Globill Hosted Payment Page (HPP) Document

Table of Contents

1 Introduction

This document is for use when integrating with the Globill Hosted Payment Page (HPP) payment solution. It is intended to assist developers to integrate their applications with the Globill payment solution platform. Prior to use you should have submitted an application and supporting documents to receive credentials to the platform: a MID and password. You should also have a separate login for the backoffice where you can see summary statistics, view individual transactions and make refunds. The backoffice is available at: https://secure.globill.net/bo

2 Globill Back-office

Our Back-office provides our merchants with a user interface to configure and manage their payments. There are 7 menu options:

1. Setup
2. Checkout Demos
3. Statistics
4. Summary Reports
5. Transactions
6. Disputes & Chargebacks
7. Payout Report

2.1 *Setup*

Here the merchant can set up their account for accepting CNP payments. There are 4 sections:

1. Merchant Contact Details
2. Merchant Banking Details
3. Email Acknowledgement after customer Payment
4. Website Details

Each section is elaborated on below.

2.1.1 Merchant Contact Details

Please submit the contact details for Admin and for Technical personnel within your company. Details asked for are Name, Email, Phone and optionally Skype ID

2.1.2 Merchant Banking Details

Please add your company's banking details. You may add more than one beneficiary banking details in which case you must select your chosen bank account. Details required are: Bank name, address and SWIFT and Account holder's name, address, account number or IBAN or ABA Routing number if available (US) and account currency.

2.1.3 Email Acknowledgement after customer Payment

This is simply a configuration setting to tell us whether you'd like us (our platform) to send an acknowledgement email to your customer and/or to yourself as a merchant.

2.1.4 Website Details

There are two parts to this section: Pricing Currency and your Webshop settings required for payments.

Pricing Currency is the currency of your web-shop's prices. By default this is USD but you can choose from 31 of the world's most popular currencies (AUD, BRL, CAD, CHF, CLP, CNY, DKK, EUR, GBP, HKD, HUF, ILS, INR, JPY, KRW, MXN, MYR, NOK, NZD, PHP, PKR, PLN, RUB, SEK, SGD, THB, TRY, TWD, USD and ZAR).

Even though you setup only one single Pricing Curency, you can still accept payments in any of the 31 currencies we support. See currencyList field below in the Sale Transaction.

The second part of this section concerns your Web-shop settings for accepting payments. It is imperative to set these fields in order to use our HPP effectively.

Here follow a description of these fields:

WebsiteID: internally assigned to you
Website URL: your web-shop URL
Customer Support Contact: country, phone and email contact details for your shop's customer support. This must be filled in because it shows up on the payment page.
Solution(s): Visa/MasterCard and/or AMEX to accept
Payment Flow: **ResultURL** is optional. It points to a page on your server where you may want to act on the payment result. Typically, you would update the user's record in your database with the payment result. **Redirect** checkbox when ticked on will redirect the user to your **ResultURL** page.

2.2 *Integration Help*

On this page, you will be presented with 7 payment checkout demos to test all aspects of the HPP. From the simplest demo where only 3 form fields are necessary to post to our HPP, a test showing the simplicity of our multi-currency offering, a test showing how you sell products with shipping another test to show how you can sell services. Other tests show how you can prefill the customer's details with data captured on your pages. The final test shows how you can use our HPP where you host your own payment page, and our HPP is 'passed-thru'- effectively hidden away.

2.3 *Statistics*

Displays the trends in your turnover over a time period. This is available for daily, weekly and monthly time-frames.

2.4 *Summary Reports*

This is a report showing the breakdown of your volumes either as a Daily report, a Weekly report or a Monthly report. Volumes refer to Sales, Refunds, Disputes and Charge-backs. Volume is presented in terms of monetary amount as well as the count of the transactions. The report output is either screen or as a downloadable spreadsheet file.

2.5 *Transactions*

Lists all transactions processed under your merchant account. You can limit the data presented in the report based on: (I) Transaction type: Sales or Refunds, (ii) Transaction status: Authorized or Not Authorized and (iii) Filter criteria. Filter the data by either of: time period, by Order ID, Transaction ID, amount, or any cardholder detail.

2.6 *Disputes & Chargebacks*

Lists all sale transaction that have been disputed by the customer. The customer can dispute for any number of reasons such as: Merchandise not received, Merchandise not as described, Credit not processed, Fraud, Incorrect transaction amount. Further notes about how best to handle each type of dispute is presented on the lower half of this page in the Backoffice.

