MyGLS API
for online parcel processing

Date / time: 2019. 07. 24, 10:49:06

Table of contents

Change log

Change No.	Date version	Description	Since
1	2019-02-01	Brand new API for MyGLS	2019-02-01

Getting started

Before you can start using this ***API*** for MyGLS system, you got to have an agreement with GLS. If you don't have the required MyGLS login credentials please contact GLS company.

Online API communication is designed via ***HTTPS***.

MyGLS API supports two approaches: ***SOAP*** (format ***XML***) and ***REST*** (format ***JSON*** or ***XML***).

API service data assembly

API development team has decided to release shared assembly *GLS.MyGLS.APIServiceData.dll* and code snippets (C# / PHP).
You can download it from here.
Library contains declaration of all API request/response classes, enumerations, etc.
It is possible to use it in third party solutions by adding reference to project.

Authentication and authorization

Every calling of API method has to be authenticated via request parameter.
You need *user name (email)* and *password*.
Don't forget to fill it in all requests.

```
public partial class APIRequestBase
{
    public APIRequestBase()
    {
        Initialize();
    }

    partial void Initialize();

    /// <summary>
    /// Password from MyGLS encrypted with SHA512 algorithm
    /// </summary>
    public byte[] Password { get; set; }
    /// <summary>
    /// MyGLS user name (email address) used to authorize request
    /// </summary>
    public string Username { get; set; }
}
```

For calling all API methods user need MyGLS authorization *"Create Parcel"*.
Password in string representation must be encrypted with ***SHA512*** algorithm to byte array.

Password SHA512 C# implementation

```
/// <summary>
/// From password calculates hash to byte array
/// </summary>
/// <param name="password">raw password</param>
/// <returns>password hash</returns>
public static byte[] Sha512(string password)
{
    using (SHA512 sha512 = new SHA512Managed())
    {
        return sha512.ComputeHash(Encoding.UTF8.GetBytes(password));
    }
}
```

Password SHA512 PHP implementation

```
<?php
  $password_hash = hash('sha512', $password);
?>
```

Country domain API URLs

We are supporting MyGLS ***API*** for 6 countries now.
Mind to use the right country ***domain***.

	Country		API URL
1	Croatia	SOAP	https://api.mygls.**hr**/ParcelService.svc?singleWsdl
		REST	https://api.mygls.**hr**/ParcelService.svc/{format}/{methodName}
		testing	https://api.**test**.mygls.**hr**/...
2	Czechia	SOAP	https://api.mygls.**cz**/ParcelService.svc?singleWsdl
		REST	https://api.mygls.**cz**/ParcelService.svc/{format}/{methodName}
		testing	https://api.**test**.mygls.**cz**/...
3	Hungary	SOAP	https://api.mygls.**hu**/ParcelService.svc?singleWsdl
		REST	https://api.mygls.**hu**/ParcelService.svc/{format}/{methodName}
		testing	https://api.**test**.mygls.**hu**/...
4	Romania	SOAP	https://api.mygls.**ro**/ParcelService.svc?singleWsdl
		REST	https://api.mygls.**ro**/ParcelService.svc/{format}/{methodName}
		testing	https://api.**test**.mygls.**ro**/...
5	Slovenia	SOAP	https://api.mygls.**si**/ParcelService.svc?singleWsdl
		REST	https://api.mygls.**si**/ParcelService.svc/{format}/{methodName}
		testing	https://api.**test**.mygls.**si**/...
6	Slovakia	SOAP	https://api.mygls.**sk**/ParcelService.svc?singleWsdl
		REST	https://api.mygls.**sk**/ParcelService.svc/{format}/{methodName}
		testing	https://api.**test**.mygls.**sk**/...

Handy structures used in request/response data exchange

Parcel class

Object containing necessary data for printing labels.

