

SiD Integration Guide

Overview of the SiD integration process

Summary of Revisions

Version	Date	Changes Made
1.0.0	30 Nov 2008	<ul style="list-style-type: none"> Original document created
1.0.1	30 Dec 2008	<ul style="list-style-type: none"> Added SiD Logo and Style Guide
1.0.2	09 Jan 2009	<ul style="list-style-type: none"> Added Order Query Web Services
1.0.3	12 Jan 2009	<ul style="list-style-type: none"> Added Table of Content Fixed and added SID_STATUS in redirect variables Added receiptno and txnid fields in XML response for Order Query Web Services
1.0.4	17 Jan 2009	<ul style="list-style-type: none"> Updated Payment Flow chart Changed seller to merchant and buyer to customer
1.0.5	23 Feb 2009	<ul style="list-style-type: none"> Addition of HTTP notification
1.0.6	08 May 2009	<ul style="list-style-type: none"> Addition of SiD Consistent Field sample code
1.0.7	25 Jun 2009	<ul style="list-style-type: none"> Correction on Cold Fusion sample code. Addition of Demo bank login details
1.0.8	15 Jul 2009	<ul style="list-style-type: none"> Correction on Cold Fusion and PHP sample codes. Added consistent field concatenation order
1.0.9	10 Aug 2009	<ul style="list-style-type: none"> Addition of Appendix D & E. Limitation of reference field to 20 characters
1.1.0	21 Aug 2009	<ul style="list-style-type: none"> Change to VB.NET & C# sample code
1.1.1	25 Aug 2009	<ul style="list-style-type: none"> Added C# .Net sample code for Order Query Web Service
1.1.2	25 Jan 2010	<ul style="list-style-type: none"> Edited Receiving the Transaction Result section and removed sample codes and place into a zip file.
1.1.3	16 Feb 2010	<ul style="list-style-type: none"> Changed the description in the style guide
1.1.4	15 Mar 2010	<ul style="list-style-type: none"> Addition of SID_DEFAULT_REFERENCE field
1.1.5	03 May 2010	<ul style="list-style-type: none"> Added clarity to the Order Query Web Service descriptions
2.0.0	08 Jun 2010	<ul style="list-style-type: none"> New URL for SiDv2

2.0.1	21 Jun 2010	<ul style="list-style-type: none"> ▪ Added Implementation Error Code
2.0.2	23 Sep 2010	<ul style="list-style-type: none"> ▪ Changed the SiD logo and style guide
2.0.3	03 Feb 2011	<ul style="list-style-type: none"> ▪ Changed SiD order query ws URL
2.0.4	17 Jun 2014	<ul style="list-style-type: none"> ▪ Updated document styling and the Order Query Web Service XML response
2.0.5	21 June 2014	<ul style="list-style-type: none"> ▪ General Update
2.0.6	16 Nov 2015	<ul style="list-style-type: none"> ▪ Updated SiD Logo and Style Guide ▪ Updated Document Styling
2.0.7	06 Apr 2016	<ul style="list-style-type: none"> ▪ Updated SiD Payment Flow ▪ Updated Submitting a Transaction using SiD Buy Button section
2.0.8	20 Apr 2016	<ul style="list-style-type: none"> ▪ Added SiD Payment Link section ▪ Update to Web Service Fields Explained ▪ Improved Request and Response XML samples
2.0.9	28 Jul 2016	<ul style="list-style-type: none"> ▪ Updated Payment link Sample
2.1.0	06 June 2017	<ul style="list-style-type: none"> ▪ Updated Submitting a Transaction to SiD section ▪ Included SiD Embedded Solution ▪ Updated Order Query Web Service section ▪ Updated SiD Payment Link section ▪ Removed Appendix D and updated Appendix C ▪ Updated Contact Information section
2.1.1	17 May 2018	<ul style="list-style-type: none"> ▪ Updated Sample JSON Code ▪ Updated SidEft Function & Sample Code ▪ Updated Buy Button Consistent Field
2.1.2	28 Aug 2018	<ul style="list-style-type: none"> ▪ Updated Support Hours
2.1.3	12 Nov 2018	<ul style="list-style-type: none"> ▪ Updated to SiD Secure EFT
2.1.4	21 May 2019	<ul style="list-style-type: none"> ▪ Added Transaction Query Service ▪ Changed Reference limit
2.1.5	21 June 2019	<ul style="list-style-type: none"> ▪ Updated Http Notifications return statuses ▪ Added HttpNotification examples
2.1.6	1 November 2019	<ul style="list-style-type: none"> ▪ Updated SiD Embedded solution download location

2.1.7	13 January 2020	<ul style="list-style-type: none">▪ Removed Bower package reference due to being deprecated
2.1.8	25 January 2020	<ul style="list-style-type: none">▪ More strict limitation to Transaction Query Service

Contents

Contents	4
Overview	6
Payment Flow	6
Submitting a Transaction to SiD	7
SiD Redirect Solution	7
SiD Buy Button Fields	7
SiD Embedded (Iframe) Solution	10
Adding the SiD Secure EFT Popup to Your Webpage	10
Referencing Required Files.....	10
Generating the SiD HTML Buy Button	11
Buy Button Consistent Field.....	13
SiD Logo and Style Guide.....	14
Example 1: Displayed in a Drop-down List	14
Example 2: Display in Tick Boxes	14
SiD Logos	14
SiD Description	15
SiD FAQs	15
Receiving the Transaction Result	16
Method 1: Redirect Notification.....	16
Method 2: HTTP Notification.....	18
Method 3: Order Query Web Service.....	19
Web Service Fields Explained	20
Sample request string	23
Sample ERROR response string	23
Sample SUCCESSFUL response string	24
Notes Regarding Security.....	25
SiD Payment Link	26
Payment Link Web Service.....	26
Web Service Fields Explained	26
Sample request string	27

Sample ERROR response string	28
Sample SUCCESSFUL response string	28
Appendix A: Timeout and Failure Scenarios and Solutions	29
Appendix B: Additional Integration Resources	30
Appendix C: Implementation Error Code	30

Overview

SiD Secure EFT makes it possible for customers without credit cards to pay merchants using their internet banking.

The merchant implementation explained in this guide is very simple and requires a developer that can:

1. Build an HTML form, also known as the buy button.
2. Receive and parse URL and FORM variables back from SiD.

Payment Flow

The steps below outline the payment flow when the customer uses SiD:

1. The customer populates the shopping cart on the merchant's website and clicks on the Checkout button.
2. On the Checkout Page, they're presented with SiD as a payment option, which they select.
3. The customer is directed to the SiD Bank Selection Page where they need to select the bank they bank with.
4. Next, they're prompted to log into their internet banking by entering their usual login details (No login details are stored by SiD).
5. He/she then selects the bank account they wish to pay from (If they have more than one account linked to their internet banking profile).
6. They then complete the payment by entering a One-time PIN received from their bank, or responding to a push message received from their bank.
7. After the payment is successfully processed, SiD prompts the customer to print/download the receipt and once done, they're logged out of their internet banking.
8. The customer is automatically navigated back to the merchant's website where their order can be filled immediately, as SiD issues an instant, guaranteed payment confirmation to the merchant.