A dispute that is unchallenged or unsuccessfully challenged by the merchant turns into either a chargeback or a refund.

2.7 *Payout Report*

This report provides a breakdown of settlement payouts on a per week basis. Payout Due is calculated as follows:
Payout Due = Sales less (Refunds + Chargebacks + Fees + Holdback)

3 Use Overview

Our HPP provides a simple and secure method for you to accept credit and debit card payments from your web or mobile application via HTTPS. To integrate our payment services your application service (be it a web page or a server script or mobile app etc.) needs to be able to post form fields to our secure HPP page. You do need to use a SSL certificate on your serve so that you can communicate with our service over HTTPS.

4 Payment Process

Step 1: The customer selects goods or services from your web shop and optionally adds them to a shopping cart. (a shopping cart is not necessary if you sell individual goods or services)

Step 2: The customer clicks on the “Pay now” button or link or similar (on your page)

Step 3: The merchant and the customer details (that are known or already filled at this point from your side) are posted to our HPP and the customer is directed to the HPP (except for the case of “passthru” set to True – more on this later)

Step 4: The customer enters their Billing (contact details and address) and Payment information (credit card number, expiry date and CVV).

Step 5: Our HPP redirects the customer's browser back to the merchant's web shop if the payment is successful. If the payment fails then the customer will be redirected back to the HPP to re-enter credit card details to try the payment again.

Note that the Payment Process described above can be customised to a certain extend by way of setting form field parameters and/or by changing certain Setup settings within the Globill.net Backoffice (more on this later).

4.1 *Data Types and Values*

The entries may either be alphanumeric, numeric or single character.
The field names are case insensitive eg cardholderFirstname and CardHolderFirstName can be used interchangeably.

· The abbreviation for alphanumeric inputs is "AN" (0-9 A-Z a-z .!@).
· The abbreviation for numeric inputs is "N" (0-9).
· The abbreviation for single letters is "CHAR".
· The abbreviation for boolean is "B". Valid values are 1/0 or true/false or yes/no

4.2 *Sale*

A Sale transaction is basically a combined transaction; an Authorization transaction and a Capture transaction. A Sale not only checks that the credit card being used in a transaction contains sufficient funds to cover the amount of the transaction, it also flags the transaction as captured, which means it is to be sent for settlement in the next settlement period.

The Sale transaction is the most used transaction in relation to trading of online services, where the product or service is delivered to the customer online.

URL for the API	https://securepayform.com/hpp.cfm
Name of API Operation	SaleTransaction

4.2.1 Sale Request

Fields that you post to our HPP can be categorized as follows:

1. Merchant fields (all fields mandatory)

2. Order fields (all fields optional except amount)

3. Billing address (all fields optional)

4. Shipping address (all fields optional)

5. Payment information (all fields optional)

6. UI fields (all fields optional)

7. Processing flow fields (all fields optional)

Field	**Data Type & Value**	**Cate gory**	**Required**
MID	N(6-10). Issued by Globill to you	1	yes
websiteID	N(1-6). Obtained from Globill Backoffice: Setup – Websites	1	yes
orderID	AN(32) Must be unique. If not provided then Globill will auto generate unique orderID for you	2	no
amount	N(10) Number in minor unit, e.g. cents; 100 dollar cents equals 1 dollar. Amount *is* required if you are *not* providing individual item pricing. Amount is the total order amount in the currency you set up as your Pricing Currency (Backoffice Setup)	2	no
currencyList	A(255) List of currency codes you want to sell in. Separate each currency with | as in the following example: USD|EUR|GBP|NOK. Default processing currency is the USD. This parameter is only necessary if you want to provide your customers with a choice of currencies to pay with.	2	no
orderServiceDesc [0..n]	AN(255) Optional string describing a single service item making up the order. If the customer's order is made up of more than one service then simply use as many orderServiceDesc fields as you have services.	2	no
orderProductDesc [0..n]	AN(255) Optional string describing the product item making up the order. If the customer's order is made up of more than one product then simply use as many orderProductDesc fields as you have services.	2	no
orderProductQty [0..n]	N(1-4) Optional number of product items of the item	2	no

