name="android:drawableStart">@drawable/ic_fail_small</item>
    </style>

    <style name="AppTheme" parent="Theme.MobileSdk.Light">
        <item name="sdkPaymentListTitleStyle">@string/myPaymentListTitleStyle</item>
        <item name="sdkPaymentListItemStyle">@style/myPaymentListItemStyle</item>
        <item name="sdkPaymentMessageSuccessStyle">@style/myPaymentMessageSuccessStyle</item>
        <item name="sdkPaymentMessageDeclineStyle">@style/myPaymentMessageDeclineStyle</item>
    </style>
```

2. Custom text colors

You can change 6 colors on UI:

1. Status bar
2. Background
3. Bottom area
4. Title color
5. Subtitle color
6. colorPrimary, colorAccent

2.1. Status bar color

You can change the status bar color by this attribute

```
    <style name="AppTheme" parent="Theme.MobileSdk.Light">
        <item name="sdkStatusBarColor">@color/myStatusBarColor</item>
    </style>
```

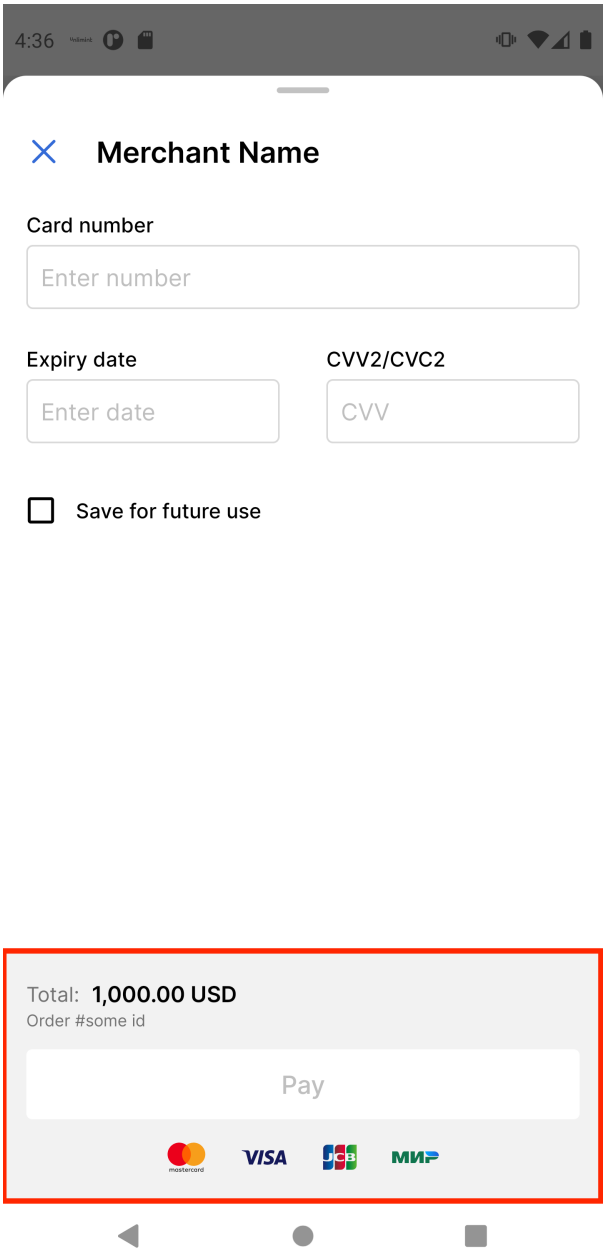
2.2. Background

You can change the background by this attribute

```
    <style name="AppTheme" parent="Theme.MobileSdk.Light">
        <item name="sdkBackgroundColor">@color/myBackgroundColor</item>
    </style>
```

2.3. Bottom area

The color of the bottom view can be changed




```
    <style name="AppTheme" parent="Theme.MobileSdk.Light">
        <item name="sdkBottomAreaColor">@color/myBottomAreaColor</item>
    </style>
```

2.4. Title color

You can change the text color of all views with style for the title (paragraph 1.1) and checkbox (paragraph 1.3). This color used in default styles, this color will be applied only if custom styles for this views is not set

