



AYYEKA

Ayyeka Web API

v2.1

Contents

- 1 Introduction** **3**

- 2 System Overview** **5**
 - 2.1 Ayyeka Data Model 5
 - 2.2 Device Settings and Commands 6

- 3 Namespace Index** **7**
 - 3.1 Packages 7

- 4 Hierarchical Index** **9**
 - 4.1 Class Hierarchy 9

- 5 Class Index** **11**
 - 5.1 Class List 11

- 6 Namespace Documentation** **13**
 - 6.1 Package Ayyeka 13
 - 6.2 Package Ayyeka.Web 13
 - 6.3 Package Ayyeka.Web.Api 13
 - 6.3.1 Enumeration Type Documentation 14
 - 6.3.1.1 ReturnCodes 14

- 7 Class Documentation** **15**
 - 7.1 Ayyeka.Web.Api.API Class Reference 15
 - 7.1.1 Detailed Description 16
 - 7.1.2 Member Function Documentation 16
 - 7.1.2.1 CreateOnDeviceStreamRule 16
 - 7.1.2.2 GetDeviceCommandStatus 17
 - 7.1.2.3 GetDeviceReportingTypes 17
 - 7.1.2.4 GetOnDeviceStreamRule 17
 - 7.1.2.5 GetOnDeviceStreamRuleCreationStatus 17
 - 7.1.2.6 GetSamplesByStreams 18
 - 7.1.2.7 GetSites 18
 - 7.1.2.8 GetStreamsBySite 18

7.1.2.9	SendRebootCommand	18
7.1.2.10	SendReportingIntervalCommand	19
7.1.2.11	SendSetSettingCommand	19
7.2	Ayyeka.Web.Api.Authentication Class Reference	19
7.2.1	Detailed Description	20
7.2.2	Member Enumeration Documentation	20
7.2.2.1	AuthenticationSvcRetCodes	20
7.2.3	Member Function Documentation	20
7.2.3.1	Login	20
7.3	Ayyeka.Web.Api.Authentication.LoginReq Class Reference	21
7.3.1	Detailed Description	21
7.4	Ayyeka.Web.Api.Authentication.LoginRes Class Reference	21
7.4.1	Detailed Description	21
7.5	Ayyeka.Web.Api.ServiceArrayResponse< T > Class Template Reference	21
7.5.1	Detailed Description	22
7.6	Ayyeka.Web.Api.ServiceScalarResponse< T > Class Template Reference	22
Index		23

Chapter 1

Introduction

Ayyeka provides programmatic access to your organization's information using a simple and secure SOAP-based interface. To use this document, you should have a basic familiarity with software development, Web Services and Ayyeka's UI.

The Ayyeka API provides the following:

- Management of logical entities such as sites and streams
- Retrieval of the last measurements
- Retrieval of historical measurements
- Remote control of field-deployed Wavelet devices

Chapter 2

System Overview

2.1 Ayyeka Data Model

Sensor measurements are stored in Ayyeka's database in a hierarchical structure.

- Sites - The entity that represents logical data acquisition locations and that can be assembled from one or more devices.
- Devices - The entity that represents the Wavelet installed in the field. Each device belongs to a specific Site.
- Streams - The entity that describes the type of measurement that is collected by the devices (sensor data). Each Stream belongs to a specific Site.
- Samples - The entity that describes measurements. Each sample belongs to a specific Stream.

Site, Stream, Sample and Device have a unique ID that identify them and are used across the API methods.

While 'Devices' represent the physical hardware in the field, 'Streams' represent logical time-series data. This enables the maintenance of a single data stream in the event of hardware replacement.

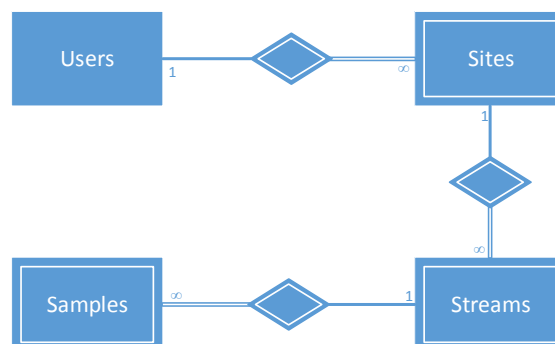


Figure 2.1: System - Block Diagram.

