CBEBirr PAYMENT GATEWAY FINAL INTEGRATION DOCUMENT

Step 1

prepare all the necessary data to be sent.The data to be sent are:

- UserID
- Password
- TransactionId
- Amount
- Merchant Code

Step 2:-.

2.1 Prepare Hashing payload as Follows

```
payload={
        "U":"UserID",
        "W": "Password",
        "T":"TransactionId",
        "A": "Amount",
        "MC": "MerchantCode",
        "Key":"Key"
        }
```

2.2 Now prepare a Method that has the following functionalities

#1 the method should Sort the inputted payload with the key as follows

```
newPayload={
        "A": "Amount",
        "Key":"Key",
        "MC": "MerchantCode",
        "T":"TransactionId",
        "W": "Password",
        "U":"UserID"
        }
```

#2 the method should Convert the above sorted map to the following String and return that string

ProcessedPayload=”A=Amount&Key=Key&MC=MerchantCode&T=TransactionId&U=UserID&W=Password”

To do this you can use the following code snippet

```
List<String> temp = new ArrayList<>();
    for (Map.Entry<String, String> entry : payload.entrySet()) {
        temp.add(String.format("%s=%s", entry.getKey(), entry.getValue()));
    }
    return String.join("&", temp);
```

2.3 Now Hash the Above processedPayload With the Hashing algorithm we provided

Step 3

Encrypt each individual data to be sent using the key and encryption algorithm we provided.

- ★ UserID
- ★ Password
- ★ TransactionId
- ★ Amount
- ★ Merchant Code
- ★ Hash value.

Step 4

After encrypting each data individually format it in the following json format

```
 {
"U":"EncryptedUserID",
"W": "EncryptedPassword",
"T":"EncryptedTransactionId",
"A": "EncryptedAmount",
"MC": "EncryptedMerchantCode",
"HV":"EncryptedHashValue"
}
```

Step 5

Encrypt the above whole json using the encryption algorithm and key we provided.

Step 6

After preparing the encrypted data send it as a query parameter as indicated below.
https://cbebirrpaymentgateway.cbe.com.et:8888/Default.aspx?r=encrypteddata

Step 7

Prepare an endpoint where we will announce you the transaction status.
7.1 we will be Sending Encrypted body as follows so accept likewise

```
{"EncVal",EncryptedResponseValue}
```

7.2 decrypt the EncryptedResponseValue and deserialize it you will get the following json

```
{
       "TransactionId":"transactionid" ,
        "State" :"transactionstatus",
        "TNDDate" : "transactiondate",
        "Signiture" : "hashvalue"
}
```

7.3 Prepare Hashing payload as Follows

```
payload={
        "TransactionId":"TransactionIdValue",
        "State": "Statevalue",
        "TNDDate":"TNDDateValue",
        “Key”:”Private Hashing Key”
        }
```

7.4 pass the payload to the method you created in step 2 and hash the response from this method
7.5 check whether the hashed value from 7.4 is equal to the signature from 7.2
If they are not equal the data is altered in some way,

Step 8

You can send a transactionId at any time to see its status along with the following Api.
https://cbebirrpaymentgateway.cbe.com.et:8888/api/cbebpg/TXNSTAT

8.1 Prepare Hashing payload as Follows

```
payload={
```

```
                "transactionId":"transactionId",
                "Tillcode":"tillcode",
                “Key”:”Private Hashing Key”
                }
```

8.2 pass the payload to the method you created in step 2 and hash the response from this method

Send your requestBody in the following format

```
{
 "transactionId":"transactionId",
 "Tillcode":"tillcode",
"Signiture":"hashvalue"
}
```

8.3 we will be Sending Encrypted response body as follows so accept like wise

EncryptedResponseValue

8.4 decrypt the EncryptedResponseValue and deserialize it you will get the following json

```
        {
        "Tillcode":"ShortCode",
        "TransactionId":"TransactionId",
        "State":"State",
        "TNDDate":"TNDDate",
        "Signiture":"6132f47a9d53914ca2ee607b051e3f294362d710f904794216e
        ee40c57b22b26"}
```

8.5 Prepare Hashing payload as Follows

```
        payload={
                "Tillcode":"ShortCode",
                "TransactionId":"TransactionId",
                "State":"State",
                "TNDDate":"TNDDate",
                “Key”:”Private Hashing Key”
                }
```

8.6 pass the payload to the method you created in step 2 and hash the response from this method

8.7 check whether the hashed value from 7.4 is equal to the signature from 7.2

If they are not equal the data is altered in some way,

Thank you!