



# API Documentation

API TYPE:  
**Digital KYC**

CALL FUNCTION:  
**getKYC**

Release date: 24 November 2017

## REVISION

Version 1.0  
24 November 2017

This document and the contents thereof are protected by copy right laws and may not be copied or re-distributed in any way.

**Document owner:** PBSA (Pty) Ltd

**Technical contact:** Philip Csaplar

**Authorised by:** Leon van der Merwe

## Contents

1. Introduction
2. Service Overview
3. SOAP Service
4. XML OUTPUT
5. PHP Client Connections
6. Management Console

Signed off by:

X \_\_\_\_\_

This document is signed with an advanced electronic signature which protects the integrity of the document and the identity of the signer.

Please ensure that the signature is VALID (content is unchanged) by clicking on the signature before using this document.

This document is tamper-evident and may not be changed or altered in any way.

Please ensure that the signature is VALID (content is unchanged) by clicking on the signature before using this document.

This document is tamper-evident and may not be changed or altered in any way.

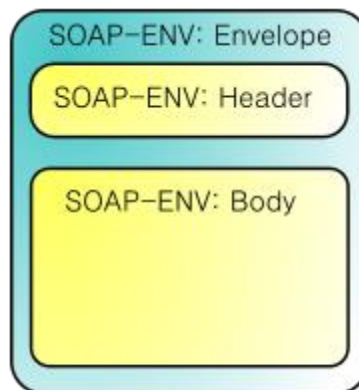
## 1. Introduction

### XML

Extensible Markup Language (XML) is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable. This interface allows businesses and integrators to integrate and retrieve identity data of individuals using XML formatted data over a secure SOAP connection.

### SOAP

Originally defined as Simple Object Access Protocol, is a protocol specification for exchanging structured information in the implementation of web services in computer networks.



For more on SOAP, please visit <http://en.wikipedia.org/wiki/SOAP> or visit <http://www.salesforce.com/us/developer/docs/api/> for a handy developers guide.

This document serves as technical guide to allow 3rd parties to access the pbVerify API Service by means of SOAP web service through a secure HTTPS tunnel.

Note that this is not a free service and all transactions must be considered billable. To activate the service please contact us for more information on 011 516 9400 and ask to speak with a pbVerify consultant.

## 2. Service Overview

### pbVerify KYC API

#### Function: getKYC

This API call gives businesses that are required to FICA their customers, the ability to digitally compare inputted ID and Address data against the SACRRA (South African Credit Risk Reporting Association) database, using a rule set of match criteria to match against the members database. The API will also display the names of the SACRRA member where that address appears. If the address does not appear according the match criteria KYC will fail using an exception rule.

The information required to perform a KYC check.

1. Consumer Details
  - a. ID Number – South African ID Number
  - b. Surname – Consumers Surname
  - c. Enquiry reason – FICA|Tracing
  - d. Address - One-line sentence of address starting with number then street name then suburb then city.
  - e. Date Range - This can be the number 1 to 36 - meaning if 3, 3 months from todays date etc..
  - f. Minimum Match - Number from 1 to 10, this is how many time you want your address to match a SACRRA Address

### 3. RESTful Service

This service supports SOAP via a HTTPS tunnel. The section below covers the connections strings and the methods exposed via the service.

<b>RESTful CONNECTION</b>	<b>POST</b>
<b>URL</b>	<b>https://www.veriid.com/kyc_api/kyc_api.wsdl</b>
<b>Expected Variables</b>	<b>memberkey , password, consumer_details</b>

#### EXPOSED FUNCTIONS

<b>FUNCTION</b>	<b>PARAMETER</b>	<b>TYPE</b>	<b>NOTES</b>	<b>RETURN</b>
<b>pbverify-kyc</b>	<b>memberkey</b>	<b>String</b>	<b>Authentication Username</b>	<b>JSON OUTPUT</b>
	<b>password</b>	<b>String</b>	<b>Authentication Password</b>	
	<b>consumer_details</b>	<b>Array</b>	<b>Array of consumer details</b>	