Submitting a Transaction to SiD

There are two integration options to submit a transaction to SiD:

1. SiD Redirect Solution
2. SiD Embedded Solution

SiD Redirect Solution

The SiD Redirect Solution is simply an HTML buy button which will redirect your customer to the SiD Secure EFT secure payment page. It needs to contain the following fields posting to the below URL:

<https://www.sidpayment.com/paySiD/>

SiD Buy Button Fields

Field Name	Required	Description
SID_MERCHANT	Yes	Set this field to the Merchant Identifier as issued to you by SiD when your account was created.
SID_CURRENCY	Yes	ZAR
SID_COUNTRY	Yes	ZA
SID_REFERENCE	Yes	Set this field to the merchant pass-through reference. This field should ideally be unique per transaction. This field will appear on all reports and will be passed back to the merchant when the customer is redirected back to the merchant's website. Use this field to reconcile to your system. Only Alphanumeric values and – (dash) and / (single slash) can be used. Max 20 characters.
SID_BUYER_REFERENCE	No	This field will populate the buyer statement reference field on the bank beneficiary pages and will appear on the buyer's statement as their own reference for the payment. If not passed, SID will populate this value with the SID_REFERENCE value. Only Alphanumeric values and – (dash) and / (single slash) can be used. Note the banks only allow a limit of 20 characters, and we will truncate any references longer than 20 characters. Also note that Standard Bank only allows 12 characters for this field, and we will truncate the beginning of the reference to accommodate any Standard bank transactions.

SID_AMOUNT	Yes	Transaction amount is decimal format, excluding any currency symbols or decimal commas, e.g.: 1955.60 or 12.00
SID_DEFAULT_REFERENCE	No	Certain banks require a new beneficiary to be added before allowing a once-off payment to be done to the new beneficiary.

		<p>This field will be used as a reference when adding a new beneficiary.</p> <p>If the buyer later uses this beneficiary manually (without using SiD), then this field will be used as the default payment reference. We suggest setting this field to the customer's account number on your system, or any other value that can be used to easily identify the buyer later. This field supports a maximum 20 Characters.</p> <ul style="list-style-type: none"> ▪ Numbers 0-9 ▪ Characters A-Z and a-z ▪ The character -(dash) ▪ The character / (slash)
SID_CUSTOM_01	No	<p>Can contain any custom data the merchant wants to pass through with the transaction. This field will appear on all reports and will be passed back to the merchant on the redirect and HTTP Notification Post. Only Alphanumeric values, - (hyphen), / (forward slash), _ (underscore), . (period) and @ (at) can be used. Spaces are not permitted and all requests should be URL encoded.</p>
SID_CUSTOM_02	No	<p>Can contain any custom data the merchant wants to pass through with the transaction. This field will appear on all reports and will be passed back to the merchant on the redirect and HTTP Notification Post. Only Alphanumeric values, - (hyphen), / (forward slash), _ (underscore), . (period) and @ (at) can be used. Spaces are not permitted and all requests should be URL encoded.</p>
SID_CUSTOM_03	No	<p>Can contain any custom data the merchant wants to pass through with the transaction. This field will appear on all reports and will be passed back to the merchant on the redirect and HTTP Notification Post. Only Alphanumeric values, - (hyphen), / (forward slash), _ (underscore), . (period) and @ (at) can be used. Spaces are not permitted and all requests should be URL encoded.</p>
SID_CUSTOM_04	No	<p>Can contain any custom data the merchant wants to pass through with the transaction. This field will appear on all reports and will be passed back to the merchant on the redirect and HTTP Notification Post. Only Alphanumeric values, - (hyphen), / (forward slash), _ (underscore), . (period) and @ (at) can be used. Spaces are not permitted and all requests should be URL encoded.</p>
SID_CUSTOM_05	No	<p>Can contain any custom data the merchant wants to pass through with the transaction. This field will appear on all reports and will be passed back to the merchant on the redirect and HTTP</p>

		Notification Post. Only Alphanumeric values, - (hyphen), / (forward slash), _ (underscore), . (period) and @ (at) can be used. Spaces are not permitted and all requests should be URL encoded.
SID_CONSISTENT	Yes	The merchant can use this field to prevent tampering of the buy button values. Please see the section entitled SiD Buy Button Consistent Field for guidelines to generating this field.

This is code for a sample buy button:

```
<FORM METHOD="POST" ACTION="https://www.sidpayment.com/paySID/">
  <INPUT TYPE="HIDDEN" NAME="SID_MERCHANT" VALUE="TESTGUIDE" />
  <INPUT TYPE="HIDDEN" NAME="SID_CURRENCY" VALUE="ZAR" />
  <INPUT TYPE="HIDDEN" NAME="SID_COUNTRY" VALUE="ZA" />
  <INPUT TYPE="HIDDEN" NAME="SID_REFERENCE" VALUE="INV-001-001" />
  <INPUT TYPE="HIDDEN" NAME="SID_BUYER_REFERENCE" VALUE="INV-002-001" />
  <INPUT TYPE="HIDDEN" NAME="SID_AMOUNT" VALUE="129.00" />
  <INPUT TYPE="HIDDEN" NAME="SID_DEFAULT_REFERENCE" VALUE="INV-002-001" />
  <INPUT TYPE="HIDDEN" NAME="SID_CONSISTENT"
  VALUE="CA59437B8063B1C2C3AEF548F91973724399D94F25ADF8F4E1A285CAED2DC13DF9AF80
  81289B9E3EBAD95B74E74D18B437C244A9AB71A7FA7442A3C3968803D1" />
  <INPUT TYPE="submit" NAME="PaySID" VALUE=" Pay using your internet
banking" />
</FORM>
```

Note: It is strongly advised to URL encode all requests.

SiD Embedded (Iframe) Solution

The SiD Embedded Solution allows merchants the ability to call the SiD payment window as a popup on their website. The benefits of this integration allow the merchant to leverage on the same level of security SiD provides with the Redirect method of integration while providing the customer with a seamless payment experience.

The below steps specify how to submit a payment using the SiD Embedded Solution. To receive the transaction response after the payment has been completed, please refer to the “Receiving The Transaction Result” section from this document.