Property	DataType	Description
ClientNumber	Integer	Unique client number provided by GLS company. REQUIRED
ClientReference	String	Client custom tag identifying parcel. STRONGLY RECOMMENDED
Count	Integer	Count of parcels sent in one shipment. DEFAULT 1
CODAmount	Decimal	Cash on delivery amount. NOT REQUIRED
CODReference	String	Cash on delivery client reference number used for payment pairing. REQUIRED if CODAmount is filled.
Content	String	Parcel info printed on label.
PickupDate	DateTime	Pick up date. DEFAULT actual date
PickupAddress	Address	The address of place where courier pick up the shipment. REQUIRED
DeliveryAddress	Address	The address of destination place. REQUIRED
ServiceList	List<Service>	Services and their special parameters.

Address class

Object containing address of pick up place and parcel destination.

Property	DataType	Description
Name	String	Name of the person or organization. REQUIRED
Street	String	Name of the street. REQUIRED
HouseNumber	String	Number of the house.
City	String	Name of the town or village. REQUIRED
ZipCode	String	Area Zip code. REQUIRED
CountryIsoCode	String	Two letter country code defined in ISO 3166-1. More…
ContactName	String	Name of person which can be asked or inform about shipment details by GLS.
ContactPhone	String	Phone number of person which can be asked or inform about shipment details by GLS.
ContactEmail	String	Email address of person which can be asked or inform about shipment details by GLS.

Service class

Object containing specific service settings.

Property	DataType	Description
Code	String	Service code (see Appendix B: List of services). REQUIRED
ADRParameter	ServiceParameterADR	Settings for ADR service REQUIRED FOR "ADR" SERVICE
AOSParameter	ServiceParameterString	Settings for AOS service REQUIRED FOR "AOS" SERVICE
CS1Parameter	ServiceParameterString	Settings for CS1 service REQUIRED FOR "CS1" SERVICE CODE
DDSParameter	ServiceParameterDateTime	Settings for DDS service REQUIRED FOR "DDS" SERVICE CODE
DPVParameter	ServiceParameterStringDecimal	Settings for DPV service REQUIRED FOR "DPV" SERVICE CODE
FDSParameter	ServiceParameterString	Settings for FDS service REQUIRED FOR "FDS" SERVICE CODE
FSSParameter	ServiceParameterString	Settings for FSS service REQUIRED FOR "FSS" SERVICE CODE
INSParameter	ServiceParameterDecimal	Settings for INS service REQUIRED FOR "INS" SERVICE CODE
MMPParameter	ServiceParameterDecimal	Settings for MMP service REQUIRED FOR "MMP" SERVICE CODE
PSDParameter	ServiceParameterStringInteger	Settings for PSD service REQUIRED FOR "PSD" SERVICE CODE
SDSParameter	ServiceParameterTimeRange	Settings for SDS service REQUIRED FOR "SDS" SERVICE CODE
SM1Parameter	ServiceParameterString	Settings for SM1 service REQUIRED FOR "SM1" SERVICE CODE
SM2Parameter	ServiceParameterString	Settings for SM2 service REQUIRED FOR "SM2" SERVICE CODE
SZLParameter	ServiceParameterString	Settings for SZL service REQUIRED FOR "SZL" SERVICE CODE
Value	String	Service value without previous special service settings

Error lists in response classes

Each response class from API methods includes list of error objects describing potential problems.

```
public DeleteLabelsResponse()
{
    DeleteLabelsErrorList = new List<ErrorInfo>();
...

public GetPrintDataResponse()
{
    GetPrintDataErrorList = new List<ErrorInfo>();
...

public GetPrintedLabelsResponse()
{
    GetPrintedLabelsErrorList = new List<ErrorInfo>();
...

public PrepareLabelsResponse()
{
    PrepareLabelsError = new List<ErrorInfo>();
...

etc.
```