```
    <style name="AppTheme" parent="Theme.MobileSdk.Light">
        <item name="sdkTitleColor">@color/myTitleColor</item>
    </style>
```

2.5. Subtitle color

You can change text color of all views with style for subtitle (paragraph 1.2). This color used in default styles, this color will be applied only if custom styles for this views is not set

```
<style name="AppTheme" parent="Theme.MobileSdk.Light">
    <item name="sdkSubtitleColor">@color/mySubtitleColor</item>
</style>
```

2.6. colorPrimary, colorAccent

colorPrimary - the back button on Toolbar has this color. The color of this button is changed by changing of colorPrimary

colorAccent - you can change cursor and selection color on EditText

```
<style name="AppTheme" parent="Theme.MobileSdk.Light">
    <item name="colorPrimary">@color/colorPrimary</item>
    <item name="colorAccent">@color/colorAccent</item>
</style>
```

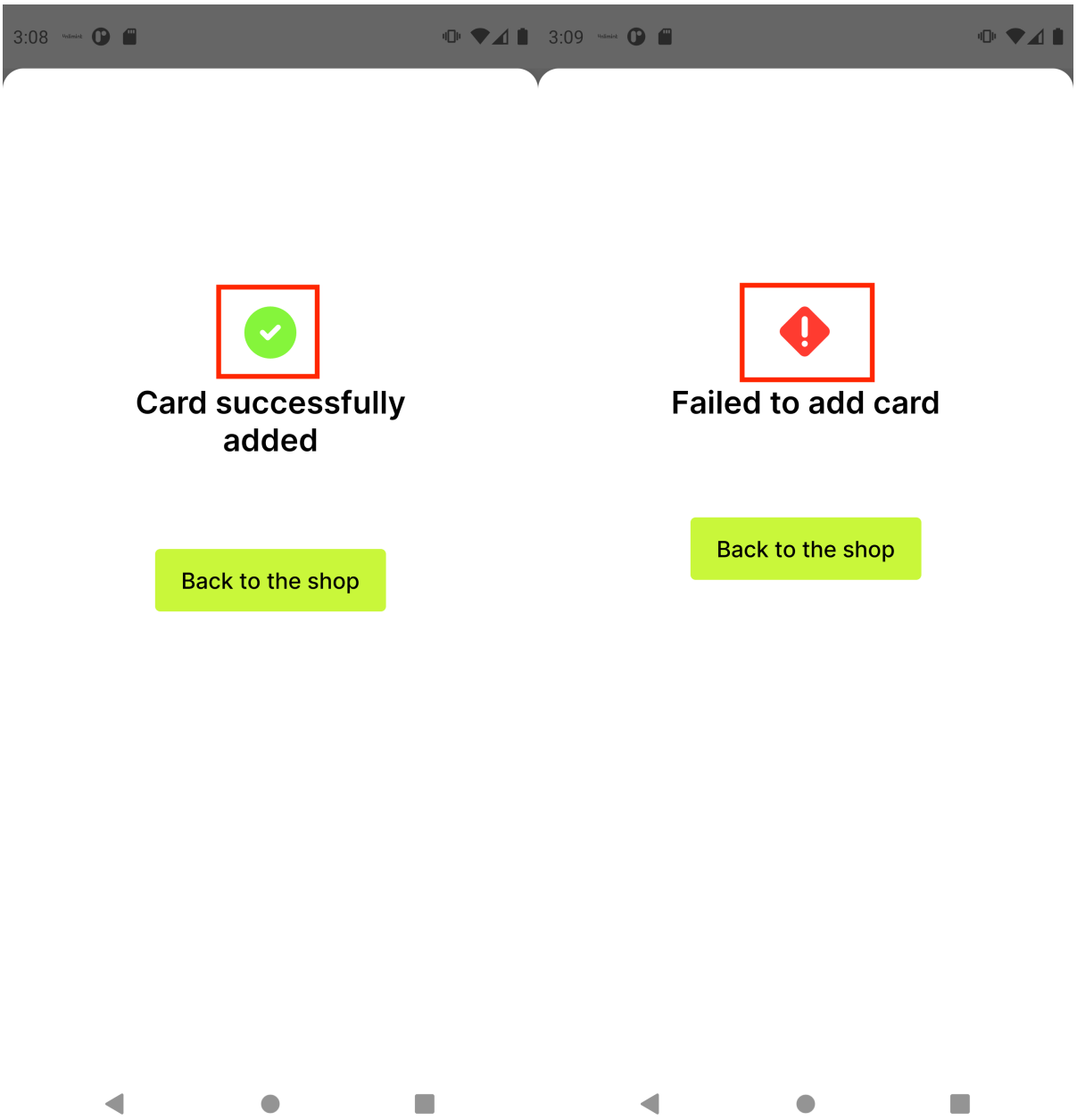
3. Custom images

You can change images for success/decline result of binding or payment

3.1. Binding

