
MyGLS API
for XXL parcels

Table of contents

Change log.....	3
Getting started.....	3
Used data structures in request/response objects.....	4
XXL data can be sent in PrepareLabels and PrintLabels method within the Parcel class.....	4
Parcel class.....	4
ParcelPropertyList class	4
Request	5
Request class GetParcelListToDispatchListRequest.....	5
Request class GetParcelListToDispatchListResponse	5
Request	7
Request class SetDispatchListRequest	7
Request class SetDispatchListResponse.....	7
Request	8
Request class GetDispatchListRequest	8
Request class GetDispatchListResponse	8
Request	10
Request class GetDispatchListReportRequest	10
Response class GetDispatchListReportResponse.....	10
List of services.....	11
Service class	11
Service parameter examples in JSON format	11
Code Examples.....	13
PrintLabels and PrepareLabels requests with ParcelPropertyList class.....	13
ParcelPropertyList class	14-18

Change log

Change No.	Date version	Description	Since
1		<ul style="list-style-type: none">- New methods for XXL dispatch lists- ParcelPropertyList class added to PrintLabels and PrepareLabels methods- Sample codes	2023-05-10
2		<ul style="list-style-type: none">- Services class added- UsedMachineService- New sample codes	2023-11-30

Getting started

This documentation contains information about new class for data sending for XXL parcels and new methods to create and print dispatch list which is needed for XXL parcels.

The documentation for basic MyGLS API can be downloaded from api.myglshu.com website.

Used data structures in request/response objects

XXL data can be sent in PrepareLabels and PrintLabels method within the Parcel class.

Parcel class

Object containing necessary data for printing labels.

Property	Data Type	Description
ClientNumber	Integer	Unique client number provided by GLS company. REQUIRED
ClientReference	String	Client custom tag identifying parcel. STRONGLY RECOMMENDED (It is optionally REQUIRED, depends on the user right.)
Count	Integer	Count of parcels sent in one shipment. (maximum 99) DEFAULT 1
CODAmount	Decimal	Cash on delivery amount. NOT REQUIRED
CODReference	String	Cash on delivery client reference number used for payment pairing. STRONGLY RECOMMENDED if CODAmount is filled.
Content	String	Parcel info printed on label. (It is optionally REQUIRED, depends on the user right.)
PickupDate	DateTime	Pick up date. DEFAULT actual date
PickupAddress	Address	The address of place where courier pick up the shipment. REQUIRED
ParcelPropertyList	ParcelProperties Class	Data of XXL parcels REQUIRED for XXL parcels
DeliveryAddress	Address	The address of destination place. REQUIRED
ServiceList	List<Service>	Services and their special parameters.

ParcelPropertyList class

Property	Data Type	Description
Content	String	Content of the parcel
Height	Integer	
Lenght	Integer	
PackageType	PackageType	Package types can be: 1 – Colli; 2 – Box; 3 – Roll; 4 – Can; 5 – Case; 6 – Reel; 7 – Sack;
Weight	Decimal	Maximum 80kg/parcel
Width	Integer	

Sample code on page 11

GetParcelListToDispatchList

Collects ParcelId for SetDispatchList method.

Request

Method	URL examples
POST	https://api.mygls.hu/ParcelService.svc/json/GetParcelListToDispatchList (format JSON, for HU)

Request class GetParcelListToDispatchListRequest

```
public partial class GetParcelListToDispatchListRequest :
    GLS.MyGLS.ServiceData.APIDTOs.Common.APIRequestBase
{
    public GetParcelListToDispatchListRequest()
    {
        Initialize();
    }

    partial void Initialize();

    public int ClientNumber { get; set; }
    public DateTime? PrintDateFrom { get; set; }
    public DateTime? PrintDateTo { get; set; }
    public bool OnlyNotAddedToDispatch { get; set; }
}
```

Request class GetParcelListToDispatchListResponse

```
public partial class GetParcelListToDispatchListResponse
{
    public GetParcelListToDispatchListResponse()
    {
        ParcelListToDispatchListErrors = new
        List<GLS.MyGLS.ServiceData.APIDTOs.Common.ErrorInfo>();
        PrintDataInfoList = new
        List<GLS.MyGLS.ServiceData.APIDTOs.LabelOperations.PrintDataInfo>();

        Initialize();
    }

    partial void Initialize();

    public List<GLS.MyGLS.ServiceData.APIDTOs.Common.ErrorInfo>
    ParcelListToDispatchListErrors { get; set; }
    public
    List<GLS.MyGLS.ServiceData.APIDTOs.DispatchList.DispatchListParcelInfo>
    ParcelInfoList { get; set; }
}
```

