

Sample App Guide for Testing Java SDK (Sample App Framework: Spring Boot)

Within this document, the step-by-step process elucidates running the Sample App to evaluate the functionality of the Java SDK

By Pinelabs Team

Prerequisites

Open JDK Version ≥ 17 (Recommended)

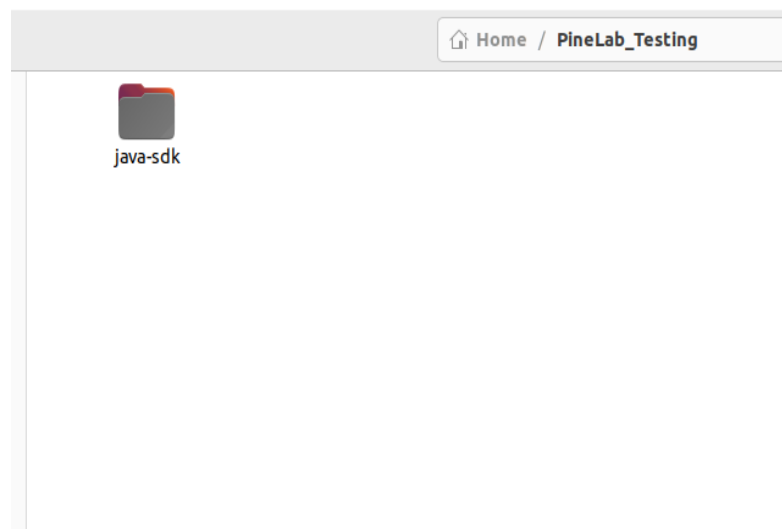
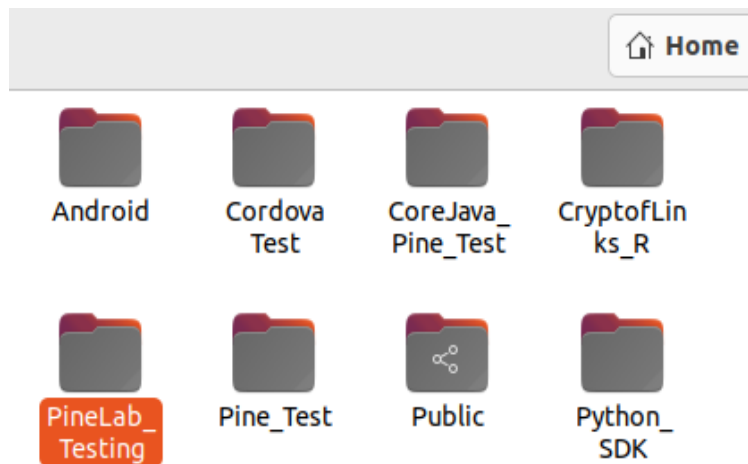
Maven Version ≥ 3 (Recommended)