Adding the SiD Secure EFT Popup to Your Webpage

In order to add the popup, there are three basic steps required:

1. Reference the required files in the `<HEAD>` or scripts section of your page.
2. Add an HTML button and give it a specific `id`.
3. Call the `sidEft` function when the button is clicked.

Referencing Required Files

The SiD Secure EFT popup requires:

- jQuery ^3.2.1
- `sidPopup.js`
- `sidPopupStyle.css`

Direct Download

The required files are available for download directly from the below repository:

<https://github.com/Sid-InstantEFT/sid-instant-eft-popup>

Generating the SiD HTML Buy Button

The following is required for your SiD Buy Button in order to implement the SiD Secure EFT Popup integration:

1. An id of sidButton
2. A call to the function SidEft or SidEftJson via the onclick event

There are two options when calling the SiD Secure EFT popup:

1. Utilising the SidEftJson function, you are able to pass in a JSON object containing all parameters
2. Utilising the SidEft function you can pass in the individual parameters to the function

SidEftJson Function

```
function SidEftJson(JsonParams);
```

NOTE: You are able to pass in the Json as a string or object.

Sample JSON Code

```
{
    "SID_MERCHANT": "TESTGUIDE",
    "SID_CURRENCY": "ZAR",
    "SID_COUNTRY": "ZA",
    "SID_REFERENCE": "INV-001-001",
    "SID_BUYER_REFERENCE": "INV-002-001",
    "SID_AMOUNT": "129.00",
    "SID_DEFAULT_REFERENCE": "INV-002-001",
    "SID_CUSTOM_01": "175598",
    "SID_CUSTOM_02": "843926",
    "SID_CUSTOM_03": "481779",
    "SID_CUSTOM_04": "941820",
    "SID_CUSTOM_05": "402892",
    "SID_CONSISTENT": "C82814C73A27E070AA27DD8F8E84DB05B39435A70ACB0DB41270528B7DBF2E67FA3920275BF97C90271DDAF842C6B5B0973016EF5AEA4BAFA620439939779AFC"
};
<button id="sidButton" OnClick="SidEftJson(JSON)">Pay by Instant EFT</button>
```

Please refer to [Buy Button Consistent Field](#) section for guidelines to generating the SID_CONSISTENT value.

SidEft Function

Please ensure the parameters are passed in with the correct order as specified in the [SID Buy Button Fields](#) section.

```
function SidEft(sid_merchant, sid_currency, sid_country, sid_reference,  
sid_default_reference, sid_amount, sid_custom_01,  
sid_custom_02, sid_custom_03, sid_custom_04, sid_custom_05, sid_consistent)
```

Sample Code

```
<button id="sidButton" onClick="SidEft(' TESTGUIDE ', 'ZAR', 'ZA', INV-001-001',  
INV-002-001', '129.00', ' INV-002-001', '175598', '843926', '481779', 941820, '402892',  
C82814C73A27E070AA27DD8F8E84DB05B39435A70ACB0DB41270528B7DBF2E67FA3920275BF97C90271DDA  
F842C6B5B0973016EF5AEA4BAFA620439939779AFC ')">Pay by Instant EFT</button>
```

Buy Button Consistent Field

By using the SID consistent field, the merchant can include a data validation field to stop tampering of the buy button variables.

After signup, SiD will issue the merchant with a secret key. This secret key should be kept protected at all times and should never be exposed in the clear in the button or on the website.

To generate the consistent field, concatenate all the buy buttons values along with the secret key into a new string applying the SHA512 hash function. The resulting hash value is then passed in the buy button as the value for the SID_CONSISTENT field.

Once SiD receives the request, SiD will follow the same procedure and use the values in the buy button and the agreed secret key to generate their own version of the consistent field. We will compare our version of the consistent field with that received in the buy button and only allow the transaction to continue should the two values match.

The order in which the fields need to be concatenated are:

1. SID_MERCHANT
2. SID_CURRENCY
3. SID_COUNTRY
4. SID_REFERENCE
5. SID_BUYER_REFERENCE (if applicable)
6. SID_AMOUNT
7. SID_DEFAULT_REFERENCE (if applicable)
8. SID_CUSTOM_01 (if applicable)
9. SID_CUSTOM_02 (if applicable)
10. SID_CUSTOM_03 (if applicable)
11. SID_CUSTOM_04 (if applicable)
12. SID_CUSTOM_05 (if applicable)
13. SID_SECRET_KEY

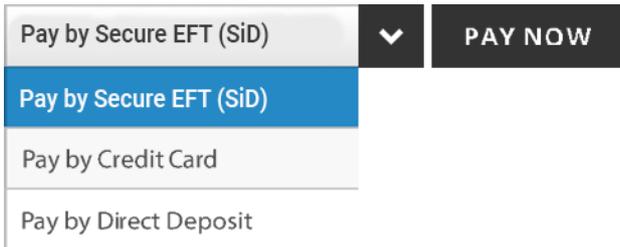
Always remember to hash the values in the same case as that you sent them in the buy button to SiD, as this will make a difference to the outcome. SiD uses UTF-8 encoding on the SID_CONSISTENT field. The SID_CONSISTENT field will always be an uppercase value.

SiD Logo and Style Guide

The SiD payment method should be presented on the merchant’s checkout/payment page. The following graphic illustrates how SiD is to be displayed on the page where the user can select SiD as their payment method.

Example 1: Displayed in a Drop-down List

Payment Options:



Example 2: Display in Tick Boxes

Pay Secure EFT (SiD)



Pay with Credit Card



SiD Logos

We ask that any of the below SiD logos be displayed in the footer of the merchant’s website:



The above images can be downloaded on the below page:

<http://sidpayment.com/merchant-resources/>

SiD Description

Merchants may wish to display more/educational information about SiD on other pages within their website. In this case the following information should be used:

Full Description

What is SiD Secure EFT?

SiD Secure EFT (Electronic Fund Transfer) has been one of South Africa's most trusted online payment methods since 2007. SiD allows you to make safe online payments directly from your bank account.

Don't worry if you've never used SiD before. It's easy! You don't have to sign up in order to use SiD, and you don't require any banking or payment information from the merchant prior to completing your payment. SiD automatically populates the payment details for you. All you need to do is follow these 5 simple steps:

1. Select the SiD payment method at checkout.
2. Select your bank.
3. Enter your usual internet banking login details. Rest assured, you are logging directly into your internet banking account and your bank security will remain in place.
4. Select the bank account you wish to pay from.
5. Complete the payment by entering a One-time PIN received from your bank, or responding to a message received from your bank (You will be directed to the receipt page on the merchant's website once payment has been completed).

The merchant will receive instant payment confirmation, so you don't have to send through proof of payment or wait for funds to clear. With SiD your purchase is completed in real time and your order can be filled immediately. SiD's automated payment process makes online payment quick, convenient and secure! For more information, visit www.sidpayment.com.

