







Get your company connected to the Axxess Reseller API

This document will outline and provide you with all the necessary technical information.



Products available via the API

 Mobile Data	 Fibre Combos	 ADSL Data
 ADSL Lines	 Axxess Voice (VoIP)	 Fixed Wireless

The API will allow you to:

- { Create new client profiles and assign available products }
- { Extract currently assigned and unassigned SIMS }
- { Extract a list of services for a specific client }
- { Extract details which include detailed diagnostics such as data session details }

Perform a variety of useful functions

Please Note: The API may be subjected to a hourly limitation on calls. On detecting possible abuse of the API or in the event that a limitation is introduced and this limit is exceeded, a temporary suspension or a limit on accepted requests may be enforced to conserve network resources and to ensure the overall quality of the web service.

1. Authentication.....	4
1.1 getSession.....	4
1.2 checkSession.....	4
2. Clients.....	5
2.1 getProvinces.....	5
2.2 getAllClients.....	5
2.3 getClientById.....	6
2.4 createClient.....	6
3. Services.....	7
3.1 getProducts.....	7
3.2 getServicesByClient.....	8
3.3 getAllUnassignedSimServices.....	8
3.4 getServiceById.....	9
3.5 getServiceSessionDetailsById.....	9
3.6 getPreviousMonthUsageById.....	10
3.7 getRadiusServiceBandwidth.....	10
3.8 getServiceUsageDetailsById.....	11
3.9 getServiceSipDetailsById.....	11
4. Create Service.....	12
4.1 createService.....	12
4.2 ADSL Service.....	12
4.3 Mobile Service.....	13
4.4 DSL Line Service.....	13
4.5 getServiceChangeHistory.....	14
4.6 VoIP Service (New).....	14
4.7 createVoiceFileAttachments.....	15
4.8 VoIP Service (Existing).....	16
4.9 Create Fixed Wireless Service (Cell C).....	16
4.9.1 getFixedWirelessHardwareProducts.....	17
4.9.2 getSmartCombosFromHardwareProduct.....	17
4.9.3 createFixedWirelessComboService.....	18
4.9.4 cancelFixedWirelessService.....	19
4.9.5 getFixedWirelessTopupProducts.....	19
4.9.6 funcFixedWirelessTopup.....	20
4.9.7 getFixedWirelessDataUsage.....	20
4.9.8 funcServiceChanges.....	21

5. Tickets	22
5.1 getDepartments	22
5.2 createTicket	22
5.3 updateTicket	22
5.4 viewTicket	23
6. Functions	24
6.1 funcServiceChanges	24
6.2 funcTopups	24
6.2.1 Mobile	25
6.2.2 ADSL	25
6.3 funcPod	25
6.4 funcPasswordReset	26
6.5 funcSuspend	26
6.6 funcLiftSuspend	26
6.7 funcSendTotalBandwidth	27
6.8 funcExpireService	27
6.9 funcFixLine	27
6.10 funcCheckLine	28
7. Fibre	29
7.1 getAddressTypes	29
7.2 getNetworkProviders	29
7.3 getNetworkProviderProducts	29
7.4 checkFibreAvailability	30
7.5 createFibreComboService	30
7.6 createFibreComboPreOrder	31
8. Fibre Map Render	32
8.1 Fibre Map Render	33
8.2 Fixed Wireless Map Render (RAIN)	34
8.3 Fixed Wireless Map Render (Cell C)	35

Out Parameters for all requests			
intCode	String	JSON string with error break down	http code: http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html
strMessage	String	Extra info on returned data, or error	
strStatus	String	Header status as per httpd code	e.g. OK; Internal Service Error

1. Authentication

Basic auth is required as part of every request. The username and password will be made available on request along with a bootstrap file written in PHP, which has done the client side requests through a Curl object implementation. View examples at the end of this document.

1.1 getSession (Method=GET)

To start using our Reseller API you first have to request a session id. This will allow you to interact with the API for 1 hour, before the session id expires.

Name	Type	Description	Values
In Parameters			
strUserName	String(64)	(*Required) Control Panel Username	FLE123
strPassword	String(64)	(*Required) Control Panel Username	Wer33khh
Out Parameters			
strSessionId	String(36)	This is a generated GUID that allows the reseller to interact with the API	369a1e34-4d6d-11e4-a967-002590383a96

1.2 checkSession (Method=GET)

To check if your session id is still active call this function and it will return if your session has expired.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	(*Required) Session identified	
Out Parameters			
strSessionId	String(36)	This is a generated GUID that allows the reseller to interact with the API	369a1e34-4d6d-11e4-a967-002590383a96

2. Clients

2.1 getProvinces (Method=GET)

This function returns a list of all the provinces on our system. Use this id on creation to set the province of your client.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	(*Required) Session identified	
Out Parameters			
arrProvinces	Array	Array break down listed below	
intProvinceId	Integer		1
strName	String(50)	Name	The display name for the province. Eg. Port Elizabeth

2.2 getAllClients (Method=GET)

This function returns an array with the details of all your clients.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	(*Required) Session identified	
Out Parameters			
arrClients	Array	Array break down listed below	
guidClientId	String(36)		1
strName	String(50)	Name	Combined names for the client. First Name + Middle Name
strLastName	String(50)		
strCell	String(10)	Cellphone number	0812244522
intIdNumber	String(13)	Identification number	
strAddress	String(50)	Address of the client	
strSuburb	String(50)	Suburb of the client	
strCity	String(50)	City of the client	
strEmail	String(50)	Email of the client	
strWorkTel	String(10)	Work telephone number of the client	0413968000
strWorkTel	String(10)	Work telephone number of the client	0413968000
intProvinceId	Integer	The id linked to the province.	
intPostalCode	Integer	Postal Code of the client	

Clients (continues)...