### 4. SOAP Response

#### getKYC - **Failure**

```
<RESULT>
  <TIMESTAMP>
    <DATE />
    <TIME />
  </TIMESTAMP>
  <ERROR_CODE />
  <ERROR_MESSAGE />
</RESULT>
```

**getKYC – Success**

```

<RESULT>
  <TIMESTAMP>
    <DATE>2017-11-24</DATE>
    <TIME>14:17:07</TIME>
  </TIMESTAMP>
  <FILE-DETAILS>
    <ConsumerDetail>
      <DisplayText>Consumer Detail</DisplayText>
      <Initials>J</Initials>
      <FirstName>JUST</FirstName>
      <Surname>GOOFY</Surname>
      <IDNo>7905015389079</IDNo>
      <BirthDate>1979-05-01</BirthDate>
      <Gender>Male</Gender>
      <TitleDesc>Mister</TitleDesc>
      <MaritalStatusDesc>Married</MaritalStatusDesc>
      <HomeTelephoneNo>0123456789</HomeTelephoneNo>
      <WorkTelephoneNo>0137532507</WorkTelephoneNo>
      <CellularNo>0722256250</CellularNo>
      <EmailAddress>just@goofy.com</EmailAddress>
      <EmployerDetail>PBVerify</EmployerDetail>
    </ConsumerDetail>
    <SuccessfulAddressMatches>
      <item0>
        <Address>TRICELIA ST 18 WEST ACRES NELSPRUIT 1201</Address>
        <SubscriberName>Telkom</SubscriberName>
        <LastUpdatedDate>2017-05-25</LastUpdatedDate>
      </item0>
      <item1>
        <Address>TRICELIA ST 18 WEST ACRES NELSPRUIT 1201</Address>
        <SubscriberName>Telkom</SubscriberName>
        <LastUpdatedDate>2017-05-25</LastUpdatedDate>
      </item1>
    </SuccessfulAddressMatches>
    <MinimumMatchCriteria>
      <matchDateRange>12 months</matchDateRange>
      <matchFound>2</matchFound>
      <matchRequested>2</matchRequested>
      <matchSuccessful>Yes</matchSuccessful>
    </MinimumMatchCriteria>
  </FILE-DETAILS>
</RESULT>

```

## 5. PHP Client Connection

```

<?php
ini_set('display_startup_errors',1);
ini_set('display_errors',1);
error_reporting(-1);

//Create Variables for Script
$Username = "YOUR USERNAME HERE";
$Password = "YOUR PASSWORD HERE";
$MemberKey = "YOUR MEMBERKEY HERE";

$Reference = 'FICA';
$searchDetails = array(
    "idNumber"=>"7905015389079",
    "surname"=>"Goofy",
    "address"=>"18A Tricelia Street West Acres Nelspruit",
    "dateRange"=>"12",
    "minimumMatch"=>"2"
);

//Connect to PBVerify
ini_set('soap.wsdl_cache_enabled',0);
ini_set('soap.wsdl_cache_ttl',0);

try {

    $client = new SoapClient("http://www.veriid.com/kyc_api/kyc_api.wsdl");

    $result = $client->getKYC(array("Username" => $Username, "Password" => $Password, "MemberKey" =>
    $MemberKey, "EnquiryReason" => $Reference, "SearchDetails" => $searchDetails));

    $result = new SimpleXMLElement($result);

    echo '<pre>' . print_r($result, true) . '</pre>';
    exit;
}catch(SoapFault $exception){
    echo $exception;
}
?>

```

## 6. Management Console

<http://www.pbverify.co.za/API-Manager>

- Use Username and Password provided by PBSA to Login
- System can be used to maintain Company Information
- Quick view of last 25 live API calls
- Reporting to view between dates of API calls
- Check status of each transaction with error codes.