SiD FAQs

To address frequently asked end user questions, the FAQs outlined in the below document should be displayed on the merchant's website, e.g. on the FAQ, Help or Support page:

<https://sidpayment.com/wp-content/uploads/2018/08/sid-user-faqs-2018-v2.pdf>

Receiving the Transaction Result

All transactions should be considered as CANCELLED / DECLINED until such time that you receive the COMPLETED notification

The following 3 methods are provided for receiving both Completed and Cancelled transaction results from SiD. We strongly recommend that all 3 methods be implemented:

1. Redirect Notification
2. HTTP Notification
3. Order Query Web Service (**Note: This is now Deprecated**)
4. Transaction Query Service (**Note: this replaces Order Query Web Service**)

Please note: All the return variables in the URL are URL encoded. Please decode the variables before generating the Notification consistent key.

Method 1: Redirect Notification

Once the payment has been completed or cancelled, the customer will be redirected back to the merchant's website. A receipt/confirmation page should be displayed that is consistent with their existing website's look and feel.

The URL where the customer is redirected to is locked in the SiD system for security reasons and can only be changed by authorised SiD staff. To change this, please get the account holder to send the request in writing to support@sidpayment.com.

The following variables are sent to the merchant when the customer is redirected back to the merchant's website.

Field Name	Description
SID_STATUS	The result of the transaction. Possible returned values are: COMPLETED CANCELLED CREATED READY See the section titled 'Status Code Explained' for a detailed explanation on this field.
SID_MERCHANT	Contains the merchant code as assigned to the merchant by SiD when the account was created.
SID_CURRENCY	ZAR

SID_COUNTRY	ZA
SID_REFERENCE	Contains the reference number as passed to SiD in the buy button.
SID_AMOUNT	The amount of the transaction as captured on the internet banking receipt page.
SID_BANK	The bank name of the bank that the customer paid from.
SID_DATE	The date and time when the transaction was completed. All dates are given in GMT +02:00.
SID_RECEIPTNO	The bank receipt number as issued and appearing on the internet banking receipt page.
SID_TNXID	Contains the unique SiD System Transaction ID for this transaction.
SID_CUSTOM_01	Contains any custom data the merchant sent through in the buy button in the SID_CUSTOM_01 field.
SID_CUSTOM_02	Contains any custom data the merchant sent through in the buy button in the SID_CUSTOM_02 field.
SID_CUSTOM_03	Contains any custom data the merchant sent through in the buy button in the SID_CUSTOM_03 field.
SID_CUSTOM_04	Contains any custom data the merchant sent through in the buy button in the SID_CUSTOM_04 field.
SID_CUSTOM_05	Contains any custom data the merchant sent through in the buy button in the SID_CUSTOM_05 field.
SID_CONSISTENT	Security field that should be used by the merchant to determine the authenticity of this message. To generate the consistent field concatenate all the above fields in the same order with the secret key and SHA512 hash the data (Note this is not the same as the Buy Button Consistent Field).

Method 2: HTTP Notification

Once the payment has been completed or cancelled, SiD can post a HTTP notification to a pre-defined URL on the merchant's server.

This service will try to run for a maximum of 3 times with a waiting period between each retry.

We strongly recommend that the HTTP Notification be implemented in addition to the Redirect Notification. This will reduce the number of client queries when the Redirect Notification fails. Another nicety would be to confirm that the status and transactional information is identical between the HTTP Notification and the Redirect Notification for extra security.

The URL where the notification is posted to is locked in the SiD system for security reasons and can only be changed by authorised SiD staff. To change this, please get the account holder to send the request in writing to support@sidpayment.com.

All the following parameters will be concatenated into a Url parameter sequence, Url encoded, and placed in the body of a standard HTTP post.

Field Name	Description
SID_STATUS	The result of the transaction. Possible returned values are: COMPLETED CANCELLED See the section titled 'Status Code Explained' for a detailed explanation on this field.
SID_MERCHANT	Contains the merchant code as assigned to the merchant by SiD when the account was created.
SID_COUNTRY	ZA
SID_CURRENCY	ZAR
SID_REFERENCE	Contains the reference number as passed to SiD in the buy button.
SID_AMOUNT	The amount of the transaction as captured on the internet banking receipt page.
SID_BANK	The bank name of the bank that the customer paid from.
SID_DATE	The date and time when the transaction was completed. All dates are given in GMT +02:00.

SID_RECEIPTNO	The bank receipt number as issued and appearing on the internet banking receipt page.
SID_TNXID	Contains the unique SiD System Transaction ID for this transaction.
SID_CUSTOM_01	Contains any custom data the merchant sent through in the buy button in the SID_CUSTOM_01 field.
SID_CUSTOM_02	Contains any custom data the merchant sent through in the buy button in the SID_CUSTOM_02 field.
SID_CUSTOM_03	Contains any custom data the merchant sent through in the buy button in the SID_CUSTOM_03 field.
SID_CUSTOM_04	Contains any custom data the merchant sent through in the buy button in the SID_CUSTOM_04 field.
SID_CUSTOM_05	Contains any custom data the merchant sent through in the buy button in the SID_CUSTOM_05 field.
SID_CONSISTENT	Security field that should be used by the merchant to determine the authenticity of this message. To generate the consistent field concatenate all the above fields in the same order with the secret key and SHA512 hash the data (Note this is not the same as the Buy Button Consistent Field).

Response

Successful transaction:

```
SID_STATUS=COMPLETED&SID_MERCHANT=SENSEPOST1&SID_COUNTRY=ZA&SID_CURRENCY=
ZAR&SID_REFERENCE=Test&SID_AMOUNT=1%2c88&SID_BANK=SID+Test+COMPLETED&SID_DATE
=2019%2f06%2f21+10%3a11%3a39&SID_RECEIPTNO=SiD+Test+COMPLETED&SID_TNXID=89966de2-
9b88-47c9-be0a-
2fe80f7a2966&SID_CUSTOM_01=717743&SID_CUSTOM_02=997928&SID_CUSTOM_03=373012&SID_
CUSTOM_04=501619&SID_CUSTOM_05=233074&SID_CONSISTENT=C00DF894380DFDF9C9DD5D112
FEE621485950EDBA15DC79D0294C3BE061207406E85240AF18C326F50ABCE0BAC6DCC2F0799A9EB
9FE392CF85C977C98BA65DAA
```

Cancelled transaction:

```
SID_STATUS=COMPLETED&SID_MERCHANT=SENSEPOST1&SID_COUNTRY=ZA&SID_CURRENCY=
ZAR&SID_REFERENCE=Test&SID_AMOUNT=1%2c88&SID_BANK=SID+Test+COMPLETED&SID_DATE
=2019%2f06%2f21+10%3a11%3a39&SID_RECEIPTNO=SiD+Test+COMPLETED&SID_TNXID=89966de2-
9b88-47c9-be0a-
2fe80f7a2966&SID_CUSTOM_01=717743&SID_CUSTOM_02=997928&SID_CUSTOM_03=373012&SID_
CUSTOM_04=501619&SID_CUSTOM_05=233074&SID_CONSISTENT=C00DF894380DFDF9C9DD5D112
FEE621485950EDBA15DC79D0294C3BE061207406E85240AF18C326F50ABCE0BAC6DCC2F0799A9EB
9FE392CF85C977C98BA65DAA
```

Method 3: Order Query Web Service **(Deprecated)**

This method is now deprecated in favour of the new Transaction Query Service detailed in method 4. The documentation for method 3 is simply as a legacy guide for those who have already implemented this.