Installation Process

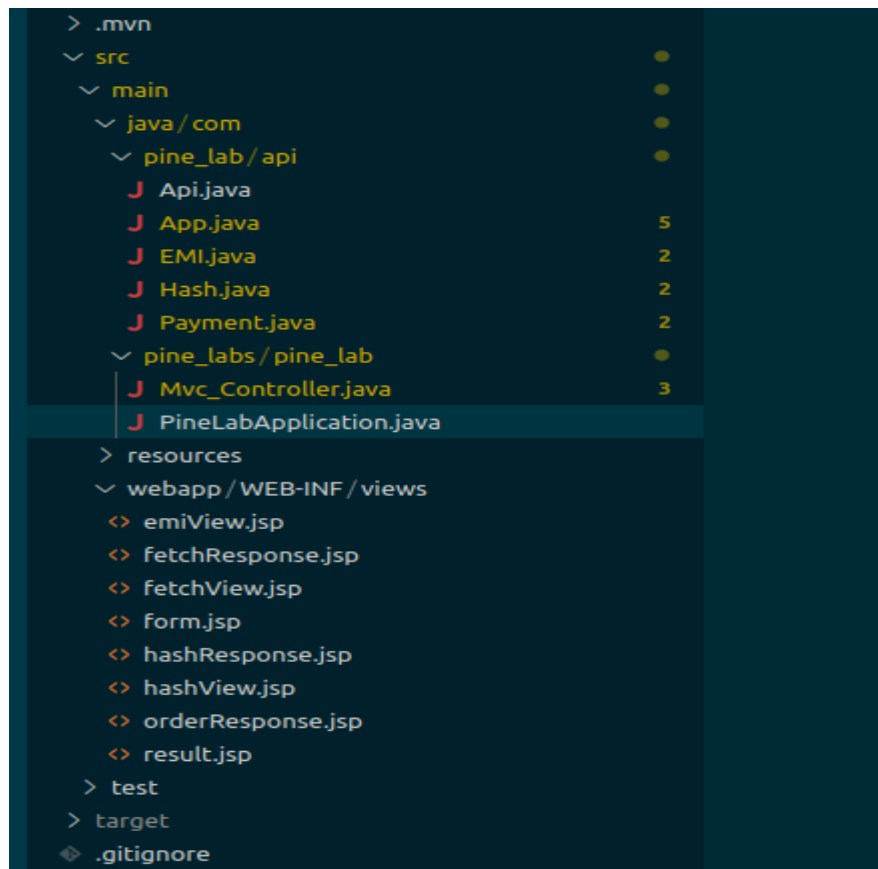
This section explains the installation flow of the Sample App

Below are the steps to extract the Sample App onto your local System and Enabling it for browser.

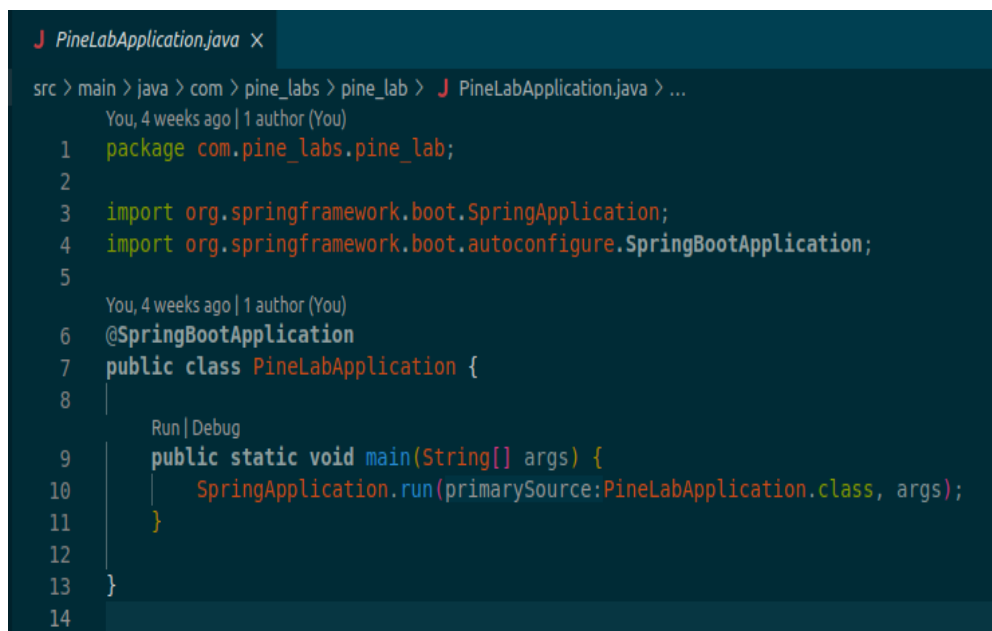
1. Extract the Sample App (Zip File) in any location on your system. Here we have extracted it at home path inside Java_Pine_Labs Folder (As shown below in image)



2. So it'll look something like this in your IDE, it'll consist of Sdk in pine_lab/api path and Controller classes will be located inside pine_labs /pine_lab path.