	described by orderProductDesc		
orderProductPrice [0..n]	N(10) Number in minor unit, e.g. cents; 100 dollar cent equals 1 dollar of the price of the item described by orderProductDesc	2	no
shippingMethod	AN(255) Optional string describing the shipping method applicable to the order. You should include any pricing information within shippingMethod	2	no
shippingPrice	N(10) Number in minor unit, e.g. cents; 100 dollar cent equals 1 dollar of the price of the shipping method	2	no
cardholderFirstname	AN(35)	3	no
cardholderLastname	AN(35)	3	no
cardholderPhone	AN(20)	3	no
cardholderEmail	AN(50)	3	no
cardholderAddress	AN(255)	3	no
cardholderCity	AN(50)	3	no
cardholderPostcode	AN(10)	3	no
cardholderRegion	AN(30)	3	no
cardholderCountryCode	AN(2) See Country Codes (ISO 3166-1) for a complete list.	3	no
shippingAddress	AN(255)	4	no
shippingCity	AN(50)	4	no
shippingPostcode	AN(10)	4	no
shippingRegion	AN(30)	4	no
shippingCountryCode	AN(2) See Country Codes (ISO 3166-1) for a complete list.	4	no
cardNumber	N(16)	5	no
cardExpireMonth	N(2)	5	no
cardExpireYear	N(4)	5	no
cardSecurityCode	N(3)	5	no
hideOrderID	B '1' to hide the orderID on the HPP	6	no
hideBilling	B '1' to hide the customer's billing details on the HPP	6	no
hideShipping	B '1' to hide the customer's shipping details on the HPP	6	no
passthru	B '1' for passthru '0' for normal behaviour. Passthru bypasses the Globill Hosted Payment Page. To effectively use passthru you need to securely capture the customer's payment details on your page before calling the HPP. You need SSL installed on your payment page to use this.	7	no
resultURL	AN(255) Optional URL to receive IPN (Instant Payment Notifications) as defined by Sales Response (see below)	7	no
redirect	B '1' to redirect the customer's browser to a URL on	7	no

	your server according to the value of resultURL.		

Sample code:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Globill.net Payment checkout Demo3</title>
</head>

<body>
<?php

$signkey="h6Dja65";     //optional - only necessary if you intend using checksum

// CATEGORY 1 FIELDS (mandatory)
$MID="400101";          //mandatory field
$websiteID="14";        //mandatory field

// CATEGORY 2 FIELDS (optional, except orderAmount)
$randomnum=rand(10000,20000);   //random number generator to create orderID
$orderID=$randomnum;            // the orderID must be unique for sale
$orderAmount="123";             //$1.23

//CATEGORY 3 FIELDS (optional)
$cardholderFirstname="Jack";
$cardholderLastname="Rudd";
$cardholderPhone="+1 234 567 890";
$cardholderEmail="test@mail.com";

$cardholderAddress="14 Bay View";
$cardholderCity="Miami";
$cardholderPostcode="33101";
$cardholderRegion="Florida";
$cardholderCountryCode ="US";

//CATEGORY 4 FIELDS (optional)

//CATEGORY 5 FIELDS (optional)
$cardNumber="4000000000000002";
$cardExpireMonth="04";
$cardExpireYear="2017";
$cardSecurityCode="135";

$hideBillingAddress="0";
$hideShippingAddress="0";