2.3 getClientById (Method=GET)

This function returns client details.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	(*Required) Session identified	
guidClientId	String(36)	(*Required)	
Out Parameters			
arrClients	Array	Array break down listed below	
guidClientId	String(36)		1
strName	String(50)	Name	Combined names for the client. First Name + Middle Name
strLastName	String(50)		
strCell	String(10)	Cellphone number	0812244522
strIdNumber	String(13)	Identification number	
strAddress	String(50)	Address of the client	
strSuburb	String(50)	Suburb of the client	
strCity	String(50)	City of the client	
strEmail	String(50)	Email of the client	
strHomeTel	String(10)	Home telephone number of the client	0413968000
strWorkTel	String(10)	Work telephone number of the client	0413968000
intProvinceId	Integer	The id linked to the province.	
intPostalCode	Integer	Postal Code of the client	

2.4 createClient (Method=PUT)

This function creates a client profile and returns a client identifier to be used for service creation.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	(*Required) Session identifier	
strName	String(50)	(*Required) Name	First name of the client
strLastName	String(50)	(*Required) Last Name	Last name of the client
strCell	String(10)	Cellphone number	0812244522
strIdNumber	String(13)	Identification number	
strCompanyName	String(50)	Company of the client	
strAddress	String(50)	Address of the client	
strSuburb	String(50)	Suburb of the client	
strCity	String(50)	City of the client	

2.4 createClient (continues)...

This function will return a list of all the provinces we have on our system. You can use this id to set the province of your client on creation.

Name	Type	Description	Values
strEmail	String(50)	Email of the client	
strHomeTel	String(10)	Home telephone number of the client	0413968000
strWorkTel	String(10)	Work telephone number of the client	0413968000
intProvinceId	Integer	The id linked to the province.	
intPostalCode	Integer	Postal Code of the client	
Out Parameters			
guidClientId	String(36)	The id that is used to retrieve a client or retrieve services of a client	
strUsername	String (50)	The client code	
strPassword	String (50)	The password	

3. Services

3.1 getProducts (Method=GET)

This function is used to retrieve all the products to be used when creating a service.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	(*Required) Session identified	
Out Parameters			
arrProducts	Array		
guidProductId	String(36)	The product id that will be used to create services	
strName	String(50)	This is the display name	Uncapped 4mb

Services (continues)...

3.2 getServicesByClient (Method=GET)

This function is used to retrieve all the service details for a particular client.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	(*Required) Session identified	
guidClientId	String(36)	(*Required) id of the client that you want to retrieve the services for	
Out Parameters			
arrServices	Array	Each index is comprised of the following:	
guidServiceId	String(36)	GUID to identify the client	
strDescription	String(50)	The description field will be used for a range of service descriptions	If it is an ADSL service it will be used for the username
guidProductId	String(36)	The product id that the service is linked to	
intSuspendReasonId	Integer	The suspend id that the service is on	This can be null if no suspend reason is set
intCurrentQuantity	Integer	The quantity of the service	This will show the amount of gigs for ADSL products
intQuantity	Integer	The default quantity of the service without any top ups	
strDateEnd	String(50)	The date that the service ends	yyyy-mm-dd

3.3 getAllUnassignedSimServices (Method=GET)

This function is used to retrieve all unassigned sim card services.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	(*Required) Session identified	
Out Parameters			
arrServices	Array	Array break down listed below	
guidServiceId	String(36)	GUID to identify the client	
strDescription	String(50)	The description field will be used for a range of service descriptions	If it is an ADSL service it will be used for the username
guidProductId	String(36)	The product id that the service is linked to	
strDateEnd	String(50)	The date that the service ends	yyyy-mm-dd

3.4 getServiceByld (Method=GET)

This function is used to retrieve all details of a particular service.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	(*Required) Session identified	
guidServiceId	String(36)	(*Required) id of the client that you want to retrieve the services for	
Out Parameters			
guidServiceId	String(36)	GUID used to identify the client	
strDescription	String(50)	The description field will be used for a range of service descriptions	If it is an ADSL service it will be used for the username
guidProductId	String(36)	The product id that the service is linked to	
intSuspendReasonId	Integer	The suspend id that the service is on	This can be null if no suspend reason is set
intQuantity	Integer	The quantity of the service	This will show the amount of gigs for ADSL products
intCurrentQuantity	Integer	The default quantity of the service without any top ups	
strDateEnd	String(50)	The date that the service ends	yyyy-mm-dd