ErrorInfo class

Property	DataType	Description
ErrorCode	Integer	Value from error enumeration (see Appendix A).
ErrorDescription	String	Human readable error description (see Appendix A) or exception message trying to describe problem.
ClientReferenceList	List<String>	List of client parcel tags identifying parcels where specific error happened.
ParcelIdList	List<Integer>	List of database parcel ID identifying parcel records where specific error happended.

```
/// <summary>
/// Error class - list of validation errors if they are present
/// </summary>
public partial class ErrorInfo
{
    public ErrorInfo()
    {
        ClientReferenceList = new List<string>();
        ParcelIdList = new List<int>();

        Initialize();
    }

    partial void Initialize();

    /// <summary>
    /// Error code - see documentation
    /// </summary>
    public int ErrorCode { get; set; }

    /// <summary>
    /// Error code description
    /// </summary>
    public string ErrorDescription { get; set; }

    /// <summary>
    /// Customers parcel tags
    /// </summary>
    public List<string> ClientReferenceList { get; set; }

    /// <summary>
    /// Database parcel ID
    /// </summary>
    public List<int> ParcelIdList { get; set; }
}
```

Supported API parcel operations

PrepareLabels

Validates parcel data for labels and adds valid parcel data to database.

```
PrepareLabelsResponse PrepareLabels(PrepareLabelsRequest prepareLabelsRequest);
```

Request

Method	URL examples
POST	https://api.mygls.hu/ParcelService.svc/json/PrepareLables **(LATEST VERSION, format JSON, for HU)**
	https://api.mygls.cz/ParcelService.svc/xml/PrepareLables_20190201 **(SPECIFIC VERSION, format XML, for CZ)**

Request class PrepareLabelsRequest

Property	DataType	Description
ParcelList	List<Parcel>	List of labels/parcels data . REQUIRED NOT EMPTY

```
public partial class PrepareLabelsRequest : APIRequestBase
{
    public PrepareLabelsRequest()
    {
        ParcelList = new List<ServiceData.APIDTOs.LabelOperations.Parcel>();

        Initialize();
    }

    partial void Initialize();

    /// <summary>
    /// Parcel list contains mandatory information for labels
    /// </summary>
    public List<ServiceData.APIDTOs.LabelOperations.Parcel> ParcelList { get; set; }
}
```

Response class PrepareLabelsResponse

Property	DataType	Description
ParcelInfoList	List<ParcelInfo>	List of successfully prepared records (ID and ClientReference) for generating labels.
PrepareLabelsError	List<ErrorInfo>	List of potential errors (see Error lists in response classes)

```
public partial class PrepareLabelsResponse
{
    public PrepareLabelsResponse()
    {
        ParcelInfoList = new List<ServiceData.APIDTOs.LabelOperations.ParcelInfo>();
        PrepareLabelsError = new List<ServiceData.APIDTOs.Common.ErrorInfo>();

        Initialize();
    }

    partial void Initialize();

    public List<ServiceData.APIDTOs.LabelOperations.ParcelInfo> ParcelInfoList { get; set; }
    public List<ServiceData.APIDTOs.Common.ErrorInfo> PrepareLabelsError { get; set; }
}
```

ParcelInfo class

Property	DataType	Description
ClientReference	String	Client custom tag identifying parcel.
ParcelId	Integer	Label/Parcel database record ID.

```
    public partial class ParcelInfo
    {
        public ParcelInfo()
        {

            Initialize();
        }

        partial void Initialize();

        /// <summary>
        /// Returned Client reference of created parcel (only when is present in request)
        /// </summary>
        public string ClientReference { get; set; }
        /// <summary>
        /// Parcel Id of created parcel
        /// </summary>
        public int ParcelId { get; set; }
    }
```

GetPrintedLabels

Generates parcel numbers and PDF document contains labels in byte array format.

```
GetPrintedLabelsResponse GetPrintedLabels(GetPrintedLabelsRequest getPrintedLabelsRequest);
```

Request

Method	URL examples
POST	https://api.mygls.hu/ParcelService.svc/json/GetPrintedLables **(LATEST VERSION, format JSON, for HU)**
	https://api.mygls.cz/ParcelService.svc/xml/GetPrintedLabels_20190201 **(SPECIFIC VERSION, format XML, for CZ)**