Use this code to change images for binding:

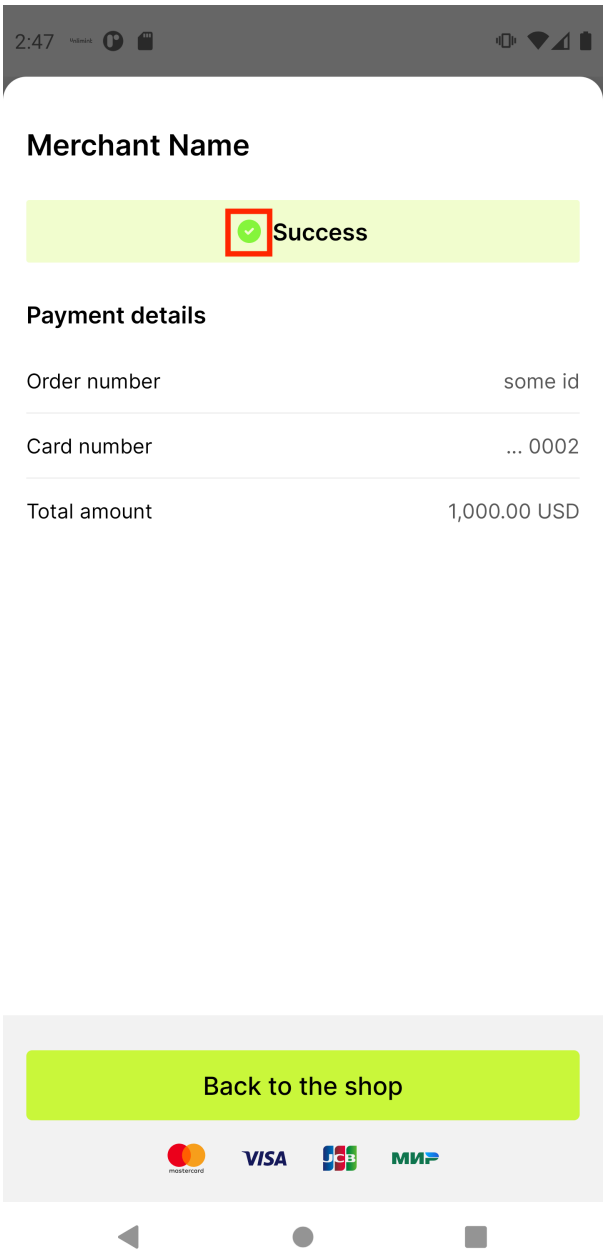
```
<style name="AppTheme" parent="Theme.MobileSdk.Light">
    <item name="sdkBindingSuccessDrawable">@drawable/ic_success</item>
    <item name="sdkBindingFailDrawable">@drawable/ic_fail</item>
</style>
```



3.2. Payment

For changing images on payment - set image on custom style for **sdkPaymentMessageSuccessStyle** and **sdkPaymentMessageDeclineStyle**

```
<style name="MyStyleForSuccessPayment">
    <item name="android:textColor">...</item>
    <item name="android:fontFamily">...</item>
    <item name="android:textSize">...</item>
    <item name="android:textStyle">...</item>
    <item name="android:drawableStart">@drawable/ic_payment_success</item>
</style>


<style name="AppTheme" parent="Theme.MobileSdk.Light">
    <item name="sdkPaymentMessageSuccessStyle">@style/MyStyleForSuccessPayment</item>
</style>
```


2:47
Merchant Name
Success
Payment details
Order number
some id
Card number
... 0002
Total amount
1,000.00 USD
Back to the shop