$checksum = hash("md5" , $MID. $orderAmount. $cardholderFirstname.
$cardholderLastname.$signkey );
 ?>
<form name="input" action="https://securepayform.com/hpp.cfm" method="post">
   <input type="text" name="MID" value="<?php echo $MID?>" />
   <input type="text" name="websiteID" value="<?php echo $websiteID?>" />
   <input type="text" name="orderID" value="<?php echo $orderID?>" />
   <input type="text" name="orderAmount" value="<?php echo $orderAmount?>" />
   <input type="text" name="cardholderFirstname" value="<?php echo
$cardholderFirstname?>" />
   <input type="text" name="cardholderLastname" value="<?php echo
```

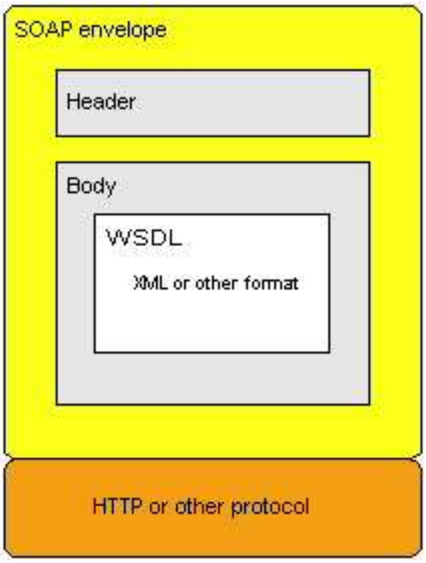
```
$cardholderLastname?>" />
   <input type="text" name="cardNumber" value="<?php echo $cardNumber?>" />
   <input type="text" name="cardExpireMonth" value="<?php echo
$cardExpireMonth?>" />
   <input type="text" name="cardExpireYear" value="<?php echo $cardExpireYear?>"
/>
   <input type="text" name="cardSecurityCode" value="<?php echo
$cardSecurityCode?>" />
   <input type="text" name="cardholderEmail" value="<?php echo
$cardholderEmail?>" />
   <input type="text" name="cardholderPhone" value="<?php echo
$cardholderPhone?>" />
   <input type="text" name="cardholderAddress" value="<?php echo
$cardholderAddress?>" />
   <input type="text" name="cardholderCity" value="<?php echo $cardholderCity?>"
/>
   <input type="text" name="cardholderPostcode" value="<?php echo
$cardholderPostcode?>" />
   <input type="text" name="cardholderRegion" value="<?php echo
$cardholderRegion?>" />
   <input type="text" name="cardholderCountryCode" value="<?php echo
$cardholderCountryCode?>" />
   <input type="text" name="hideBillingAddress" value="1" />
   <input type="text" name="hideShippingAddress" value="1" />
   <input type="text" name="passthru" value="1" />
   <input type="submit" value="pay" />
</form>
</body>
</html>
```

4.2.2 Sale Response

Field	Data Type & Value
transactionID	N(10-20)
orderID	AN(32)
status	“Authorized” / “Not Authorized”
errorCode	N(3) See Error Codes and Error Messages
errorMessage	AN(100) See Error Codes and Error Messages
statementDescriptor	AN(50) e.g. mywebshop

4.3 *Refund*

A refund is the process of refunding a previously settled transaction. A refund appears as a credit on the cardholder's credit card statement. Refund requests are sent as a SOAP-based Web Service. SOAP provides the envelope for sending Web Services messages over the Internet. It is part of the set of standards specified by the W3C. SOAP is an alternative to Representational State Transfer (REST) and JavaScript Object Notation (JSON).



The SOAP envelope contains two parts:

1. An optional header providing information on authentication, encoding of data, or how a recipient of a SOAP message should process the message.
2. The body that contains the message. These messages can be defined using the WSDL specification.

SOAP commonly uses HTTP and is used here to call a remote procedure (API operation).
NOTE: SOAP at one time stood for Simple Object Access Protocol. Starting with SOAP Version 1.2, the letters in the acronym have no particular meaning.

URL for the API	https://securepayform.com/servicesAPI/Transact.cfc?wsdl
Name of API Operation	RefundTransaction

4.3.1 Refund Request

Field	**Data Type & Value**	**Mandatory**
MID	N(6-10)	Mandatory
password	AN(64)	Mandatory
transactionID	AN(10-20)	Mandatory
amount	N(10) Number in minor unit, e.g. cents; 100 dollar cent equals 1 dollar.	Mandator

4.3.2 Refund Response

Field	**Data Type & Value**
transactionID	N(10-20)
status	“Authorized” / “Not Authorized”
errorCode	N(3) See Error Codes and Error Messages
errorMessage	AN(100) See Error Codes and Error Messages

Note: It is possible to do multiple refunds on a transaction, but the sum of the refunded amounts must be equal to or less than the amount of the original transaction.

4.3.3 Refund Request Example

Coming soon …

4.3.4 Refund Response Example

Coming soon…

5 Integration Procedure and Integration Test

To facilitate integration, a test gateway is available for test transactions with test card numbers. When you wish to switch from test to live mode, please notify us. The switch from test to live gateway does NOT require any change on merchant's part. All URLs and parameters will remain the same.