Request class GetPrintedLabelsRequest

Property	DataType	Description
ParcelIdList	List<Integer>	List of labels/parcels database record ID. REQUIRED NOT EMPTY
PrintPosition	Integer	Number of page quarter ACCEPTED ONLY FOR A4-FORMAT 1 2 3 4
ShowPrintDialog	Bool	Flag for third party PDF reader (if supported application shows print dialog immediately after opening document)

```
public partial class GetPrintedLabelsRequest : APIRequestBase
{
    public GetPrintedLabelsRequest()
    {
        ParcelIdList = new List<int>();
        Initialize();
    }

    partial void Initialize();
    /// <summary>
    /// List of parcel IDs to be returned in label stream.
    /// </summary>
    public List<int> ParcelIdList { get; set; }
    /// <summary>
    /// Starting position for printing labels, used only in A4 format
    /// </summary>
    public int PrintPosition { get; set; }
    /// <summary>
    /// True - show print dialog automatically
    /// </summary>
    public bool ShowPrintDialog { get; set; }
}
```

Response class GetPrintedLabelsResponse

Property	DataType	Description
Labels	Byte[]	PDF document in byte array.
GetPrintedLabelsErrorList	List<ErrorInfo>	List of potential errors (see Error lists in response classes)

```
public partial class GetPrintedLabelsResponse
{
    public GetPrintedLabelsResponse()
    {
        GetPrintedLabelsErrorList = new List<ErrorInfo>();
        Initialize();
    }

    partial void Initialize();

    public List<ErrorInfo> GetPrintedLabelsErrorList { get; set; }
    /// <summary>
    /// Returned labels byte stream.
    /// </summary>
    public byte[] Labels { get; set; }
}
```

PrintLabels

Calls both PrepareLabels and GetPrintedLabels in one step.

So, it validates parcel data for labels, adds valid parcel data to database, generates parcel numbers and PDF document containing labels in byte array format.

```
PrintLabelsResponse PrintLabels(PrintLabelsRequest printLabelsRequest);
```

Request

Method	URL examples
POST	https://api.mygls.hu/ParcelService.svc/json/PrintLables **(LATEST VERSION, format JSON, for HU)**
	https://api.mygls.cz/ParcelService.svc/xml/PrintLables_20190201 **(SPECIFIC VERSION, format XML, for CZ)**

Request class PrintLabelsRequest

Property	DataType	Description
ParcelList	List<Parcel>	List of labels/parcels data . REQUIRED NOT EMPTY
PrintPosition	Integer	Number of page quarter ACCEPTED ONLY FOR A4-FORMAT 1 2 3 4
ShowPrintDialog	Bool	Flag for third party PDF reader (if supported application shows print dialog immediately after opening document)

```
public partial class PrintLabelsRequest : APIRequestBase
{
    public PrintLabelsRequest()
    {
        ParcelList = new List<ServiceData.APIDTOs.LabelOperations.Parcel>();
        Initialize();
    }

    partial void Initialize();
    /// <summary>
    /// Parcel list contains mandatory information for labels
    /// </summary>
    public List<ServiceData.APIDTOs.LabelOperations.Parcel> ParcelList { get; set; }
    /// <summary>
    /// Starting position for printing labels (used only for A4 format)
    /// </summary>
    public int PrintPosition { get; set; }
    /// <summary>
    /// True - show print dialog automatically
    /// </summary>
    public bool ShowPrintDialog { get; set; }
}
```

Response class PrintLabelsResponse

Property	DataType	Description
Labels	Byte[]	PDF document in byte array.