The Order Query Web Service allows you to query the status of any transaction during or after the transaction. This is useful to keep the merchant's system in sync with SiD.

Implementation can be done in 2 ways, you can check for individual transaction's status and information or you can query numerous transactions in a single call.

We strongly recommend that the Order Query Web Service be implemented in addition to the Redirect and HTTP Notification methods. This will reduce the number of client queries if the Redirect and/or the HTTP Notification methods fail. Another benefit would be to confirm that the status and transactional information is identical between the Order Query Web Service and the Redirect and HTTP Notification for extra security.

To retrieve the information, the merchant would have to connect to the SiD Web Service interface. The Web Service and method details are shown below.

Web Service WSDL File: <https://www.sidpayment.com/api/?wsdl>

Method Name: `sid_order_query`

The method accepts one XML string as argument, named "sid_order_query_request". The method returns a XML string as output.

Please refer to the XML examples below for sample request and response strings.

The XSD (XML Schema Definition) files for the request and response data can be found at the below URLs.

https://www.sidpayment.com/api/schemas/sid_order_query_request.xsd

https://www.sidpayment.com/api/schemas/sid_order_query_response.xsd

Web Service Fields Explained

sid_order_query_request:

The `sid_order_query_request` XML string contains the merchant information (and login details). The string also needs to contain the order information of the transaction (or transactions) being queried. The merchant would have to supply the same data as used in the buy button when the transaction was initially created.

Field Name	Description
<code>../merchant/code</code>	The SID_MERCHANT Code, provide by SiD Support
<code>../merchant/uname</code>	SiD Order Query Web Service username, provided by SiD Support.
<code>../merchant/pword</code>	SiD Order Query Web Service password, provided by SiD Support.
<code>../ orders/transaction/country</code>	The country code the transaction is taking place in.
<code>../ orders/transaction/currency</code>	The currency code the transaction is taking place in.
<code>../ orders/transaction/amount</code>	The amount that the transaction will be processed for.
<code>../ orders/transaction/reference</code>	The reference for the transaction.

sid_order_query_response:

The `sid_order_query_response` string contains the request outcome. Only responses where the error code is equal to 0 (zero) can be seen as successful responses.

Field Name	Description
<code>../data/outcome/errorcode</code>	This element contains the request outcome in the XML element. Only responses where the error code is equal to 0 (zero) can be seen as successful responses.
<code>../data/outcome/errordescription</code> <code>../data/outcome/errorsolution</code>	Where the <code>error_code</code> field is not equal to zero, these two fields will have values which will explain to the merchant what has gone wrong.
<code>../data/merchant/code</code>	The Merchants Code
<code>../data/orders/transaction/status</code>	<p>COMPLETED</p> <p>Orders with a status of COMPLETED was successfully paid and the merchant will receive the funds in their bank account within 1 to 2 business days, depending on the bank. This is the ONLY transaction status that a merchant should use to credit a customer.</p> <p>CANCELLED</p> <p>The order was cancelled. The merchant should NOT credit the customer.</p> <p>CREATED</p> <p>Customer has selected their bank but has not logged into their internet banking account as yet to begin their transaction.</p> <p>READY</p> <p>Customer has selected their bank and has logged into their bank account to commence with their payment.</p>
<code>../data/orders/transaction/country/code</code> <code>../data/orders/transaction/country/name</code>	This will contain the country code and country name that the transaction took place in.

<pre>../data/orders/transaction/currency/code</pre> <pre>../data/orders/transaction/currency/name</pre> <pre>../data/orders/transaction/currency/symbol</pre>	This will contain the currency code, currency name and currency symbol for the transaction.
<pre>../data/orders/transaction/bank/name</pre>	This will be the Bank that the transaction took place from.
<pre>../data/orders/transaction/amount</pre>	The amount of the transaction.
<pre>../data/orders/transaction/reference</pre>	The reference of the transaction.
<pre>../data/orders/transaction/receiptno</pre>	The receipt number that was generated by the Bank.
<pre>../data/orders/transaction/tnxid</pre>	The Transaction ID.
<pre>../data/orders/transaction/date_created</pre>	This is the date and time when the SiD transaction was created. The customer has selected their preferred bank but has not logged into their internet banking account as yet.
<pre>../data/orders/transaction/date_ready</pre>	This is the date and time when the customer has logged into their internet banking account to commence with their payment.
<pre>../data/orders/transaction/date_completed</pre>	<p>This is the date and time when the transaction was either "completed" for successful transactions or "cancelled" for cancelled transactions.</p> <p>For example:</p> <p>If the customer clicked on the cancel button to opt-out of the transaction, this field will contain the date and time we received the cancellation request.</p> <p>If the customer turns off the power to his PC, this field will be blank. We never received any request confirming or cancelling this transaction.</p> <p>If the customer completes the transaction, this field will contain the date and time the SiD database received the request to complete the transaction.</p>

Request

```
<?xml version="1.0" encoding="UTF-8"?>
  <sid_order_query_request>
    <merchant>
      <code>PHONE</code>
      <uname>sid@phonewarehouse.co.za</uname>
      <pword>112233</pword>
    </merchant>
    <orders>
      <transaction>
        <country>ZA</country>
        <currency>ZAR</currency>
        <amount>106.00</amount>
        <reference>8</reference>
      </transaction>
      <transaction>
        <country>ZA</country>
        <currency>ZAR</currency>
        <amount>106.00</amount>
        <reference>7</reference>
      </transaction>
    </orders>
  </sid_order_query_request>
```

Response

Error:

```
<?xml version="1.0" encoding="UTF-8"?>
<sid_order_queryResponse xmlns="http://tempuri.org/">
  <sid_order_queryResult>
    <sid_order_query_response>
      <data signature="jaHL4CL8Xbmtk968NjztNGDDfAWIkzh+11c=">
        <outcome errorcode="SIDAPI0001" errordescription="Login
Failed." errorsolution="Please correct details." />
        FAILED
      </data>
    </sid_order_query_response>
  </sid_order_queryResult>
</sid_order_queryResponse>
```