3.5 getServiceSessionDetailsByld (Method=GET)

This function is used to retrieve session details of a particular service.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	(*Required) Session identified	
guidServiceId	String(36)	(*Required) id of the client that you want to retrieve the services for	
Out Parameters			
arrSessions			
strUsername	String(50)		
strNasPort	String		
strNASIPAddress	String(15)		
strPhoneNumber	String(10)	The current calling station for the adsl service	
strStartTime	String(50)	The session start time	
strStopTime	String(50)	The session stop time	
strDuration	String		
strTerminateCause	String(50)	The reason the service disconnected	
strFramedIP	String(15)		
intUploadMB	String		
intDownloadMB	String		
intTotalMB	String		

3.6 getPreviousMonthUsageById

This function is used to create a client and will return a client identifier that will be used for creation of services.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	Session identified	
guidServiceId	String(36)	(*Required) id of the client that you want to retrieve the services for	
Out Parameters			
intReturnCode	String	JSON string with error break down	http code: http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html
arrSessions			
strUsername	String(50)	The client code	
strNasPort	String(50)		
strNASIPAddress	String(15)		
strPhoneNumber	String(10)	The current calling station for the adsl service	
strStartTime	String(50)	The session start time	
strStopTime	String(50)	The session stop time	
strDuration	String		
strTerminateCause	String(50)	The reason the service disconnected	
strFramedIP	String(15)		
intUploadMB	String		
intDownloadMB	String		
intTotalMB	String		

3.7 getRadiusServiceBandwidth (Method=GET)

This function takes in a guidServiceId and returns the current available bandwidth for that service.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	(*Required) Session identified	
guidServiceId	String(36)	GUID to identify the service	
Out Parameters			
intPurchased	Integer	Bandwidth Purchased (incl topups/servicechanges)	536870912
intRollover	Integer	If the service has rollover. Else it will be 0	0
intAvailable	Integer	Bandwidth Available. Excluding rollover.	504583630

- All bandwidth amounts are in bytes.
- Uncapped accounts will display 'N/A'

3.8 getServiceUsageDetailsById (Method=GET)

This function is used to retrieve usage of a particular service.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	(*Required) Session identified	
guidServiceId	String(36)	(*Required) id of the client that you want to retrieve the services for	
Out Parameters			
strAccountStartDate			
strDate			
intTotal			
intTotalUp			
intTotalDown			
arrUsage	Array	Array break down per day listed below	
intTotal			
intUp			
intDown			

3.9 getServiceSipDetailsById

These are the details of your Voice Service (number,password)

Name	Type	Description	Values
In Parameters			
guidServiceId	String(36)	Service identifier	
Out Parameters			
arrDetails	Array	Array containing the username and password of the Voice Service	

4. Create Service

4.1 createService (Method=PUT)

This function allows you to create a new service for a client.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	(*Required) Session identified	
guidClientId	String(36)	(*Required) Id of the client that you want to retrieve the services for	
guidProductId	String(36)	(*Required) The productid that the service is linked to	
Out Parameters			
decBalance	Decimal(18,3)	Balance that you have left to purchase services	
arrServices	Array	Array break down listed below	

4.2 ADSL Service (Method=PUT)

These details are required to create an ADSL service.

Name	Type	Description	Values
In Parameters			
intQuantity	Integer	(*Required) The per gig amount for the ADSL service	5 for a 5 gig service
strUsername	String(20)	(*Required) Username for the router	test@axess
strDateStart	String	(*Feature not yet available) Date for when the service will begin and the invoice created	
Out Parameters			
guidServiceId	String(36)	The identifier for the service	These will only be returned on successful service creation.
strUsername	String (50)		
strPassword	String (50)	Password for your router	

Create Service (continues)...

4.3 Mobile Service (Method=PUT)

These details are required to create a mobile service.

Name	Type	Description	Values
In Parameters			
guidMobileServiceId	String(36)	(*Required) The GUID of the sim card for data allocation	
strDateStart	String	(*Feature not yet available) Date for when the service will begin and the invoice created (Feature unavailable presently)	
Out Parameters			
guidServiceId	String(36)	The identifier for the service	This will only be returned on successful service creation.

4.4 DSL Line Service (Method=PUT)

These details are required to create a DSL line service.

Name	Type	Description	Values
In Parameters			
strLineNumber	String(50)	(*Required) The line number of the DSL line that needs to be created	
strOwnerName	String(50)	(*Required) The owner name of the line being created	
strAddress	String(250)	(*Required) Full address of the place where the line is	
strSuburb	String(100)	(*Required) The suburb of the place where the line is	
strCity	String(100)	(*Required) The town or city of the place the line is	
intPostalCode	Integer	(*Required) The postal code of the place the line is	
intIsTransfer	Integer	(*Required) Is it a transfer 0 1	
Out Parameters			
guidServiceId	String(36)	The identifier for the service	This will only be returned on successful service creation.

Create Service (continues)...