3. Now run the main method inside PineLabApplication.java class. After that navigate to url (<http://localhost:8081/form>) to test all functions provided by Sample App. (You can change port number 8081 according to your requirements from Application Properties file)



Usage Process

This section explains the usage flow of Sample App

Create Order Process

In order for us to create an order and generate a URL for payment we need to locate to url <http://localhost:8081/form> (Below shows the image of how Create order page will look like on browser)

Test Form

Create Order

[Create Order](#) | [Fetch Order Details](#) | [EMI Calculation](#) | [Hash Verification](#)

Merchant ID	Access Code	
<input type="text" value="106598"/>	<input type="text" value="4a39a6d4-46b7-474d-929d-21bf0e9ed607"/>	
Secret	Gateway Mode	
<input type="text" value="55E0F73224EC458A8EC0B68F7B47ACAE"/>	<input type="text" value="Sandbox"/>	
Transaction ID	Amount (In Paisa)	Callback URL
<input type="text" value="Transaction ID"/>	<input type="text" value="4000000"/>	<input type="text" value="http://localhost:8081/orderResponse"/>
Product Details		
<input type="text" value='[{"product_code":"testSKU1","product_amount":"2000000"}, {"product_code":"testSKU2","product_amount":"2000000"}]'/>		
Payment Mode		
<input type="button" value="Select Payment Mode"/>		
▼ Customer Details		
▼ Billing Address		
▼ Shipping Address		
▼ Additional Fields		
<input type="button" value="Pay Now"/>		

STEPS TO FOLLOW

1. Merchant Id, Access Code, Secret Key, Gateway Mode (Testing/Prod), Transaction Id, Amount in Paisa(Total amount of Products), Callback Url & Payment mode are all required fields and some of them have been pre filled with Test keys which you can change according to your requirements.
2. Product Details is Conditional (Meaning, Required in case of EMI payment mode)
3. Customer Details, Billing Address, Shipping Address and Additional Fields are Optional fields so aren't necessary for this process.
4. You need to keep Transaction Id unique every time while creating order else due to duplicate Id error you won't be redirected to anywhere.
5. If you add another product or subtract from existing products in Product Details then you need to make sure sum of total products amount is to be equal to Amount in Paisa as well if not change amount in Paisa according to total of Products in Product list, for test we have added 4000000 in amount in Paisa which equals to total amount of products in Product details you can change according to your needs.

6. After clicking on Pay Now you'll be redirected to Pinelabs Payment link page and you can select any payment mode like for testing Netbanking can be selected and after clicking on pay now you can generate success or Failure response.

The screenshot displays the Plural by Pine Labs payment interface. At the top right, there is a 'Cancel' button with a close icon. The order total is shown as ₹40000. On the left, a sidebar titled 'Choose a payment option' lists 'Credit Debit Cards', 'Net Banking' (highlighted in green), 'UPI/BHIM', and 'Wallets'. The main area is titled 'Select Bank' and shows 'IDBI Bank' as the selected option. A 'Pay Now' button is visible. The footer includes 'Safe & Secure Payments' and various payment logos like VISA, MasterCard, and SafeKey.

7. After that you'll be redirected to url provided in Callback Url field in HTML form which is part of this sample App and you'll get response something like shown in below image.

```
localhost:8081/orderResponse

{
  "merchant_id": "106598",
  "merchant_access_code": "4a39a6d4-46b7-474d-929d-21bf0e9ed607",
  "unique_merchant_txn_id": "650acb67d3752hvhv",
  "pine_pg_txn_status": "4",
  "txn_completion_date_time": "30/11/2023 11:57:17 AM",
  "amount_in_paisa": "4000000",
  "txn_response_code": "1",
  "txn_response_msg": "SUCCESS",
  "acquirer_name": "BILLDESK",
  "pine_pg_transaction_id": "14225606",
  "captured_amount_in_paisa": "4000000",
  "refund_amount_in_paisa": "0",
  "payment_mode": "3",
  "udf_field_1": "",
  "udf_field_2": "",
  "udf_field_3": "",
  "udf_field_4": "",
  "Acquirer_Response_Code": "0300",
  "Acquirer_Response_Message": "DEFAULT",
  "parent_txn_status": "",
  "parent_txn_response_code": "",
  "parent_txn_response_message": "",
  "dia_secret": "864B2941592516C626042E07034E70A9BCCD38F14D89A3977697044145EDE553",
  "dia_secret_type": "SHA256"
}
```