The following test cards will only be accepted while the merchant is on the test gateway.

5.1 *Test Cards*

Choose from any of the card details below to perform test payment checkouts.

Card	Cardnumber	Response Status	Response Message
Visa	4000000000000002	Authorized	Transaction successfully completed
Visa	4000000000000002	Not Authorized	Card expired*
Visa	4000000000000028	Authorized	Insufficient funds
Visa	any invalid Visa card number	Not Authorized	Invalid card number
Visa	4000000000000036	Not Authorized	Declined
MasterCard	5200000000000015	Authorized	Transaction successfully completed
MasterCard	5200000000000015	Not Authorized	Card expired*
MasterCard	5200000000000023	Not Authorized	Insufficient funds
MasterCard	any invalid MasterCard number	Not Authorized	Invalid card number
MasterCard	5200000000000049	Not Authorized	Declined
MasterCard	5200000000000064	Not Authorized	Bank not available

Use and card expiry dte in the future and any 3 digits for CVV.
* To simulate a card expired test, use any card expiry date in the past and any 3 digits for CVV.

6 Parameter Values

6.1 *Request Parameters*

If a field is mandatory, but no value is given by the customer, then the value "NA" should be submitted.

In the case of customerIP, a valid IP address must be submitted. In case no IP address is available then 127.0.0.1 can be submitted.

If Fraud Screening (FS) policy is used then the following fields become mandatory:
· cardHolderAddress

· cardHolderZipcode
· cardHolderCity
· cardHolderState
· cardHolderPhone

6.1.1 Fraud Screening (FS) Policy

Fraud Screening (FS) is a fraud detection system that stores extracts of the payment and other activity of cardholders in order to profile their activity, and also stores extracts of the overall global or segmented payment activities. This profile information is then used to effectively detect and minimize fraudulent activity.

FSPolicy	Description
	no fraud screening is performed.
SkipSkip Fraud Screening for selected transactions.	Skip Fraud Screening for selected transactions.
Ignore	Call Fraud Screening but ignore result (execute transaction).
Reject	Accept transaction in the case of CHALLENGE or ERROR, reject in all other circumstances.

6.1.2 Using Integrated MaxMind minFraud Fraud Protection Service

minFraud is the online fraud detection service that combines MaxMind's GeoIP® technology with other in-house developed order variable checks, such as open proxy detection. It calculates fraud score for a transaction based on transaction details.

Globill provides means to perform fraud checking against MaxMind's minFraud service. Whenever calculated fraud score exceeds the threshold set for a website, the transaction is rejected automatically with an error code of *640 - Transaction rejected by MaxMind minFraud fraud protection service.*

In order to use it three conditions must be fulfilled:

1. In the back office MaxMind check should be enabled on a website
2. The fraud score threshold (0.0 - 100.0) should be set for the website. Do note that a LOW minFraud riskScore means it is LESS likely that a transaction is fraud.
3. MaxMind related parameters should be passed along with transaction details when performing Sale or Authorize transaction request

If all the above conditions are met, fraud score is calculated for transactions being processed.

MaxMind related parameters should be put into userVar1 field of a Sale or Authorize request. Parameters are to be placed in an XML document with minFraud root tag. Each of the parameters shown below should be placed in a separate tag.

Field	Value	Mandatory/Optional
ip	ip	Mandatory
city	string	Mandatory
region	string	Mandatory
postal	string	Mandatory
country	string	Mandatory
domain	string	Optional
bin	string	Optional
binName	string	Optional

binPhone	string	Optional
custPhone	string	Optional
forwardedIP	ip	Optional
emailMD5	md5	Optional
usernameMD5	md5	Optional
passwordMD5	md5	Optional
shipAddr	string	Optional
shipCity	string	Optional
shipRegion	string	Optional
shipPostal	string	Optional
shipCOuntry	string	Optional
txnID	string	Optional
sessionID	string	Optional
user_agent	string	Optional
accept_language	string	Optional

Refer to minFraud integration reference for detailed explanation of these fields.

Example XML document containing minFraud parameters:

```
<minFraud xmlns="https://secure.globill.net/minFraud">
        <i>127.0.0.1</i>
        <city>London</city>
        <country>UK</country>
        <!-- Other fields go here -->
</minFraud>
```

Note 1: *userVar1* should be properly escaped based on the interface you use. For SOAP interface at least < and & symbols should be escaped.

Note 2: If *userVar1* if malformed or minFraud service fails to calculate fraud score for some reason the check will be skipped and transaction will proceed its normal flow.

Note 3: In order to simplify integration of this feature a schema file is provided as part of the examples to validate the minFraud parameters, also the PHP examples demonstrates use of the MaxMind minFraud service.

6.1.3 Error Codes and Error Messages

Errorcode	**Error Message**
000	Transaction successfully completed
900001	Call for approval
900002	Card expired
900003	Insufficient funds
900004	Invalid card number
900005	Bank interface timeout
900007	Declined

900009	Lost card
900011	Suspected fraud
900012	Card reported as stolen
900013	Restricted card
900014	Excessive card usage
900207	Declined; authentication failed
900020	Auth Declined
900029	Transaction not completed
991001	Invalid expiry date
991002	Invalid amount
900205	Unexpected authentication result (phase 1 3DS)
900206	Unexpected authentication result (phase 2 3DS)
990022	Bank not available
990053	Error processing transaction
900201	Phase 1 3DS authentication complete
900202	Phase 2 3DS authentication complete
900208	Not enrolled for authentication
900209	Transaction verification failed (phase 2)
900210	Authentication complete; transaction must be restarted
990024	Duplicate transaction detected. Please check before submitting

6.1.4 Response Status

Status	Description
Authorized	The transaction is approved by the acquiring bank
Not Authorized	The transaction is declined

7 Country Codes (ISO 3166-1)

ISO 3166-1 alpha-2 codes are two-letter country codes defined in ISO 3166-1, part of the ISO 3166 standard published by the International Organization for Standardization (ISO), to represent countries, dependent territories, and special areas of geographical interest.

Please refer to http://en.wikipedia.org/wiki/ISO_3166-1_alpha-2

8 Currency Codes (ISO 4217)

ISO 4217 is the international standard describing three-letter codes (also known as the currency code) to define the names of currencies established by the International Organization for Standardization (ISO).

Please refer to http://en.wikipedia.org/wiki/ISO_4217

9 Frequently Asked Questions (FAQ)

9.1 *3D Secure*

3D Secure adds another authentication step for online payments. Merchants are encouraged to use 3D Secure to achieve higher coverage against fraud losses. When a merchant does not use 3D Secure they are liable for fraudulent transactions even if the transaction was properly authorized. 3D Secure encompasses both Visa's Verified by Visa and MasterCard's SecureCode security solutions for online transactions. These solutions use personal passwords or like to help protect cardholders' card numbers against unauthorized use. Once activated a cardholder's card number cannot be used for online purchases without providing a personal password or like protection.

9.2 *Authorization (Auth)*

The process of checking that the credit card being used in a transaction contains sufficient funds to cover the amount of the transaction. Note that if sufficient funds are found, the amount is held for a given period of time, waiting to be withdrawn when settlement occurs (the period of time varies based on the issuing bank of the credit card).

9.3 *Authorization and Capture (Sale)*

An Authorization and Capture not only checks that the credit card being used in a transaction contains sufficient funds to cover the amount of the transaction, it also flags the transaction as captured meaning it is to be sent for settlement in the next settlement period.

9.4 *Card Identification Digits (CID)*

The 4-digit code found on the front of AMEX cards, the CID is used as an extra security step to verify that the person using the credit card is the actual cardholder.

9.5 *Card Verification Value (CVV/CVV2)*

The 3-digit code found on the back of Visa cards, the CVV2 is used as an extra security step to help to verify that the person using the credit card is the actual cardholder.

9.6 *Card Verification Code (CVC)*

The 3-digit code found on the back of MasterCard cards, the CVC2 is used as an extra security step to help to verify that the person using the credit card is the actual cardholder.

9.7 *Credit Fund Transfer (CFT)*

Credit Fund Transfer (CFT) enables merchants transfer money back into a given cardholder account, e.g.payouts from a casino account or like scenarios.

9.8 *Cancel*

A cancel is the process of reversing a previously authorized transaction, but not yet captured, transaction. It means that the cancel transaction will never appear on the cardholder's credit card statement. Usually authorizations is valid for a given period of time (the period of time varies based on the issuing bank of the credit card).

9.9 *Capture*

When a capture is performed (either in an Sale or Capture only transaction) it is the process of flagging an already authorized transaction to be settled in the next settlement period.

9.10 *Rebill*

Business and website owners often want to store credit card numbers for later to allow re-billing of customers for recurring orders, and allows easier checkout for repeat customers on a website. Normally rebilling requires businesses and websites to store credit card informations, which are subjected to strict PCI-DSS standards. The rebill transaction mechanism enables businesses and websites to perform re-billing of customers for recurring orders without having to store credit card information as this is handled by Globill.

9.11 *Refund*

A refund is the process of refunding a previously settled transaction. This will appear as a credit on the cardholder's credit card statement.

9.12 *Reversal*

A reversal is the process of reversing a previously captured, but not yet settled, transaction. It means that the transaction will never appear on the cardholder's credit card statement.

9.13 *Sale*

A sale is Authorization and Capture, see Authorization and Capture for more details

9.14 *Settle (Settlement)*

The process of settling a transaction is when the money is taken from the cardholder's account and put into the merchant's account. Once a transaction is settled, it will appear as a charge on the cardholder's credit card statement.