2.2 Device Settings and Commands

Device configuration consists of multiple settings that define device behavior. The user can change device behavior by submitting device commands to the server via the API.

Please note that the server cannot initiate transmission to send prepared device commands to the field-deployed device. All transmission sessions are initiated by the field-deployed device. Commands in the server queue are submitted in a first in first out (FIFO) order once the device initiates transmission with the server. At the end of each device transmission, the device will receive a bulk of commands to process, assuming commands have been set in the queue.

In order to receive the current device configuration, call API's `GetDeviceCurrentSettings` method (see 7.1 for details). User can change the setting by submitting a command via API to change a setting by calling API's `SendSetSettingCommand` method (see 7.1 for details).

In order to track the command status you should call the API method `GetDeviceCommandStatus`.

Each command is tracked as with one of the following status indicators:

- PENDING - The command is waiting for the next time the device will transmit.
- SUCCESS - The command was sent to the device, and the device configuration was successful.
- FAIL - The command was sent to the device, but the device configuration was unsuccessful.
- PARTIAL FAIL - The command was sent, but only part of the device configuration was successful.
- CANCEL - The command was cancelled before it was sent to the device.
- UNKNOWN - An internal error occurred on the server while sending the command.

For the purpose of simplification, Ayyeka models out common user setting changes as follows:

- Reporting Priorities Types - Send and Get the reporting interval types of the device (see the API calls `SendReportingIntervalCommand` 7.1 , `GetDeviceReportingTypes` 7.1 for details)
- On Device Stream Rule - Create and Get the stream on device rule, which specifies the stream's thresholds and actions, i.e. changing reporting interval or mail/SMS notification (see the API calls `CreateOnDeviceStreamRule` 7.1, `GetOnDeviceStreamRule` 7.1, `GetOnDeviceStreamRuleCreationStatus` 7.1 for details).

Moreover, the user can request a device reboot by calling the API's `SendRebootCommand` command (see `SendRebootCommand` 7.1 for details).

Chapter 3

Namespace Index

3.1 Packages

Here are the packages with brief descriptions (if available):

Ayyeka	13
Ayyeka.Web	13
Ayyeka.Web.Api	13

Chapter 4

Hierarchical Index

4.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Ayyeka.Web.Api.Authentication.LoginReq	21
Ayyeka.Web.Api.Authentication.LoginRes	21
Ayyeka.Web.Api.ServiceArrayResponse< T >	21
Ayyeka.Web.Api.ServiceScalarResponse< T >	22
WebService	
Ayyeka.Web.Api.API	15
Ayyeka.Web.Api.Authentication	19

Chapter 5

Class Index

5.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

- Ayyeka.Web.Api.API**
 - Web** (p. 13) Service for exploring **Ayyeka** (p. 13) sites and streams, retrieving data samples, and managing devices. All of the web methods in this class should include the security token user's, an authentication string retrieved at `https://soapapi.ayyeka.com/Authentication.asmx` and they are expect to receive the next HTTP header: 15
- Ayyeka.Web.Api.Authentication**
 - Web** (p. 13) Service for connecting to **Ayyeka** (p. 13)'s services and data. Each **Ayyeka** (p. 13)'s service expects to receive the **Ayyeka-Auth-Token: <token-string>** HTTP header, where **<token-string>** is a long string received in **Login(LoginReq req)** (p. 20) web method. This token is temporary: it is created for a client with a timeout that expires if it is not in use by the client to initiate any new requests for any of **Ayyeka** (p. 13)'s web services for some time. After the token expires, the client can request a new token from **Login(LoginReq req)** (p. 20). 19
- Ayyeka.Web.Api.Authentication.LoginReq**
 - Login Request object. Used in **Login(LoginReq req)** (p. 20). Contains the Username and Password properties, both of string type. 21
- Ayyeka.Web.Api.Authentication.LoginRes**
 - Login response object. Used in **Login(LoginReq req)** (p. 20). This object has the properties listed below: 21
- Ayyeka.Web.Api.ServiceArrayResponse< T >**
 - Service Response is a wrapper class. It wraps a generic array of values with the relevant metadata. The metadata is the response return codes and error messages. 21
- Ayyeka.Web.Api.ServiceScalarResponse< T >** 22