Success:

```
<?xml version="1.0" encoding="UTF-8"?>
<sid_order_query_response>
  <data>
    <outcome errorcode="0" errordescription="" errorsolution="" />
    <merchant>
      <code>PHONE</code>
    </merchant>
    <orders>
      <transaction>
        <status>CANCELLED</status>
        <country>
          <code>ZA</code>
          <name>South Africa</name>
        </country>
        <currency>
          <code>ZAR</code>
          <name>South Africa</name>
          <symbol>R</symbol>
        </currency>
        <bank>
          <name>ABSA</name>
        </bank>
        <amount>106.00</amount>
        <reference>8</reference>
        <receiptno>0001</receiptno>
        <tnxid>e73a3d6c-46b5-42cf-j3c3-a33c95b0d864</tnxid>
        <date_created>2014/04/25 10:45:35 AM</date_created>
        <date_ready />
        <date_completed />
      </transaction>
      <transaction>
        <status>COMPLETED</status>
        <country>
          <code>ZA</code>
          <name>South Africa</name>
        </country>
        <currency>
          <code>ZAR</code>
          <name>South Africa</name>
          <symbol>R</symbol>
        </currency>
      </transaction>
    </orders>
  </data>
</sid_order_query_response>
```

```
</currency>
  <bank>
    <name>ABSA</name>
  </bank>
  <amount>106.00</amount>
  <reference>9</reference>
  <receiptno>0002</receiptno>
  <tnxid>b5e28622-dbf9-415a-9625-0dc4239d1c04</tnxid>
  <date_created>2014/04/25 10:44:09 AM</date_created>
  <date_ready />
  <date_completed />
</transaction>
</orders>
</data>
  <signature>VeKRu60x6krPnH5d2ubqTG9QvoE0tBePPpnmfq4r0=</signature>
</sid_order_query_response>
```

Method 4: Transaction Query Service

The Transaction Query Service is a standard REST API GET call, that takes URL parameters to filter the results

Request

GET:

<https://www.sidpayment.com/services/api/v30/transactions/query?>

Headers:

This API function requires Basic auth consisting of the Merchants administrator username and password (same credentials used to log into merchant portal). These credentials lock this query down to only show results from this merchant.

Below is an Example of what the credentials should look like once base 64 encoded.

Authorization : Basic b25lQHNIbnNlcG9Hyg2QTVGQjRDQUVEQkJE

Parameters Explained:

All parameters are optional, however at least one param must be supplied. If one of the date parameters is supplied, both must be supplied. These parameters should be concatenated to the end of the URL as standard URL Query Parameters (e.g. .../query?one=1&two=2&three=3)

Param	Description	Values
startDate	<p>A string representing the start date of the date range.</p> <p>Note: that the date range may not exceed 3 months, and if one of the date parameters is supplied, both must be supplied. Also the start date must be before (earlier) the end date. And if no date range is supplied at all, a default range of the last 24 hours is applied to all queries, except when a Transaction Id param is supplied, then no date limitation is applied.</p> <p>Format: yyyy-MM-dd</p>	2019-01-01
endDate	<p>A string representing the end date of the date range.</p> <p>Note: that the date range may not exceed 3 months, and if one of the date parameters is supplied, both must be supplied. Also the start date must be before (earlier) the end date. And if no date range is supplied at all, a default range of the last 24 hours is applied to all queries, except when a Transaction Id param is supplied, then no date limitation is applied.</p> <p>Format: yyyy-MM-dd</p>	2019-02-01
status	<p>A string representing one of the 4 statuses a transaction can be in.</p> <p>Options: CREATED, READY, CANCELLED, COMPLETED</p>	CREATED READY CANCELLED COMPLETED

sellerReference	<p>A string representing the seller reference field which is supplied by the merchant.</p> <p>Note: this string is limited to only 20 characters, the same limitation that is applied to all references fields when creating a transaction.</p>	INV-01-001
amount	<p>A string representing the amount field field which is supplied by the merchant.</p> <p>Note: this must be in a decimal value, using a full stop (.) as decimal separator, and must not contain any other separation characters, or currency symbols.</p>	123.45
tnxId	<p>This is a GUID field, this value is generated by SiD when a transaction is created and is returned at the end of a transaction through Redirect, and/or Http Notifications.</p> <p>Note: If this field is populated without any date range specified, a default date range is not applied, as it is with all other parameters. Because you can only ever receive one transaction when querying by transaction Id.</p>	33dc0961-7734-4e95-a817-041a555ed49d
skip	<p>This is an int value, which represents the number of Transaction records to skip in the result set. Currently we limit all payloads to only 10 000 return records, and the skip value can be used to select the next 10 000 records in subsequent calls.</p> <p>Note: If not supplied, this defaults to 0.</p> <p>e.g. If a query is made with a skip value of 0 The following snippet from the json result shows a total of 93 757 records for this query, but the payload only contains 10 000 records, starting at 1 - 10 000</p> <pre>"totalRecords": 93757, "firstRecord": 1, "lastRecord": 10000,</pre> <p>e.g. If a subsequent query with the same parameters is made, but the skip value is set to 10 000 (which matches the 'lastRecord' field of the previous result) The following snippet from the json result still shows a total of 93 757 records for this query, and the payload still only contains 10 000 records, but the starting record is now 10 001 - 20 000</p> <pre>"totalRecords": 93757, "firstRecord": 10001, "lastRecord": 20000,</pre> <p>Using this mechanism, all records in this example can be returned by making a total of 10 api calls using the same parameters but incrementing the skip value by 10 000 records each time. The result set having used a skip value of 90 000 will then look as follows. If the 'lastRecord' field matches the 'totalRecords' field, all transactions have been returned throughout all the api calls.</p> <pre>"totalRecords": 93757, "firstRecord": 90001, "lastRecord": 93757,</pre>	10000

Request Example:

Get: <https://www.sidpayment.com/services/api/v30/transactions/query?startDate=2019-01-01&endDate=2019-05-01&status=COMPLETED&sellerReference=INV-001-001&amount=1.33&skip=0>