Fetch Order Process

In order for us to fetch order details we need to click on link Fetch Order Details in create order page or we can navigate to it directly by using path (<http://localhost:8081/fetchView>)

Test Form

Fetch Order Details

[Create Order](#) | [Fetch Order Details](#) | [EMI Calculation](#) | [Hash Verification](#)

Merchant ID

106598

Access Code

4a39a6d4-46b7-474d-929d-21bf0e9ed607

Secret

55E0F73224EC458A8EC0B68F7B47ACAE

Transacrion ID

Transacrion ID

Gateway Mode

Sandbox

Fetch Details

STEPS TO FOLLOW

1. Merchant Id, Access Code, Secret Key, Gateway Mode (Testing/Prod), Transaction Id are required fields and some of them have been pre filled with Test keys that you can change according to your requirement.
2. You need enter Transaction Id of which you previously created Payment or if having txn id for testing purpose you can enter that as well.
3. Now you need to click on fetch order and it'll give you response something like shown in below image

```
localhost:8081/fetch
{
  "ppc_MerchantID": "106598",
  "ppc_MerchantAccessCode": "4a39a6d4-46b7-474d-929d-21bf0e9ed607",
  "ppc_PinePGTxnStatus": "7",
  "ppc_TransactionCompletionDateTime": "21/09/2023 05:44:07 PM",
  "ppc_UniqueMerchantTxnID": "650acb67d3752",
  "ppc_Amount": "10000",
  "ppc_TxnResponseCode": "1",
  "ppc_TxnResponseMessage": "SUCCESS",
  "ppc_PinePGTransactionID": "12071550",
  "ppc_CapturedAmount": "10000",
  "ppc_RefundedAmount": "0",
  "ppc_PaymentMode": "3",
  "ppc_AcquirerName": "BILLDESK",
  "ppc_DIA_SECRET": "BA36309D5573B7C4DD68E85D8A7FBC1E68C5CD7B3789043C8AF9D6EFB6FF718C",
  "ppc_DIA_SECRET_TYPE": "SHA256",
  "ppc_Parent_TxnStatus": "4",
  "ppc_ParentTxnResponseCode": "1",
  "ppc_ParentTxnResponseMessage": "SUCCESS",
  "ppc_CustomerMobile": "9876543210",
  "ppc_UdfField1": "",
  "ppc_UdfField2": "",
  "ppc_UdfField3": "",
  "ppc_UdfField4": "",
  "ppc_AcquirerResponseCode": "0300",
  "ppc_AcquirerResponseMessage": "DEFAULT"
}
```

EMI Details Process

In order for us to calculate Emi we need to click on Emi Calculation or navigate to this path (<http://localhost:8081/emiView>).

Test Form

EMI Calculation

[Create Order](#) | [Fetch Order Details](#) | [EMI Calculation](#) | [Hash Verification](#)

Merchant ID	Access Code
106598	4a39a6d4-46b7-474d-929d-21bf0e9ed607
Secret	Amount (In Paisa)
55E0F73224EC458A8EC0B68F7B47ACAE	4000000
Gateway Mode	
Sandbox	
Product Details	
[{"product_code":"testSKU1","product_amount":"2000000"}, {"product_code":"testSKU2","product_amount":"2000000"}]	