Chapter 6

Namespace Documentation

6.1 Package Ayyeka

Namespaces

- package **Web**

6.2 Package Ayyeka.Web

Namespaces

- package **Api**

6.3 Package Ayyeka.Web.Api

Classes

- class **API**

Web (p. 13) Service for exploring **Ayyeka** (p. 13) sites and streams and retrieving data samples and manage. All of the web methods in this class are bound to the user's authentication string, which can be received at `https://soapapi.ayyeka.com/Authentication.asmx` and they expect to receive the next HTTP header:

- class **Authentication**

Web (p. 13) Service for connecting to **Ayyeka** (p. 13)'s services and data. Each **Ayyeka** (p. 13)'s service expects to receive the Ayyeka-Auth-Token: <token-string> HTTP header, where <token-string> is a long string received in **Login(LoginReq req)** (p. 20) web method. This token is temporary: it is created for a client with a timeout that expires if the client does not initiate any new requests for any of **Ayyeka** (p. 13)'s web services for some time. After the token expires, the client can request a new token from **Login(LoginReq req)** (p. 20).

- class **ServiceArrayResponse**

Service Response is a wrapper class. It wraps generic array of values with the relevant metadata. The metadata is the response return codes and error messages.

- class **ServiceScalarResponse**

Enumerations

- enum **ReturnCodes** { **OK** = 0, **Error**, **UnauthorizedAccess** }

Service Return Codes. These codes define the response state - OK, Error and UnauthorizedAccess

6.3.1 Enumeration Type Documentation

6.3.1.1 enum `Ayyeka.Web.Api.ReturnCodes`

Service Return Codes. These codes define the response state - OK, Error and UnauthorizedAccess

Chapter 7

Class Documentation

7.1 Ayyeka.Web.Api.API Class Reference

Web (p.13) An authentication string is required for **Ayyeka** (p.13) web methods to explore sites and streams, retrieve data samples, and manage devices. The web methods in this class are bound to the user's, which can be received at <https://soapapi.ayyeka.com/Authentication.asmx> and they are expect to receive the next HTTP header:

Inherits WebService.

Public Member Functions

- **ServiceArrayResponse**< Containers.Site > **GetSites** ()
The function receives the number of sites to pull from the database and returns the IDs relevant to the user sites, their display names, and a RetCode that indicates the state of the request.
- **ServiceArrayResponse**< Containers.Device > **GetDevicesBySite** (ulong SiteId)
- **ServiceArrayResponse**< Containers.Stream > **GetStreamsBySite** (ulong SiteId)
The function receives a site ID and the number of streams to pull from the DB, and it returns the site IDs of the relevant streams, their display names, and a RetCode that indicates the state of the request.
- **ServiceArrayResponse**< Containers.Sample > **GetSamplesByStreams** (int NumOfRecords, Samples↔ Selector[] SamplesSelector)
The function receives a number of streams to pull from the DB, and an array of NewSamplesSelector objects. Each NewSamplesSelector holds the requested stream ID, and an ID of the most recently requested sample. The function returns a list of samples, starting from the ID of the last requested sample, and relevant metadata.
- **ServiceArrayResponse**< DeviceReportingType > **GetDeviceReportingTypes** (ulong DeviceId)
Use this method to receive all reporting interval types defined for the device. The device supports multiple types of reporting intervals, but only one is used by the device at any given moment. By default, the device behavior is to use the interval defined in Reporting Type Id - 0. However, if auto connect defined, then the device will use the the interval defined in Reporting Type Id - 1. If OnDeviceStreamRule defined then the device will use the interval type that specified in the current level that stream samples are in.
- **ServiceScalarResponse**< uint > **SendRebootCommand** (ulong DeviceId)
Use this method to submit the command for Device to reboot.
- **ServiceScalarResponse**< uint > **SendReportingIntervalCommand** (ulong DeviceId, uint Typeld, uint ReportingInterval)
Use this method to submit Device Command to change the interval defined in specified Reporting Type. Read more about Reporting Types in GetDeviceReportingTypes method summary.
- **ServiceArrayResponse**< Containers.DeviceSetting > **GetDeviceCurrentSettings** (ulong DeviceId)
Device configuration contains multiple settings that affect the behavior of the device. Each setting belongs to a specific Module in the device firmware. There are setting that repeats for diffenet channels in specified Module. Therefore, each setting defined by tuple ModuleId - Module Identification in firmware i.e. GPIO, MODBUS etc SettingId - Setting Identification in the module Index - An index of setting If this setting repeated for each channel, else index is 0 Variant - Type of the setting i.e. u8, u16, s32, string