Headers: Authorization : Basic b25lQHNIbnNlcG9Hyg27jb206QTVGQjRDQUVEQkJE

Response: 200 OK

Payload:

```
{
  "queryStartDate": "2019-01-01 00:00:00",
  "queryEndDate": "2019-05-01 00:00:00",
  "queryStatus": "COMPLETED",
  "querySellerReference": "INV-001-001",
  "queryAmount": "1.33",
  "queryTxnId": null,
  "querySkip": 0,
  "totalRecords": 2,
  "firstRecord": 1,
  "lastRecord": 2,
  "transactions": [
    {
      "amount": "1.33",
      "bank": "Nedbank",
      "buyerReference": null,
      "country": "ZA",
      "currency": "ZAR",
      "custom01": "692565",
      "custom02": "526559",
      "custom03": "137069",
      "custom04": "874221",
      "custom05": "606000",
      "dateCompleted": "2019-02-15 08:47:21",
      "dateCreated": "2019-02-15 08:45:55",
      "defaultReference": "INV-002-001",
      "merchant": "SENSEPOST1",
      "receiptNumber": "20190215/NEDBANK/002065488770",
      "sellerReference": "INV-001-001",
      "status": "COMPLETED",
      "transactionId": "2384c8f5-e564-4b66-87bd-6f11d791fce5"
    },
    {
      "amount": "1.33",
      "bank": "Nedbank",
      "buyerReference": null,
      "country": "ZA",
      "currency": "ZAR",
      "custom01": "782672",
      "custom02": "614862",
      "custom03": "126729",
```

```
"custom04": "390000",
"custom05": "836653",
"dateCompleted": "2019-04-04 05:58:37",
"dateCreated": "2019-04-04 05:57:23",
"defaultReference": "INV-002-001",
"merchant": "SENSEPOST1",
"receiptNumber": "20190404/NEDBANK/002119985934",
"sellerReference": "INV-001-001",
"status": "COMPLETED",
"transactionId": "7754aba1-f420-4474-9350-136abfbe4acc"
}
]
}
```

Response

The response will be in a standard JSON format and contains the following fields.

Main Fields Explained:

Fields	Description
queryStartDate queryEndDate queryStatus querySellerReference queryAmount queryTxId querySkip	At the top of the payload are all the parameters that were supplied to the query, these fields can be used to validate what data is being returned in the payload
totalRecords firstRecord lastRecord	<p>These record count fields explain how many records there are in total, and how many records are returned in this payload. Currently we limit all payloads to only 10 000 return records per API call, therefore multiple calls may have to be made with an incremented skip parameter to obtain all records.</p> <p>The following snippet from the json result shows a total of 93 757 records for this query, but the payload only contains 10 000 records, starting at 1 - 10 000</p> <pre>"totalRecords": 93757, "firstRecord": 1, "lastRecord": 10000,</pre> <p>The following snippet from the json result still shows a total of 93 757 records for this query, and the payload still only contains 10 000 records, but the starting record is now 10 001 - 20 000</p> <pre>"totalRecords": 93757, "firstRecord": 10001, "lastRecord": 20000,</pre> <p>Using this mechanism, all records in this example can be returned by making a total of 10 api calls using the same parameters but incrementing the skip value by 10 000 records each time. The result set having used a skip value of 90 000 will then look as follows. If the 'lastRecord' field matches the 'totalRecords' field, all transactions have been returned throughout all the api calls.</p> <pre>"totalRecords": 93757, "firstRecord": 90001, "lastRecord": 93757,</pre>
transactions	This is a list of all the returned transactions from the query, this is explained in detail below

Transaction Fields Explained:

Fields	Description
amount	The amount that the transactions was requested to processed
bank	The bank that the user selected to process the EFT from
buyerReference	The buyer reference field that the merchant supplied
country	The country code
currency	The currency code
custom01 custom02 custom03 custom04 custom05	All the custom fields that the merchant supplied when creating the transaction
dateCompleted	The date and time that the transaction was completed on. This also indicates the date and time that the funds were transferred on.
dateCreated	The date and time that the transaction was created
defaultReference	The default reference field that the merchant supplied
merchant	The merchant code that the transaction relates to. Note: This matches the credentials that were passed in through the basic auth header.
receiptNumber	The receipt number that was supplied by the bank for the EFT transaction
sellerReference	The seller reference field that the merchant supplied
status	The status of the transaction. Options: CREATED, READY, CANCELLED, COMPLETED
transactionId	This is a GUID field, this value is generated by SID when a transaction is created and is returned at the end of a transaction through Redirect, and/or Http Notifications.

Response Example:

```
{
  "queryStartDate": "2019-01-01 00:00:00",
  "queryEndDate": "2019-05-01 00:00:00",
  "queryStatus": "COMPLETED",
  "querySellerReference": "INV-001-001",
  "queryAmount": "1.33",
  "queryTnxId": null,
  "querySkip": 0,
  "totalRecords": 2,
  "firstRecord": 1,
  "lastRecord": 2,
  "transactions": [
    {
      "amount": "1.33",
      "bank": "Nedbank",
      "buyerReference": null,
      "country": "ZA",
      "currency": "ZAR",
      "custom01": "692565",
      "custom02": "526559",
      "custom03": "137069",
      "custom04": "874221",
      "custom05": "606000",
      "dateCompleted": "2019-02-15 08:47:21",
      "dateCreated": "2019-02-15 08:45:55",
      "defaultReference": "INV-002-001",
      "merchant": "SENSEPOST1",
      "receiptNumber": "20190215/NEDBANK/002065488770",
      "sellerReference": "INV-001-001",
      "status": "COMPLETED",
      "transactionId": "2384c8f5-e564-4b66-87bd-6f11d791fce5"
    }
  ]
}
```

Possible Responses:

HTTP codes	Reason
200 : OK	We received your query, successfully processed it, and responded with the results
400 : Bad Request	One or more of the input parameters was not correct
401 : Unauthorized	Either the basic auth was not supplied, or authentication failed
500 : Internal Server Error	An internal to SiD error occurred

Notes Regarding Security

1. With regards to the Return and Notification URLs, it is crucial to also validate the SID_CONSISTENT field in the Return and Notification URL response. This field is designed to stop tampering of any of the response fields.
2. There cannot be a COMPLETED transaction without the receipt number being populated (passed back as SID_RECEIPTNO).
3. All information passed back that was initially sent through to us, such as the merchant code, country and currency should be validated to make sure nothing has changed.
4. Between the different notifications methods, validate that the status field (passed back as SID_STATUS) are all the same.

SiD Payment Link

SiD Payment Link offers a service which allows merchants to generate a short-link which they can then place on their invoices, or embed into emails which allow for a SiD payment to take place without the need for a website. SiD offers this service through the use of a Web Service.

Payment Link Web Service

To retrieve the Payment Link, the merchant would have to connect to the SiD Web Service interface. The Web Service and method details are explained below.

Web Service WSDL File: <https://www.sidpayment.com/webservices/?wsdl>

Method Name: `sid_biller_link`

The method accepts one XML string as argument, named "sid_biller_link_request". The method returns an XML string as output.

Please refer to the XML examples below for sample request and response strings.