Calculate EMI

STEPS TO FOLLOW

1. Merchant Id, Access Code, Secret Key, Gateway Mode (Testing/Prod), Amount in Paisa, Product Details are required fields and they have been pre filled with Test keys you can change according to your requirement.
2. If you add another product or subtract from existing products in Product Details then you need to make sure sum of total products amount is to be equal to Amount in Paisa as well if not change amount in Paisa according to total of Products in Product list, for test we have added 4000000 in amount in Paisa which equals to total amount of products in Product details you can change according to your needs.
3. After clicking on Calculate EMI button you'll be able to see response as shown in below image.

```
localhost:8081/emiCalculator
{
  "issuer": {
    "list_emi_tenure": [
      {
        "offer_scheme": {
          "product_details": [
            {
              "schemes": [
                {
                  "scheme_id": 48150,
                  "program_type": 112,
                  "is_scheme_valid": true
                }
              ],
              "product_code": "testSKU1",
              "product_amount": 2000000,
              "subvention_cashback_discount": 48733,
              "product_discount": 10000,
              "subvention_cashback_discount_percentage": 24500,
              "product_discount_percentage": 0,
              "subvention_type": 1,
              "additional_cashback": "5% upto 300",
              "bank_interest_rate_percentage": 150000,
              "bank_interest_rate": 48733
            },
            {
              "schemes": [
                {
                  "scheme_id": 48150,
                  "program_type": 112,
                  "is_scheme_valid": true
                }
              ],
              "product_code": "testSKU2",
              "product_amount": 2000000,
              "subvention_cashback_discount": 48733,
              "product_discount": 10000,
              "subvention_cashback_discount_percentage": 24500,
              "product_discount_percentage": 0,
              "subvention_type": 1,
              "additional_cashback": "5% upto 300",
              "bank_interest_rate_percentage": 150000,
              "bank_interest_rate": 48733
            }
          ],
          "emi_scheme": {
            "scheme_id": 48151,

```

Hash Verification Process

In order for us to verify hash we need to click on Hash Verification link or navigate to this path (<http://localhost:8081/hashVerificationView>)

STEPS TO FOLLOW

Test Form
Hash Verification
[Create Order](#) | [Fetch Order Details](#) | [EMI Calculation](#) | [Hash Verification](#)

Merchant ID: 106598
Access Code: 4a39a6d4-46b7-474d-929d-21bf0e9ed607
Secret: 55E0F73224EC458A8EC0B68F7B47ACAE
Gateway Mode: Sandbox

Response Data
Enter Response Data

Verify Hash

1. Merchant Id, Access Code, Secret Key, Gateway Mode (Testing/Prod), Response Data are required fields and some of them have been pre filled with Test keys you can change according to your requirement.
2. You need to paste response data that you got while creating order or while fetching details of order in Response Data as shown below.

Test Form
Hash Verification
[Create Order](#) | [Fetch Order Details](#) | [EMI Calculation](#) | [Hash Verification](#)

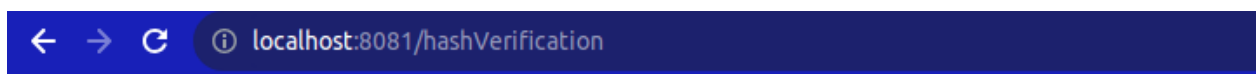
Merchant ID: 106598
Access Code: 4a39a6d4-46b7-474d-929d-21bf0e9ed607
Secret: 55E0F73224EC458A8EC0B68F7B47ACAE
Gateway Mode: Sandbox

Response Data

```
{
  "ppcs_MerchantID": "106598",
  "ppcs_MerchantAccessCode": "4a39a6d4-46b7-474d-929d-21bf0e9ed607",
  "ppcs_PinePOTransactionStatus": "7",
  "ppcs_TransactionCompletionDateTime": "21/09/2023 05:44:07 PM",
  "ppcs_UniqueMerchantTxnID": "650acbe7d3752",
  "ppcs_Amount": "10000",
  "ppcs_TxnResponseCode": "1",
  "ppcs_TxnResponseMessage": "SUCCESS",
  "ppcs_PinePOTransactionID": "12071550",
  "ppcs_PinePOTransactionStatus": "1"
}
```

Verify Hash

3. Now after clicking on Verify Hash button you'll be get Response like True or False whether hash has been verified or not.



Verification Status: **True**

TDR/GST Information:

Please note no additional charges like TDR, GST, etc are handled in our Plugins and the same needs to be manually handled at the merchant end.

TLS 1.2 information:

Java 8 and above provide TLS 1.2 support.

Note: The exact version might depend on the specific Java vendor (Oracle, OpenJDK, etc.).