Parameters

DeviceId	Device Id
----------	-----------

Returns

- **ServiceScalarResponse**< Containers.DeviceCommand > **GetDeviceCommandStatus** (ulong DeviceId, uint CommandId)
Use this method if you want to get the status of submitted Set Setting Command.
- **ServiceScalarResponse**< uint > **SendSetSettingCommand** (ulong DeviceId, string ModuleId, string SettingId, int Index, string Variant, string Value)
Use this method to submit DeviceCommand to set new value for one of the Device settings. See GetDeviceCurrentSettings summary for details.
- **ServiceScalarResponse**< Containers.OnDeviceScalarStreamRule > **GetOnDeviceStreamRule** (ulong StreamId)
Use this method to receive the OnDeviceStreamRule if it was defined before. OnDeviceStreamRule is a rule that specify the actions that device should take when the stream samples enters a specific level(threshold). Action can be one ore more of the following: Set Current Reporting Type, Set new Sampling Interval, Notify a Group, Stop/Start sampling other Stream on Device.
- **ServiceScalarResponse**< bool > **CreateOnDeviceStreamRule** (ulong StreamId, Ayyeka.Web.Api.Containers.OnDeviceScalarStreamRule rule)
Use this method to create a CreateOnDeviceStreamRule , rule that defines the behavior of a device for a stream samples. OnDeviceStreamRule is a rule that specify the actions that device should take when the stream samples enters a specific level(threshold). Action can be one ore more of the following: Set Device Current Reporting Type, Set new Sampling Interval, Notify a Group via SMS/MAIL, Stop/Start sampling other Stream on Device.
- **ServiceScalarResponse**< Ayyeka.Web.Api.Containers.DeviceCommandStatus > **GetOnDeviceStreamRuleCreationStatus** (ulong StreamId)
Use this method to get status of creation of OnDeviceStreamRule of specific Stream

7.1.1 Detailed Description

Web (p.13) Service for exploring **Ayyeka** (p.13) sites and streams and retrieving data samples and manage. All of the web methods in this class are bound to the user's authentication string that can be received at <https://soapapi.ayyeka.com/Authentication.asmx> and they are expect to receive the next HTTP header:

Ayyeka-Auth-Token: <token-string>

Where <token-string> is some long string that is received in Authentication.asmx service. Without it, any web method in this service will return an error code.

7.1.2 Member Function Documentation

7.1.2.1 ServiceScalarResponse<bool> Ayyeka.Web.Api.API.CreateOnDeviceStreamRule (ulong StreamId, Ayyeka.Web.Api.Containers.OnDeviceScalarStreamRule rule)

User this method to create a CreateOnDeviceStreamRule , rule that defines the behavior of a device for a stream samples. OnDeviceStreamRule is a rule that specify the actions that device should take when the stream samples enters a specific level(threshold). Action can be one ore more of the following: Set Device Current Reporting Type, Set new Sampling Interval, Notify a Group via SMS/MAIL, Stop/Start sampling other Stream on Device.

Parameters

StreamId

<i>rule</i>	
-------------	--

Returns

7.1.2.2 ServiceScalarResponse<Containers.DeviceCommand> Ayyeka.Web.Api.API.GetDeviceCommandStatus (ulong *DeviceId*, uint *CommandId*)

Use this method to retrieve the status of submitted Set Setting Commands.