4.5 getServiceChangeHistory (Method=GET)

This function takes in a guidServiceId and a date (YYYY-MM) and returns all the servicechange/topup history for the given month.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	(*Required) Session identified	
guidServiceId	String(36)	GUID to identify the service	
strDate	String(36)	Year, Month of history requested	Eg: '2016-04'
Out Parameters			
arrDetails	String(36)	This is a generated GUID that allows the reseller to interact with the api	369a1e34-4d6d-11e4-a967-002590383a96
OriginalProductGuidId	String(36)	Guid to identify start product	
OriginalQuantity	String(36)	Quantity of start service	
ChangeProductGuidId	String(36)	Guid to identify the change service	
ChangeQuantity	String(36)	Quantity of change service	
DateCreated	String(36)		
DateStart	String(36)		
Type	String(36)	To identify what type of change it was	"Topup" "ServiceChange"

4.6 VoIP Service (New)

These are the details that are required for creation of a NEW Voip Service

Name	Type	Description	Values
In Parameters			
guidProductId	String(36)	Product identifier	
strVoiceNumberType	String(36)	Identifier for either a new or ported voip number	'new'
strOwnerName	String(36)	The owner name of the Voip Service	
strAddress	String(36)	The owner's address	
strSuburb	String(36)	The owner's suburb	
strCity	String(36)	The owner's City	
intPostalCode	Int(4)	The owner's postal code	
strEmail	String(36)	The owner's email address	
strCell	String(36)	The owner's cellphone number	
Out Parameters			
guidServiceId	String(36)	The identifier for the service	These will only be returned on successful service creation.

Create Service (continues)...

VoIP Service (Existing)

*When porting a voip number, certain documents are needed.

Before Calling the createService api call, documents first need to be uploaded.

The Documents needed are:

- Copy Of Your ID
- Latest paid invoice of current Service Provider
- Proof of Payment of existing network operators account
- Signed REQUEST FORM

Getting the Request Form.

The Request Form can be obtained by logging into your control panel (<https://rcp.axxess.co.za>) and going to My Account -> Documents . They will be under the Voice Porting Documents tab.

Once the form is filled in, re-upload it and the other documents using the following:

4.7 createVoiceFileAttachments (Method=POST)

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	Product identifier	
strNumber	String(36)	The phone number you are trying to port	
fileIdCopy	file	Copy Of Your ID	
filePayment	file	Proof of Payment of existing network operators account	
fileStatement	file	Latest paid invoice of current Service Provider	
fileSignedRequest	file	Signed REQUEST FORM	
Out Parameters			
intReturnCode	String	JSON string with error break down	http code: http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html

4.8 VoIP Service (Existing)

These are the details that are required for creation of a **NEW** Voip Service

Name	Type	Description	Values
In Parameters			
guidProductId	String(36)	Product identifier	
strVoiceNumberType	String(36)	Identifier for either a new or ported voip number	'existing'
strPortedNumber	String(36)	The number you want to port. (this must match the number submitted in the createVoiceFileAttachments Call)	
strOwnerName	String(36)	The owner name of the Voip Service	
strAddress	String(36)	The owner's address	
strSuburb	String(36)	The owner's suburb	
strCity	String(36)	The owner's City	
intPostalCode	Int(4)	The owner's postal code	
strEmail	String(36)	The owner's email address	
strCell	String(36)	The owner's cellphone number	
Out Parameters			
guidServiceId	String(36)	The identifier for the service	These will only be returned on successful service creation

4.9 Create Fixed Wireless Service (Cell C)

To create a fixed wireless combo service you will need to run `createFixedWirelessComboService`. This function requires a smart combo product (`guidProductId`) which you need to get from the result of running `getSmartCombosFromHardwareProduct`. In order to get a smart combo product from `getSmartCombosFromHardwareProduct` you need to pass in a hardware product (`guidProductId`) which is returned when running `getFixedWirelessHardwareProducts`.

See example below:

```
Run getFixedWirelessHardwareProducts
```

```
Run getSmartCombosFromHardwareProduct using the guidProductId returned from the function above  
(getFixedWirelessHardwareProducts)
```

```
Run createFixedWirelessComboService using guidProductId returned from the function above  
(getSmartCombosFromHardwareProduct)
```


Create Service (continues)...

4.9.1 getFixedWirelessHardwareProducts (Method=GET)

Get Hardware Products for Fixed Wireless (Routers/Sims).

These are the details that are required for getting the hardware products associated with Fixed Wireless.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	Session identifier	
guidNetworkProviderId	String(36)	Network Provider identifier	
Out Parameters			
arrFixedWirelessHardwareProducts	Array	Array breakdown listed below	
guidProductId	String(36)	Product identifier	
strName	String(36)	Hardware product name	

4.9.2 getSmartCombosFromHardwareProduct (Method=GET)

Get all Fixed Wireless Smart Combos associated with a specific hardware product.

These are the details that are required for getting the Fixed Wireless smart combos.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	Session identifier	
guidNetworkProviderId	String(36)	Network Provider identifier	
guidProductId	String(36)	Product identifier	
Out Parameters			
arrNetworkProviderProducts	Array	Array breakdown listed below	
guidProductId	String(36)	Product identifier	
strName	String(36)	Name of Smart Combo Package	
strHardware	String(36)	Name of hardware item associated with Smart Combo	

Create Service (continues)...

