

MonCash REST API WITH Java



Documentation

INTRODUCTION	2
USING WITH MAVEN PROJECT	2
USING AS JAR LIB	2
MAKING FIRST CALL	3



Introduction

Moncash REST API is a simple way of integrating the payment gateway into your site to take payments. In this article we're going to look at a simple example that will use Moncash SDK to take payments using Java.

Using with Maven Project

This SDK can be use in the maven project. Alternatively, if you prefer not to use Maven, but want to load the Jar libraries, you can get it by doing a direct download from <https://github.com/ecelestin/ecelestin-Moncash-sdk-java>.

Currently, Moncash Java Rest API SDK is available at github <https://github.com/ecelestin/ecelestin-Moncash-sdk-java.git> . To use it in your project, you need to include the maven or Jar file as a dependency in your project in pom.xml or your lib folder respectively.

```
<repositories>
  <repository>
    <id>Moncash-sdk-java</id>
    <url>https://github.com/ecelestin/Moncash-sdk-java</url>
    <snapshots>
      <enabled>true</enabled>
      <updatePolicy>always</updatePolicy>
    </snapshots>
  </repository>
</repositories>
<dependencies>
  <dependency>
    ...

    <groupId>com.digicelgroup.moncash</groupId>
    <artifactId>sdk-moncash-api</artifactId>
    <version>1.0-SNAPSHOT</version>
  </dependency>
</dependencies>
```

Using as Jar lib

If you don't want to use maven, you can grab the SDK zip that contains Moncash Java Rest API SDK and get the Jar file name `sdk-moncash-api-{version}-SNAPSHOT.jar`.

Making First Call

After the installation, so that we can import all using classes into your code. For our example code, we will include it at the top of the file that will be required by our request and response code.



```
import com.digicelgroup.moncash.APIContext;
import com.digicelgroup.moncash.http.Constants;
import com.digicelgroup.moncash.payments.Payment;
import com.digicelgroup.moncash.payments.PaymentCapture;
import com.digicelgroup.moncash.payments.PaymentCreator;
import com.digicelgroup.moncash.payments.TransactionId;
import org.apache.http.HttpStatus;
```

1. Initiating of the Api context

```
APIContext apiContext = new APIContext("client id", "client secret"
Constants.SANDBOX);
```

2. Creating of payment

```
PaymentCreator paymentCreator = new PaymentCreator();
Payment payment = new Payment();
payment.setOrderId(System.currentTimeMillis()+"");
payment.setAmount(50);
PaymentCreator creator = paymentCreator.execute(apiContext,PaymentCreator.class, payment);
if(creator.getStatus() !=null &&
creator.getStatus().compareTo(HttpStatus.SC_ACCEPTED+"")==0) {
    logger.info("redirect to the link below");
    //creator.redirectUri() method return the payment gateway url
    logger.info(creator.redirectUri());
}else if(creator.getStatus()==null){
    logger.warning("Error");
    logger.warning(creator.getError());
    logger.warning(creator.getError_description());
}else{
    logger.warning("Error");
    logger.warning(creator.getStatus());
    logger.warning(creator.getError());
    logger.warning(creator.getMessage());
    logger.warning(creator.getPath());
}
```

3. Retrieving a payment by the transactionId

```
PaymentCapture paymentCapture = new PaymentCapture();
TransactionId transactionId = new TransactionId();
transactionId.setTransactionId("12874819");
PaymentCapture capture = paymentCapture.execute(apiContext, PaymentCapture.class,
transactionId);

if(capture.getStatus() !=null && capture.getStatus().compareTo(HttpStatus.SC_OK+"")==0) {
    logger.info("Transaction");
    logger.info(capture.getPayment().getMessage());
    logger.info("transactio_id="+capture.getPayment().getTransaction_id());
    logger.info("Payer="+capture.getPayment().getPayer());
    logger.info("amount="+capture.getPayment().getCost()+"");
} else {
    logger.info(capture.getStatus());
}
}
```

4. Retrieving a payment by the orderId

```
PaymentCapture paymentCapture = new PaymentCapture();
OrderId orderId = new OrderId();
orderId.setOrderId("12874819");
PaymentCapture capture = paymentCapture.execute(apiContext, PaymentCapture.class, orderId);

if(capture.getStatus() !=null && capture.getStatus().compareTo(HttpStatus.SC_OK+"")==0) {
    logger.info("Transaction");
    logger.info(capture.getPayment().getMessage());
    logger.info("transactio_id="+capture.getPayment().getTransaction_id());
    logger.info("Payer="+capture.getPayment().getPayer());
    logger.info("amount="+capture.getPayment().getCost()+"");
} else {
    logger.info(capture.getStatus());
}
}
```