Parameters

<i>DeviceId</i>	Device Id
<i>CommandId</i>	Device Command Id

Returns

7.1.2.3 ServiceArrayResponse<DeviceReportingType> Ayyeka.Web.Api.API.GetDeviceReportingTypes (ulong *DeviceId*)

Use this method to receive all reporting interval types defined for the device. The device supports multiple types of reporting intervals but only one is used by device in any moment. By Default, behavior of the Device is to use the interval that defined in Reporting Type Id - 0. However, if auto connect is defined, then the device will use the interval defined in Reporting Type Id - 1. If OnDeviceStreamRule is defined, then the device will use the interval type specified by the current level of stream samples.

Parameters

<i>DeviceId</i>	Device Id
-----------------	-----------

Returns

7.1.2.4 ServiceScalarResponse<Containers.OnDeviceScalarStreamRule> Ayyeka.Web.Api.API.GetOnDeviceStreamRule (ulong *StreamId*)

Use this method to receive the OnDeviceStreamRule if it was defined before. OnDeviceStreamRule is a rule that specify the actions that device should take when the stream samples enters a specific level(threshold). Action can be one ore more of the following: Set Current Reporting Type, Set new Sampling Interval, Notify a Group, Stop/Start sampling other Stream on Device.

Parameters

<i>StreamId</i>	
-----------------	--

Returns

7.1.2.5 ServiceScalarResponse<Ayyeka.Web.Api.Containers.DeviceCommandStatus> Ayyeka.Web.Api.API.GetOnDeviceStreamRuleCreationStatus (ulong *StreamId*)

Use this method to get status of creation of OnDeviceStreamRule of specific Stream

Parameters

<i>StreamId</i>	Stream Id
-----------------	-----------

Returns

7.1.2.6 ServiceArrayResponse<Containers.Sample> Ayyeka.Web.Api.API.GetSamplesByStreams (int NumOfRecords, SamplesSelector[] SamplesSelector)

The function receives a number of streams to pull out of the DB, and an array of NewSamplesSelector objects. Each NewSamplesSelector holds the requested stream id, and an id of the last requested sample. The function returns a list of samples, starting from the id of the last requested sample, and a relevant metadata.

Parameters

<i>SamplesReq</i>	Number of stream ids, and an array of NewSamplesSelector objects. See more at GetSamplesReq and NewSamplesSelector
-------------------	--

Returns

List of samples and a relevant metadata. See more at GetSamplesRes

7.1.2.7 ServiceArrayResponse<Containers.Site> Ayyeka.Web.Api.API.GetSites ()

The function receives a number of sites to be pulled from the database, and returns the IDs of the relevant to the user sites, their names, and a RetCode that indicates the state of the request.

Parameters

<i>siteReq</i>	Number of sites to pull out. See more at GetSitesReq
----------------	--

Returns

List of sites and other metadata. See more at GetSitesRes

7.1.2.8 ServiceArrayResponse<Containers.Stream> Ayyeka.Web.Api.API.GetStreamsBySite (ulong SiteId)

The function receives a site id, and a number of streams to pull out of the DB, and returns the ids of the relevant streams, their names, and a RetCode that indicates the state of the request.

Parameters

<i>streamReq</i>	Site id and a number of streams to pull out. See more at GetStreamsReq
------------------	--

Returns

List of streams and other metadata. See more at GetStreamsRes

7.1.2.9 ServiceScalarResponse<uint> Ayyeka.Web.Api.API.SendRebootCommand (ulong DeviceId)

Use this method to submit the command to reboot the device.

Parameters

<i>DeviceId</i>	Device Id
-----------------	-----------

Returns

7.1.2.10 ServiceScalarResponse<uint> Ayyeka.Web.Api.API.SendReportingIntervalCommand (ulong *DeviceId*, uint *TypeId*, uint *ReportingInterval*)

Use this method to submit a device command to change the interval defined in specified Reporting Type. Read more about Reporting Types in GetDeviceReportingTypes method summary.

Parameters

<i>DeviceId</i>	Device Id
<i>TypeId</i>	Reporting Type id
<i>ReportingInterval</i>	Reporting Interval in minutes

Returns

7.1.2.11 ServiceScalarResponse<uint> Ayyeka.Web.Api.API.SendSetSettingCommand (ulong *DeviceId*, string *ModuleId*, string *SettingId*, int *Index*, string *Variant*, string *Value*)

Use this method to submit DeviceCommand to set new value for one of the device settings. See GetDeviceCurrentSettings summary for details.