4.9.3 createFixedWirelessComboService (Method=POST)

Purchase Fixed Wireless Service.

These are the details that are required for the creation (purchase) of a Fixed Wireless smart combo.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	Session identifier	
guidClientId	String(36)	Client identifier	
guidProductId	String(36)	Product identifier	
guidNetworkProviderId	String(36)	Network Provider identifier	
strFirstName	String(36)	First name of the client	
strLastName	String(36)	Last name of the client	
strCell	String(36)	Cell phone number of the client	
strAddress	String(36)	Street Address of the client	
strSuburb	String(36)	Suburb of the client	
strCity	String(36)	City of the client	
strCode	Int(4)	Postal code of client	
strIdNumber	String(36)	Optional ID number of the client (Required if strPassportNumber is empty)	
strPassportNumber	String(36)	Optional Passport number of the client (Required if strIdNumber is empty)	
strUsername	String(36)	Optional username for data service	
fileCopyOfId	File	Copy of clients ID document	
fileProofOfAddress	File	Copy of proof of address	
Out Parameters			
decBalance	Decimal(18,3)	Credit balance	
arrServices	Array	Array breakdown listed below	
guidServiceId	String(36)	Service identifier of purchased services	
strDescription	String(36)	Description of purchased services (Data service & combo service)	

Create Service (continues)...

4.9.4 cancelFixedWirelessService (Method=PUT)

Cancel Fixed Wireless service or cancel Fixed Wireless order.

These are the details that are required for canceling a fixed wireless service or canceling a Fixed Wireless order.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	Session identifier	
guidNetworkProviderId	String(36)	Network Provider identifier	
guidServiceId	String(36)	Service identifier	
strDate	String(36)	The date that the service needs to end	yyyy-mm-dd
Out Parameters			
intReturnCode	String	JSON string with error breakdown	http code: http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html

4.9.5 getFixedWirelessTopupProducts (Method=GET)

Get the Top-up products that you can use to top-up your Fixed Wireless service with.

These are the details that are required for getting fixed wireless top-up products.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	Session identifier	
guidNetworkProviderId	String(36)	Network Provider identifier	
Out Parameters			
arrProducts	Array	Array breakdown listed below	
guidProductId	String(36)	Product identifier	
strName	String(36)	Name of top-up product	

Create Service (continues)...

4.9.6 funcFixedWirelessTopup (Method=PUT)

Top-up your Fixed Wireless data service.

These are the details that are required for topping up a Fixed Wireless service.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	Session identifier	
guidServiceId	String(36)	Service identifier	
guidProductId	String(36)	Product identifier of top-up product	
Out Parameters			
intReturnCode	String	JSON string with error breakdown	http code: http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html

4.9.7 getFixedWirelessDataUsage (Method=GET)

Get usage for the last 7 days (including today), current month and the last 2 months.

These are the details that are required for retrieving the data usage.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	Session identifier	
guidNetworkProviderId	String(36)	Network Provider identifier	
guidServiceId	String(36)	Session identifier	
Out Parameters			
arrUsage	Array	Array breakdown listed below	
decMonth1	Decimal(18,3)	Current month usage (GB)	55.32
decMonth2	Decimal(18,3)	Previous month usage (GB)	55.32
decMonth3	Decimal(18,3)	Month before last usage (GB)	55.32
declast7Days	Decimal(18,3)	Usage for the last 7 days (GB)	55.32

Create Service (continues)...

4.9.8 funcServiceChanges (Method=PUT)

Service change your existing Fixed Wireless smart combo to a different smart combo package.

These are the details that are required for actioning the service change.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	Session identifier	
guidServiceId	String(36)	Service identifier	
guidProductId	String(36)	Product identifier. The product you want to change to	
intQuantity	Int(1)		1
strDateStart	String(36)	The date that the change needs to take place	yyyy-mm-dd
Out Parameters			
intReturnCode	String	JSON string with error breakdown	http code: http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html

5. Tickets

5.1 getDepartments (Method=GET)

This function is used for retrieving all the departments that will be used when creating a ticket.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	(*Required) Session identified	
Out Parameters			
arrDepartments	Array		
guidDepartmentId	String(36)	The department id that will be used to create tickets	
strName	String(50)	This is the display name	Shipping

5.2 createTicket (Method=GET)

This function is used for retrieving all the departments that will be used when creating a ticket.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	(*Required) Session identified	
guidDepartmentId	String(36)	Department related to ticket query	
strSubject	String(255)	Your ticket subject	
strMessage	String(255)	The body message of your ticket	
Out Parameters			
arrDetails	String(36)	The id of the ticket that was just created	

5.3 updateTicket (Method=GET)

This function is used for retrieving all the departments that will be used when creating a ticket.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	(*Required) Session identified	
strTicketId	String(36)	The TicketId Related to the ticket you updating.	
strMessage	String(255)	The body message of your ticket update	
Out Parameters			
intReturnCode	String	String JSON with error breakdown	http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html

Tickets (continues)...