Property	DataType	Description
PrintLabelsErrorList	List<ErrorInfo>	List of potential errors (see Error lists in response classes)
PrintLabelsInfoList	List<PrintLabelsInfo>	List of successfully prepared records (ID and ClientReference) for generating labels.

```
public partial class PrintLabelsResponse
{
    public PrintLabelsResponse()
    {
        PrintLabelsErrorList = new List<ErrorInfo>();
        PrintLabelsInfoList = new List<PrintLabelsInfo>();
        Initialize();
    }

    partial void Initialize();
    /// <summary>
    /// Byte array labels response. Default is PDF format
    /// </summary>
    public byte[] Labels { get; set; }
    public List<ErrorInfo> PrintLabelsErrorList { get; set; }
    public List<PrintLabelsInfo> PrintLabelsInfoList { get; set; }
}
```

PrintLabelsInfo class (inherits ParcelInfo)

Property	DataType	Description
ClientReference	String	Client custom tag identifying parcel.
ParcelId	Integer	Label/Parcel database record ID.
ParcelNumber	Long	Parcel number

```
public partial class PrintLabelsInfo : ParcelInfo
{
    public PrintLabelsInfo()
    {
        Initialize();
    }

    partial void Initialize();
    /// <summary>
    /// Parcel number of printed parcel
    /// </summary>
    public long ParcelNumber { get; set; }
}
```

GetPrintData

Validates parcel data for labels, adds valid parcel data to database, generates parcel numbers and returns data for custom generating labels.

```
GetPrintDataResponse GetPrintData(GetPrintDataRequest getPrintDataRequest);
```

Request

Method	URL examples
POST	https://api.mygls.hu/ParcelService.svc/json/GetPrintData **(LATEST VERSION, format JSON, for HU)**
	https://api.mygls.cz/ParcelService.svc/xml/GetPrintData_20190201 **(SPECIFIC VERSION, format XML, for CZ)**

Request class GetPrintDataRequest

Property	DataType	Description
ParcelList	List<Parcel>	List of labels/parcels data . REQUIRED NOT EMPTY

```
public partial class GetPrintDataRequest : APIRequestBase
{
    public GetPrintDataRequest()
    {
        ParcelList = new List<ServiceData.APIDTOs.LabelOperations.Parcel>();
        Initialize();
    }

    partial void Initialize();
    public List<ServiceData.APIDTOs.LabelOperations.Parcel> ParcelList { get; set; }
}
```

Response class GetPrintDataResponse

Property	DataType	Description
GetPrintDataErrorList	List<ErrorInfo>	List of potential errors (see Error lists in response classes)
PrintDataInfoList	List<PrintDataInfo>	List of successfully prepared records (ID, ClientReference, ParcelNumber) for generating labels.

```
public partial class GetPrintDataResponse
{
    public GetPrintDataResponse()
    {
        GetPrintDataErrorList = new List<ErrorInfo>();
        PrintDataInfoList = new List<PrintDataInfo>();
        Initialize();
    }

    partial void Initialize();
    public List<ErrorInfo> GetPrintDataErrorList { get; set; }
    public List<PrintDataInfo> PrintDataInfoList { get; set; }
}
```

PrintDataInfo class

Property	DataType	Description
Parcel	Parcel	Object containing necessary data for printing labels.
ParcelId	Integer	Label/Parcel database record ID.
ParcelNumber	Long	Parcel number
ParcelNumberWithCheckdigit	Long	Parcel number contains last check-digit
DepotNumber	String	Depot number
TourNumber	String	Driver tour number

```
public partial class PrintDataInfo
{
    public PrintDataInfo()
    {
        Initialize();
    }

    partial void Initialize();
    /// <summary>
    /// Depot number - optional
    /// </summary>
    public string DepotNumber { get; set; }
    public GLS.MyGLS.ServiceData.APIDTOs.LabelOperations.Parcel Parcel { get; set; }
    /// <summary>
    /// Stored Parcel Id for future use
    /// </summary>
    public int ParcelId { get; set; }
    /// <summary>
    /// Stored parcel number
    /// </summary>
    public long ParcelNumber { get; set; }
    /// <summary>
    /// Parcel number in full format with checkdigit (must be printed on label)
    /// </summary>
    public long ParcelNumberWithCheckdigit { get; set; }
    /// <summary>
    /// Driver tour number - optional
    /// </summary>
    public string TourNumber { get; set; }
}
```