Parameters

<i>DeviceId</i>	Device Id
<i>ModuleId</i>	Firmware Module Id
<i>SettingId</i>	
<i>Index</i>	
<i>Variant</i>	
<i>Value</i>	

Returns

The documentation for this class was generated from the following file:

- C:/Development/apisrv/API/APIv2.1/API.asmx.cs

7.2 Ayyeka.Web.Api.Authentication Class Reference

Web (p. 13) Service for connecting to **Ayyeka** (p. 13)'s services and data. Each of **Ayyeka** (p. 13)'s service expects to receive the Ayyeka-Auth-Token: <token-string> HTTP header, where <token-string> is a long string received in **Login(LoginReq req)** (p. 20) web method. This token is temporary: it is created for a client with a timeout that expires if it is not in use, that is, in case the client does not initiate any new requests for any of **Ayyeka** (p. 13)'s web services for some time. After the token expires, the client can request a new token from **Login(LoginReq req)** (p. 20).

Inherits WebService.

Classes

- class **LoginReq**

*Login Request object. Used in **Login(LoginReq req)** (p. 20). Contains the Username and Password properties, both of string type.*

- class **LoginRes**

*Login response object. Used in **Login(LoginReq req)** (p. 20). This object has the properties listed below:*

Public Types

- enum **AuthenticationSvcRetCodes** { **OK** = 0, **ErrorSystem** = -1, **ErrorAuthenticationFailed** = -2 }

***Authentication** (p. 19) Service Return Codes. These codes define the response state - OK, ErrorSystem and ErrorAuthenticationFailed*

Public Member Functions

- **LoginRes Login (LoginReq req)**

The function receives the username and password, and returns an authentication token on success, or an error message on failure, for example when the username or the password is incorrect.

7.2.1 Detailed Description

Web (p. 13) Service for connecting to **Ayyeka** (p. 13)'s services and data. Each **Ayyeka** (p. 13)'s service expects to receive the Ayyeka-Auth-Token: <token-string> HTTP header, where <token-string> is a long string received in **Login(LoginReq req)** (p. 20) web method. This token is temporary: it is created for a client with a timeout that expires if it is not in use, that is, in case the client does not initiate any new requests for any of **Ayyeka** (p. 13)'s web services for some time. After the token expires, the client can request a new token from **Login(LoginReq req)** (p. 20).

7.2.2 Member Enumeration Documentation

7.2.2.1 enum Ayyeka.Web.Api.Authentication.AuthenticationSvcRetCodes

Authentication (p. 19) Service Return Codes. These codes define the response state - OK, ErrorSystem and ErrorAuthenticationFailed

7.2.3 Member Function Documentation

7.2.3.1 LoginRes Ayyeka.Web.Api.Authentication.Login (LoginReq req)

The function receives the username and password and returns an authentication token on success or an error message on failure, for example if the username or password is incorrect.

Parameters

<i>req</i>	username and password. See more at LoginReq (p. 21)
------------	--

Returns

Authentication (p. 19) token and any relevant metadata. See more at **LoginRes** (p. 21)

The documentation for this class was generated from the following file:

- C:/Development/apisrv/API/APIGateway/Authentication.asmx.cs

7.3 Ayyeka.Web.Api.Authentication.LoginReq Class Reference

Login Request object. Used in **Login(LoginReq req)** (p.20). Contains the Username and Password properties, both of string type.

Properties

- string **Username** [get, set]
- string **Password** [get, set]

7.3.1 Detailed Description

Login Request object. Used in **Login(LoginReq req)** (p.20). Contains the Username and Password properties, both of string type.