5.4 viewTicket (Method=GET)

This function is used for retrieving all the departments that will be used when creating a ticket.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	(*Required) Session identified	
strTicketId	String(36)	The TicketId Related to the ticket you viewing.	
Out Parameters			
arrDetails	Array	Array of details pertaining to the ticket.	http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html
Subject	String(36)	Ticket Subject	
DateCreated	String(36)	Date Created of the ticket	
guidDepartmentId	String(36)	Guid to identify the department	
Details(array)	Array	Array of ticket messages	
Message	String(255)	Ticket message	
Operator	String(36)	Author of ticket message	
DateCreated	String(36)	Date created of the ticket messages	

6. Functions

6.1 funcServiceChanges (Method=PUT)

This function allows you to change existing services.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	(*Required) Session identified	
guidServiceId	String(36)	(*Required) The service id for the service that the change is for	
guidProductId	String(36)	(*Required) The product id that the service is changing to	
intQuantity	Integer	This is optional field that will be used only if the product is a unit product	Capped 3 gig to a 6 gig the quantity will be 6. Uncapped services and lines will not need this
strDateStart	String(50)	(*Required) The start date of the service change. If you set this to any date later then the current day it will start on the first of the next month	

6.2 funcTopups (Method=PUT)

This allow for creation topups.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	(*Required) Session identified	
guidServiceId	Integer	(*Required) id of the client that you want to retrieve the services for	
guidProductId	String(36)	(*Required) The product id that the service is changing to	
intQuantity	Integer	(*Required) This function only applies to unit services so this field must be populated	Capped 3 gig to a 6 gig the quantity will be 3.
Out Parameters			
intReturnCode	String	JSON string with error break down	http code: http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html

Functions (continues)...

6.2.1 Mobile

- guidProductId

- Topup via a guidProductId:

<u>guidProductId:</u>	<u>Name</u>
009bfff3-cc80-11e4-b901-000c2994d8d2	Mobile 500MB
009c0062-cc80-11e4-b901-000c2994d8d2	Mobile 1GB
009c00d1-cc80-11e4-b901-000c2994d8d2	Mobile 2GB
288f5951-16b0-11e6-83e7-000c2994d8d2	Mobile 3GB
009c0142-cc80-11e4-b901-000c2994d8d2	Mobile 4GB
288f59df-16b0-11e6-83e7-000c2994d8d2	Mobile 5GB
009c01b1-cc80-11e4-b901-000c2994d8d2	Mobile 6GB
009c021f-cc80-11e4-b901-000c2994d8d2	Mobile 8GB
009c028e-cc80-11e4-b901-000c2994d8d2	Mobile 10GB
009c02fe-cc80-11e4-b901-000c2994d8d2	Mobile 12GB
009c03db-cc80-11e4-b901-000c2994d8d2	Mobile 16GB
009c0447-cc80-11e4-b901-000c2994d8d2	Mobile 20GB
009c04b5-cc80-11e4-b901-000c2994d8d2	Mobile 24GB

- intQuantity = 1;

6.2.2 ADSL

- guidProductId

- guidProductId of the current service.

- intQuantity

- The Amount of gigs you want to topup by.

6.3 funcPod (Method=GET)

This function sends a packet of disconnect on an ADSL service.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	(*Required) Session identified	
guidServiceId	String(36)	(*Required) id of the service that you want to POD	

Functions (continues)...

6.4 funcPasswordReset (Method=GET)

This function will set a service to either expire now, or for the end of the month.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	(*Required) Session identified	
guidServiceId	String(36)	(*Required) id of the service that you want to POD	
strPassword	String(10)	(* Optional) Password you want to use.	
Out Parameters			
strPassword	String(50)	New reset password	

6.5 funcSuspend (Method=GET)

This function suspends a service.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	(*Required) Session identified	
guidServiceId	String(36)	(*Required) id of the service that you want to suspend	

6.6 funcLiftSuspend

This function will lift the suspended status whatever service identifier is passed in. (only for services that have used the 'funcSuspend')

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	(*Required) Session identified	
guidServiceId	String(36)	(*Required) id of the service that you want to suspend	
Out Parameters			
intReturnCode	String	JSON string with error break down	http code: http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html

Functions (continues)...

6.7 funcSendTotalBandwidth (Method=POST)

This function sends total usage via email.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	(*Required) Session identified	
intDayStart	Integer	Starting day of the month. If no start day has been set then it will default to the 1st	
intDayEnd	Integer	End day of the month. If no end day has been set then it will default to the current day	

6.8 funcExpireService

This function will set a service to either expire now, or for the end of the month.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	(*Required) Session identified	
guidServiceId	String(36)	(*Required) id of the service that you want to suspend	
strDate	String(36)	(*Required) The date you want the expiry to run on. (only valid for todays date or the last day of the months date)	Format(Y-m-d) eg. '2015-09-30'
Out Parameters			
intReturnCode	String	JSON string with error break down	http code: http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html

6.9 funcFixLine (Method=GET)

This function runs a Uniweb port recreate on a DSL line service.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	(*Required) Session identified	
guidServiceId	String(36)	(*Required) id of the service that you want to run	

Functions (continues)...