DeleteLabels

Set DELETED state for already printed labels/parcels with specific database record ID.

```
DeleteLabelsResponse DeleteLabels(DeleteLabelsRequest deleteLabelsRequest);
```

Request

Method	URL examples
POST	https://api.mygls.hu/ParcelService.svc/json/DeleteLables **(LATEST VERSION, format JSON, for HU)**
	https://api.mygls.cz/ParcelService.svc/xml/DeleteLables_20190201 **(SPECIFIC VERSION, format XML, for CZ)**

Request class DeleteLabelsRequest

Property	DataType	Description
ParcelIdList	List<Integer>	List of labels/parcels database ID. REQUIRED NOT EMPTY MAX. 50 ITEMS PER REQUEST

```
/// <summary>
/// Delete labels request class
/// </summary>
public partial class DeleteLabelsRequest : APIRequestBase
{
    public DeleteLabelsRequest()
    {
        ParcelIdList = new List<int>();

        Initialize();
    }

    partial void Initialize();

    /// <summary>
    /// List of parcel ID to be deleted.
    /// </summary>
    public List<int> ParcelIdList { get; set; }
}
```

Response class DeleteLabelsResponse

Property	Description
SuccessfullyDeletedList	List of successfully deleted labels/parcels database ID. When deleted parcel wasn't only one in shipment, there is filled array of sub-parcels ID.
DeleteLabelsErrorList	List of potential errors (see Error lists in response classes)

```
public partial class DeleteLabelsResponse
{
    public DeleteLabelsResponse()
    {
        DeleteLabelsErrorList = new List<Common.ErrorInfo>();
        SuccessfullyDeletedList = new List<SuccessfullyDeleted>();

        Initialize();
    }

    partial void Initialize();

    public List<Common.ErrorInfo> DeleteLabelsErrorList { get; set; }
    public List<SuccessfullyDeleted> SuccessfullyDeletedList { get; set; }
}
```

ModifyCOD

Changes COD amount for specific parcel.

```
ModifyCODResponse ModifyCOD(ModifyCODRequest modifyCODRequest);
```

Request

Method	URL examples
POST	https://api.mygls.hu/ParcelService.svc/json/ModifyCOD **(LATEST VERSION, format JSON, for HU)**
	https://api.mygls.cz/ParcelService.svc/xml/ModifyCOD_20190201 **(SPECIFIC VERSION, format XML, for CZ)**

Request class ModifyCODRequest

Property	DataType	Description
ParcelId	Integer	Label/parcel database ID. REQUIRED IF ParcelNumber IS NULL
ParcelNumber	Long	Parcel number. REQUIRED IF ParcelId IS NULL
CODAmount	Decimal	Cash on delivery amount. ZERO OR POSITIVE

```
public partial class ModifyCODRequest : APIRequestBase
{
    public ModifyCODRequest()
    {
        Initialize();
    }

    partial void Initialize();
    /// <summary>
    /// New cash on delivery amount
    /// </summary>
    public decimal CODAmount { get; set; }
    /// <summary>
    /// Parcel ID to change cash on delivery amount - optional
    /// </summary>
    public int? ParcelId { get; set; }
    /// <summary>
    /// Parcel Number to change cash on delivery amount - optional
    /// </summary>
    public long? ParcelNumber { get; set; }
}
```

Response class ModifyCODResponse

Property	DataType	Description
Successful	Bool	True = modifying COD without error
ModifyCODError	List<ErrorInfo>	List of potential errors (see Error lists in response classes)

```
public partial class ModifyCODResponse
{
    public ModifyCODResponse()
    {
        ModifyCODError = new List<ErrorInfo>();
        Initialize();
    }

    partial void Initialize();
    public List<ErrorInfo> ModifyCODError { get; set; }
    /// <summary>
    /// true - modify cash on delivery completed without error
    /// </summary>
    public bool Successful { get; set; }
}
```