The documentation for this class was generated from the following file:

- C:/Development/apisrv/API/APIGateway/Authentication.asmx.cs

7.4 Ayyeka.Web.Api.Authentication.LoginRes Class Reference

Login response object. Used in **Login(LoginReq req)** (p.20). This object has the properties listed below:

Properties

- **AuthenticationSvcRetCodes RetCode** [get, set]
- string **ErrorMessage** [get, set]
- string **AuthToken** [get, set]
- long **UserID** [get, set]

7.4.1 Detailed Description

Login response object. Used in **Login(LoginReq req)** (p.20). This object has the properties listed below:

AuthenticationSvcRetCodes RetCode - an enum return code, could be:

OK = 0,

ErrorSystem = -1,

ErrorAuthenticationFailed = -2.

string ErrorMessage - an error message, in case there is one.

string AuthToken - an authentication token that should be added to any HTTP request for any of **Ayyeka** (p.13)'s web services.

Use: Ayyeka-Auth-Token: <token-string>

The documentation for this class was generated from the following file:

- C:/Development/apisrv/API/APIGateway/Authentication.asmx.cs

7.5 Ayyeka.Web.Api.ServiceArrayResponse< T > Class Template Reference

Service Response is a wrapper class. It wraps a generic array of values with the relevant metadata. The metadata is the response return codes and error messages.

Properties

- **ReturnCodes RetCode** [get, set]
- string **ErrorMessage** [get, set]
- T[] **Value** [get, set]

7.5.1 Detailed Description

Service Response is a wrapper class. It wraps a generic array of values with the relevant metadata. The metadata is the response return codes and error messages.

Template Parameters

<i>T</i>	Generic Response data. For example: Site, Stream, Sample
----------	--

The documentation for this class was generated from the following file:

- C:/Development/apisrv/API/APIv2.1/API.asmx.cs

7.6 Ayyeka.Web.Api.ServiceScalarResponse< T > Class Template Reference

Properties

- **ReturnCodes RetCode** [get, set]
- string **ErrorMessage** [get, set]
- T **Value** [get, set]

The documentation for this class was generated from the following file:

- C:/Development/apisrv/API/APIv2.1/API.asmx.cs

Index

- AuthenticationSvcRetCodes
 - Ayyeka::Web::Api::Authentication, 20
- Ayyeka, 13
- Ayyeka.Web, 13
- Ayyeka.Web.Api, 13
- Ayyeka.Web.Api.API, 15
- Ayyeka.Web.Api.Authentication, 19
- Ayyeka.Web.Api.Authentication.LoginReq, 21
- Ayyeka.Web.Api.Authentication.LoginRes, 21
- Ayyeka.Web.Api.ServiceArrayResponse< T >, 21
- Ayyeka.Web.Api.ServiceScalarResponse< T >, 22
- Ayyeka::Web::Api
 - ReturnCodes, 14
- Ayyeka::Web::Api::API
 - CreateOnDeviceStreamRule, 16
 - GetDeviceCommandStatus, 17
 - GetDeviceReportingTypes, 17
 - GetOnDeviceStreamRule, 17
 - GetOnDeviceStreamRuleCreationStatus, 17
 - GetSamplesByStreams, 18
 - GetSites, 18
 - GetStreamsBySite, 18
 - SendRebootCommand, 18
 - SendReportingIntervalCommand, 19
 - SendSetSettingCommand, 19
- Ayyeka::Web::Api::Authentication
 - AuthenticationSvcRetCodes, 20
 - Login, 20
- CreateOnDeviceStreamRule
 - Ayyeka::Web::Api::API, 16
- GetDeviceCommandStatus
 - Ayyeka::Web::Api::API, 17
- GetDeviceReportingTypes
 - Ayyeka::Web::Api::API, 17
- GetOnDeviceStreamRule
 - Ayyeka::Web::Api::API, 17
- GetOnDeviceStreamRuleCreationStatus
 - Ayyeka::Web::Api::API, 17
- GetSamplesByStreams
 - Ayyeka::Web::Api::API, 18
- GetSites
 - Ayyeka::Web::Api::API, 18
- GetStreamsBySite
 - Ayyeka::Web::Api::API, 18
- Login
 - Ayyeka::Web::Api::Authentication, 20
- ReturnCodes
 - Ayyeka::Web::Api, 14
- SendRebootCommand
 - Ayyeka::Web::Api::API, 18
- SendReportingIntervalCommand
 - Ayyeka::Web::Api::API, 19
- SendSetSettingCommand
 - Ayyeka::Web::Api::API, 19