6.10 funcCheckLine (Method=GET)

This function runs an interrogation on a DSL line service.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	(*Required) Session identified	
guidServiceId	String(36)	(*Required) id of the service that you want to check the line for	
Out Parameters			
strDownloadCurrentSpeed	String(50)	Shows the current speed in kbps	8189kbps
strDownloadInterference	String(50)	Shows current interference in dB	10dB
strDownloadNoiseRatio	String(50)	Shows current noise ratio in dB	12.2dB
strUploadCurrentSpeed	String(50)	Shows the current speed in kbps	8189kbps
strUploadInterference	String(50)	Shows current interference in dB	10dB
strUploadNoiseRatio	String(50)	Shows current noise ratio in dB	12.2dB

Example code

Basic auth is required as part of every request. The username and password will be made available on request along with a bootstrap file written in PHP which has done the client side requests through a Curl object implementation. The code is documented with both required inputs as well as expected output for example:

```
/**
 *
 *
 * @param array $d array("strSessionId", "guidServiceId")
 *
 * @return object json encoded with:
 * public 'intCode' => int 200
 * public 'strStatus' => string 'OK' (length=2)
 * public 'strMessage' => null
 * public 'strDownloadCurrentSpeed' => string '10239 kbps'
 * public 'strDownloadInterference' => string '12.2 dB'
 * public 'strDownloadNoiseRatio' => string '12.7 dB'
 * public 'strUploadCurrentSpeed' => string '1021 kbps'
 * public 'strUploadInterference' => string '16.9 dB'
 * public 'strUploadNoiseRatio' => string '12.3 dB'
 *
 */
case "funcCheckLine":
{
$url = "https://apitest.axxess.co.za/" . "calls/rsapi/funcCheckLine.json";
$curl->setBasicAuthentication($Username, $Password);
$curl->setOpt(CURLOPT_SSL_VERIFYPEER, false);
$curl->setOpt(CURLOPT_SSL_VERIFYHOST, 2);
$curl->get($url, $d);
break;
}
```

7. Fibre

7.1 getAddressTypes

This function will return a list of all the address types used to place a fibre order.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	Session identified	
Out Parameters			
arrAddressTypes	Array		1

7.2 getNetworkProviders

This function will return a list of all the network providers.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	Session identified	
Out Parameters			
arrNetworkProviders	Array	Array break down listed below	1
guidNetworkProviderId	String(36)	Network provider identifier	
strName	String(36)	Network provider identifier	

7.3 getNetworkProviderProducts

This function will return the product identifiers for a specific network provider.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	(*Required) Session identified	
guidNetworkProviderId	String(36)	Network provider identifier	
Out Parameters			
arrNetworkProviderProducts	Array	Array break down listed below	1
guidProductId	String(36)	Product Identifier	
strName	String(36)	Product Name	

Fibre (continues)...

7.4 checkFibreAvailability

This checks to see whether given coordinates are capable of receiving fibre.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	Session identified	
strLongitude	String(36)	Longitude	25.5868448999999
strLatitude	String(36)	Latitude	-33.9549916
strAddress	Text	Physical Address	185 Cape Road, Port Elizabeth
Out Parameters			
arrAvailableProvidersGuids	Array	Array break down listed below	1
guidNetworkProviderId	String(36)	Network provider identifier	
intPreOrder	Integer	If the location is pre-order	

7.5 createFibreComboService

Create a fibre service.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	Session identified	
guidClientId	String(36)	Id of the client you want to retrieve the services for	
guidProductId	String(36)	The product id that the service is linked to	
guidNetworkProviderId	String(36)	Network provider identifier	
strOwner	String(36)	Fibre Line Owner Name	
strCell	String(36)	Fibre Line Owner Cell	
strAddress	String(36)	Fibre Line Owner Address	
strSuburb	String(36)	Fibre Line Owner Suburb	
strCity	String(36)	Fibre Line Owner City	
strCode	String(36)	Fibre Line Owner Postal Code	
strLatLong	String(36)	Fibre Line Installation coordinates	
strAddressType	String(36)	Retrieved from the 'getAddressType' function	
strBuildingId	String(36)	Fibre Line Installation Building Number	
strFloorId	String(36)	Fibre Line Installation Floor	
strUnitNumber	String(36)	Fibre Line Installation Unit Number	
strBlockName	String(36)	Fibre Line Installation Block Name	
Out Parameters			
arrServices	Array	Array break down listed below	1
guidServiceId	String(36)	Identifies for the combo you just purchased	
decBalance	String(36)	Credit Balance.	

Fibre (continues)...

7.6 createFibreComboPreOrder

Create a pre-order fibre service.