Appendix A: API error codes

Error number	Meaning	Since change number
1	Request parameter is null	1
2	Parcel ID list is null	1
3	Parcel ID list is empty	1
4	Parcel ID not exists	1
5	Access denied for this parcel ID	1
6	Parcel with this ID has different status than PRINTED	1
7	Missing parcel data in request	1
8	COD amount has to be >= 0	1
9	Parcel number not exists	1
10	Parcel number was not assigned yet	1
11	Parcel list is null	1
12	Parcel list is empty	1
13	Parcel validation issue	1
14	User not exists	1
15	User is not authorized to access parcel	1
16	Label is empty	1
17	There are no parcel numbers	1
18	Parcel label is already generated	1
19	Parcel number generator failed	1
20	Parcel numbers were not generated	1
21	There are no printable labels	1
22	Count of parcels for deleting is out of limit	1
1000	Unexpected exception happened	1
1001	Internal Problem	1

Appendix B: List of services

Not all services are available in each country or area.

Service code	Service name	Parameter
24H	Service guaranteed delivery shipment in 24 Hours	
ADR	Agreement about Dangerous goods by Road	
AOS	Addressee Only Service	
COD	Cash On Delivery service	
CS1	Contact Service	
DAW	Delivery At Work service	
DDS	Day Definite Service	
DPV	Declared Parcel Value service	
FDS	Flexible Delivery Service	
FSS	Flexible delivery Sms Service	
INS	Insurance Service	
LDS		
MCC		
MMP	Middle Man Price service	
PCC		
PRS		
PSD	Parcel Shop Delivery service	
PSS	Pick & Ship Service	
SAT	SATurday service	
SBS	Stand By Service	
SDS	Scheduled Delivery Service	
SM1	SMs service	
SM2	SMs pre-advice	
SRS		
SZL	document return service (SZáLlítólevél visszaforgatás)	
T09	Express service	
T10	Express service	
T12	Express service	
TGS	Think Green Service	
XS	Exchange Service	

Appendix C: Copy / Paste snippet section

Password SHA512 C# implementation

```
/// <summary>
/// From password calculates hash to byte array
/// </summary>
/// <param name="password">raw password</param>
/// <returns>password hash</returns>
public static byte[] Sha512(string password)
{
    using (SHA512 sha512 = new SHA512Managed())
    {
        return sha512.ComputeHash(Encoding.UTF8.GetBytes(password));
    }
}
```

Password SHA512 PHP implementation

```
<?php
    $password_hash = hash('sha512', $password);
?>
```

Appendix D: Jargon

API (Application Program Interface) is a set of routines, rules and tools for building software. In this case, it is a set of clearly defined methods for parcel processing. It helps to develop fast and clear communication between MyGLS online system and customer systems. More...

Domain (in this case means "country code top level domain") is used and reserved for country, sovereign state or territory identified with a country code. More...

HTTPS (Hypertext Transfer Protocol Secure) is used for encrypted communication over a computer network. More...

JSON (JavaScript Object Notation) is an open standard file format that uses human readable object consisting of attribute-value pair and array data. More...

REST (Representational State Transfer) is a software architectural style that defines a set of constraints to be used for creating web services. More...

SHA512 (Secure Hash Algorithm 512 bits) is a cryptographic hash function from set of SHA-2 family. Method computes 64 bytes (512 bits / 8 bits per byte) from any content. Every byte can store values 0-255 (0x00-0xFF), so hexadecimal string takes 128 chars (64 bytes * 2 characters per byte). More...

SOAP (Simple Object Access Protocol) is specification for exchanging structured information in web service implementation. It uses XML for describing message format. More...

URL (Uniform Resource Locator) is reference to specific web resource – network location and a mechanism for retrieving it. More...

XML (Extensible Markup Language) defines a set of rules for encoded documents in a format that is both human and machine readable. More...