The XSD (XML Schema Definition) files for the request and response data can be found at the below URLs.

https://www.sidpayment.com/webservices/schemas/sid_biller_link_request.xsd

https://www.sidpayment.com/webservices/schemas/sid_biller_link_response.xsd

Web Service Fields Explained

sid_biller_link_request:

The `sid_biller_link_request` XML string contains the merchant information and login details. The string also needs to contain the order information of the transaction (or transactions) which the merchant is wanting to process.

Field Name	Description
<code>../merchant/code</code>	The SID_MERCHANT Code, provide by SiD Support
<code>../merchant/username</code>	SiD Order Query Web Service username, provided by SiD Support.
<code>../merchant/password</code>	SiD Order Query Web Service password, provided by SiD Support.
<code>../transaction/amount</code>	The amount that the transaction will be processed for.
<code>../transaction/custom01 - custom05</code>	Contains any custom data the merchant sent through.

../transaction/country/code	The country code the transaction is taking place in.
../transaction/currency/code	The currency code the transaction is taking place in.
../transaction/reference/merchant	Contains the reference number for the merchant.
../transaction/reference/buyer	Contains the reference number for the buyer.

sid_biller_link_response:

The `sid_order_query_response` string contains the request outcome. Only responses where the error code is equal to 0 (zero) can be seen as successful responses.

Field Name	Description
../data/transaction/link	This is the unique link which when accessed will enable a transaction to be processed.
../data/outcome/errorCode	This element contains the request outcome in the XML element. Only responses where the error code is equal to 0 (zero) can be seen as successful responses.
../data/outcome/errorDesc ../data/outcome/errorSolution	Where the <code>errorCode</code> field is not equal to zero, these two fields will have values which will explain to the merchant what has gone wrong.

Sample request string

```
<?xml version="1.0" encoding="UTF-8"?>
<sid_biller_link_request>
  <merchant code="PHONE" username="sid@phonewarehouse.co.za" password="112233" />
  <transaction amount="1.77" custom01="cust-1" custom02="cust-2" custom03="cust-3"
    custom04="cust-4" custom05="cust_5">
    <country code="ZA" />
    <currency code="ZAR" />
    <reference merchant="INV-001-001" buyer="INV-002-001" />
  </transaction>
</sid_biller_link_request>
```

Sample ERROR response string

```
<?xml version="1.0" encoding="UTF-8"?>
<sid_biller_link_response>
  <data>
    <outcome errorCode="BILLER002" errorDesc="" errorSolution="Please ensure
      that your xml string is structured correctly" />
  </data>
</sid_biller_link_response>
```

Sample SUCCESSFUL response string

```
<?xml version="1.0" encoding="UTF-8"?>
<sid_biller_link_response>
  <data>
    <outcome errorCode="0" errorDesc="" errorSolution="" />
    <transaction link="https://sidpay.info?000000" />
  </data>
</sid_biller_link_response>
```

Appendix A: Timeout and Failure Scenarios and Solutions

	Possible Failures	Solution
1	Notification timed out between SiD's server and the Merchant's server	<p>Payment was successful but SiD failed to notify the merchant.</p> <p>Causes that result in these timeouts are:</p> <ul style="list-style-type: none"> SiD's internet connection or bandwidth dropped The merchant's internet connection or bandwidth dropped SiD's website or server encounters problems The merchant's website or server encounters problems <p>Solution: Merchant should query SiD using the Order Query Web Service for any timeout transactions. We recommend every 15 minutes for 4 times.</p>
2	Browser/Client timed out <u>without</u> the funds being debited from the client's bank account	<p>The client's payment was unsuccessful.</p> <p>Causes that result in these timeouts are:</p> <ul style="list-style-type: none"> Client's internet connection or bandwidth dropped Client's internet bandwidth is too slow to complete the transaction The bank's internet connection or bandwidth dropped The bank's website or server encounters problems <p>Solution: The client should retry the payment.</p>
3	Browser/Client timed out <u>with</u> the funds being debited from the client's bank account	<p>The client's payment was successfully debited from their bank account.</p> <p>Causes that result in these timeouts are:</p> <ul style="list-style-type: none"> Client's internet connection or bandwidth dropped after the payment was submitted to the bank The bank's website or server encountered a problem generating the receipt SiD's website or server encounters problems <p>Solution: The merchant will receive extra payment in their bank account which is not listed in the SiD report. The merchant should credit the client and deliver on the order.</p>
4	Client manually completes a payment using a previous reference number (without using SiD)	<p>We've noticed that ABSA bank stores the beneficiary details and reference number used in once-off payments. If the client manually logs into their internet banking, and make a second payment to the same merchant, after first using SiD. The first reference number is used in the second payment. The result is that you will receive 2 payments with the same reference number.</p> <p>Solution: We recommend the merchant search for duplicated transactions from the same client and advice the client of the corrective action.</p>

Appendix B: Additional Integration Resources

Sample code and additional integration resources are available from:

<http://sidpayment.com/getting-started/>

Appendix C: Implementation Error Code

These could be common errors whilst implementing SID.

Error Code	Cause	Solution
PAYSID0001	SID_MERCHANT is blank or invalid.	Ensure this field has a value and it is correct.
PAYSID0002	SID_COUNTRY is blank or invalid.	Ensure this field has a value and it is correct.
PAYSID0003	SID_CURRENCY is blank or invalid.	Ensure this field has a value and it is correct.
PAYSID0004	SID_AMOUNT is blank or invalid.	Ensure this field has a legitimate value.
PAYSID0005	Transaction amount too low.	Confirm seller limits with customer support.
PAYSID0006	Transaction amount too high.	Confirm seller limits with customer support.
PAYSID0007	SID_REFERENCE is invalid	The SID_REFERENCE field may not be left blank and may only be a maximum of 20 characters long.
PAYSID0008	Merchant has no terminals loaded.	Confirm that your merchant has terminals loaded with customer support.
PAYSID0009	SID_CONSISTENT does not match.	Ensure that the required fields are concatenated in the correct order. Do not HTML/URL encode any fields before concatenating the fields. Ensure you include your secret key before hashing the string. Ensure you are using the SHA512 hash function. If required, ensure that all characters in the hash are upper case before sending the consistent field.
PAYSID0025	Conversion to number failed on SID_AMOUNT.	Ensure that the amount does not contain any currency symbols or decimal commas. A decimal point is allowed to indicate cents.
PAYSID0030	A security violation occurred.	Ensure that the Consistent key passed in is properly generated as per error PAYSID0009.

Contact Details

Feel free to contact our Customer Care Centre if you have any questions. We're available from 8am to 6pm Monday – Friday & 9am to 4pm on Saturday - Sunday & public holidays.

support@sidpayment.com | +27(0) 11 555 1101 | www.sidpayment.com