Name	Type	Description	Values
In Parameters			
strSessionId	String(36)	Session identified	
guidClientId	String(36)	Id of the client you want to retrieve the services for	
guidProductId	String(36)	The product id that the service is linked to	
guidNetworkProviderId	String(36)	Network provider identifier	
strAddress	String(36)	Fibre Line Owner Address	
strSuburb	String(36)	Fibre Line Owner Suburb	
strCity	String(36)	Fibre Line Owner City	
strCode	String(36)	Fibre Line Owner Postal Code	
strLatLong	String(36)	Fibre Line Installation coordinates	
strAddressType	String(36)	Retrieved from the 'getAddressType' function	
strBuildingId	String(36)	Fibre Line Installation Building Number	
strFloorId	String(36)	Fibre Line Installation Floor	
strUnitNumber	String(36)	Fibre Line Installation Unit Number	
strBlockName	String(36)	Fibre Line Installation Block Name	
Out Parameters			
intReturnCode	String	JSON string with error break down	http code: http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html

8. Map Render Requirements

To use our customised Google Maps, you must first register your app project on the Google Cloud Platform Console and get a Google API key.

Start here:

<https://developers.google.com/maps/documentation/javascript/get-api-key>

Please ensure that the following API's are enabled in the Google Cloud Platform Console:

- Maps JavaScript API
- Geocoding API
- Places API

To view your list of enabled APIs:

- Go to the Google Cloud Platform Console.
- Click the Select a project button, then select the same project you set up for the Maps JavaScript API and click Open.
- From the list of APIs on the Dashboard, look for the API.
- If you see the API in the list, you're all set. If the API is not listed, enable it:
 - At the top of the page, select ENABLE API to display the Library tab. Alternatively, from the left side menu, select Library.
 - Search for the API, then select it from the results list.
 - Select ENABLE. When the process finishes, the API will appear in the list of APIs on the Dashboard.

8.1 Fibre Map Render

```
<?php
$mapApiKey = 'YOUR_GOOGLE_MAPS_APIKEY';
$mapWidth = '70%';
$mapHeight = '500px';
?>

<html>
<head></head>
<body>
<form method="post">

    <script type="text/javascript" id="fibrescript">
        (function() {
            var ax = document.createElement('script');
            ax.id = 'mainscript';
            ax.type = 'text/javascript';
            ax.async = true;
            ax.src = 'https://rcp.axxess.co.za/public/js/fibremapJs.php?key=<?php echo $mapApiKey;
                ?>&width=<?php echo $mapWidth; ?>&height=<?php echo $mapHeight; ?>';
            var s = document.getElementsByTagName('script')[0];
            s.parentNode.insertBefore(ax, s);
        })();
    </script>

    <input type="submit" value="submit" />
</form>
</body>
</html>
```

Instructions:

On \$_POST, you will get the following:

```
'address-input' => string 'ADDRESS_OF_MAP_MARKER'
'latlong-input' => string 'LAT_LONG_OF_MAP_MARKER'
```

With these values, you can call the checkFibreAvailability() API call.

8.2 Fixed Wireless Map Render (RAIN)

```
<?php
$mapApiKey = 'YOUR_GOOGLE_MAPS_APIKEY';
$mapWidth = '70%';
$mapHeight = '500px';
?>

<html>
<head></head>
<body>
<form method="post">

    <script type="text/javascript" id="fixedwirelessscript">
        (function() {
            var ax = document.createElement('script');
            ax.id = 'mainscript';
            ax.type = 'text/javascript';
            ax.async = true;
            ax.src = 'https://rcp.axxess.co.za/public/js/fixedwirelessmapJs.php?key=<?php echo $mapApiKey;
                ?>&width=<?php echo $mapWidth; ?>&height=<?php echo $mapHeight; ?>';
            var s = document.getElementsByTagName('script')[0];
            s.parentNode.insertBefore(ax, s);
        })();
    </script>

    <input type="submit" value="submit" />
</form>
</body>
</html>
```

Instructions:

On \$_POST, you will get the following:

```
'address-input' => string 'ADDRESS_OF_MAP_MARKER'
'latlong-input' => string 'LAT_LONG_OF_MAP_MARKER'
```

With these values, you can call the checkFibreAvailability() API call.

8.3 Fixed Wireless Map Render (Cell C)

```
<?php
$mapApiKey = 'YOUR_GOOGLE_MAPS_APIKEY';
$mapWidth = '70%';
$mapHeight = '500px';
?>

<html>
<head></head>
<body>
<form method="post">

    <script type="text/javascript" id="fixedwirelesscellscript">
        (function() {
            var ax = document.createElement('script');
            ax.id = 'mainscript';
            ax.type = 'text/javascript';
            ax.async = true;
            ax.src = 'https://rcp.axxess.co.za/public/js/fixedwirelesscellcmapJs.php?key=?php echo $mapApiKey;
                ?>&width=?php echo $mapWidth; ?>&height=?php echo $mapHeight; ?>';
            var s = document.getElementsByTagName('script')[0];
            s.parentNode.insertBefore(ax, s);
        })();
    </script>

    <input type="submit" value="submit" />
</form>
</body>
</html>
```

Instructions:

On \$_POST, you will get the following:

```
'address-input' => string 'ADDRESS_OF_MAP_MARKER'
'latlong-input' => string 'LAT_LONG_OF_MAP_MARKER'
```

With these values, you can call the checkFibreAvailability() API call.