One item of ParcelInfoList

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

    partial void Initialize();

    public string ClientReference { get; set; }
    public string DeliveryAddressName { get; set; }
    public string PickupAddressName { get; set; }
    /// Stored Parcel Id for future use
    public int ParcelId { get; set; }
    public long ParcelNumber { get; set; }
}
```

ParcelId has to be used in SetDispatchList method.

SetDispatchList

Prints dispatch list using ParcelId from GetParcelListToDispatchList or PrintLabels method.

Request

Method	URL examples
POST	https://api.myglshu.com/ParcelService.svc/json/SetDispatchList (format JSON, for HU)

Request class SetDispatchListRequest

```
public partial class SetDispatchListRequest :
    GLS.MyGLS.ServiceData.APIDTOs.Common.APIRequestBase
    {
        public SetDispatchListRequest()
        {
            ParcelIdList = new List<int>();

            Initialize();
        }

        partial void Initialize();

        public int ClientNumber { get; set; }
        public List<int> ParcelIdList { get; set; }
    }
```

Request class SetDispatchListResponse

```
public partial class SetDispatchListResponse
    {
        public SetDispatchListResponse()
        {
            DispatchListErrorList = new List<Common.ErrorInfo>();

            Initialize();
        }

        partial void Initialize();

        public string FormattedId { get; set; }
        public byte[] DispatchListReportPdf { get; set; }
        public List<GLS.MyGLS.ServiceData.APIDTOs.Common.ErrorInfo>
        DispatchListErrorList { get; set; }
    }
```

GetDispatchList

Collects FormattedId for reprint.

Request

Method	URL examples
POST	https://api.mygl.s.hu/ParcelService.svc/json/GetDispatchList (format JSON, for HU)

Request class GetDispatchListRequest

```
public partial class GetDispatchListRequest :
    GLS.MyGLS.ServiceData.APIDTOs.Common.APIRequestBase
{
    public GetDispatchListRequest()
    {
        Initialize();
    }

    partial void Initialize();

    public int ClientNumber { get; set; }
    public DateTime? DateFrom { get; set; }
    public DateTime? DateTo { get; set; }
}
```

Request class GetDispatchListResponse

```
public partial class GetDispatchListResponse
{
    public GetDispatchListResponse()
    {
        DispatchListResultItemList = new List<DispatchList>();
        DispatchListErrorList = new List<Common.ErrorInfo>();

        Initialize();
    }

    partial void Initialize();

    public List<DispatchList> DispatchListResultItemList { get; set; }
    public List<GLS.MyGLS.ServiceData.APIDTOs.Common.ErrorInfo>
    DispatchListErrorList { get; set; }
}
```

One item of DispatchListItemList.

```
public class DispatchList
{
    public string FormattedId { get; set; }
    public int ClientNumber { get; set; }
}
```

FormattedId has to be used in GetDispatchListReport method.

GetDispatchListReport

Reprint dispatch lists. Use the FormattedId from GetDispatchList response.

Request

Method	URL examples
POST	https://api.myglshu.com/ParcelService.svc/json/GetDispatchListReport (format JSON, for HU)

Request class GetDispatchListReportRequest

```
public partial class GetDispatchListReportRequest :
    GLS.MyGLS.ServiceData.APIDTOs.Common.APIRequestBase
{
    public GetDispatchListReportRequest()
    {
        Initialize();
    }

    partial void Initialize();

    public int ClientNumber { get; set; }
    public string FormattedId { get; set; }
}
```

Response class GetDispatchListReportResponse

```
public partial class GetDispatchListReportResponse
{
    public GetDispatchListReportResponse()
    {
        DispatchListReportErrorList = new List<Common.ErrorInfo>();
        Initialize();
    }

    partial void Initialize();

    public byte[] DispatchListReportPdf { get; set; }

    public List<GLS.MyGLS.ServiceData.APIDTOs.Common.ErrorInfo>
        DispatchListReportErrorList { get; set; }
}
```

List of services

Service code	Service name	Parameter
AOS	Addressee Only Service	AOSParameter(Value) REQUIRED (Name of the recipient / contact person can be used)
SZL	Document Return Service	SZLParameter(Value) REQUIRED (Document number – string, max. 15 char)
USM	Used Machine Service	

Service class

Property	DataType	Description
Code	String	Service code
AOSParameter	ServiceParameterString	Setting for AOS service REQUIRED FOR “AOS” SERVICE
SZLParameter	ServiceParameterString	Setting for SZL service REQUIRED FOR “SZL” SERVICE
USMParameter	ServiceParameterString	Setting for USM service REQUIRED FOR “USM” SERVICE

Service parameter examples in JSON format

AOS service

```
"ServiceList": [{
  "Code": "AOS",
  "AOSParameter": {
    "Value": "teszt név"
  }
}]
```

SZL service

```
"ServiceList": [{
  "Code": "SZL",
  "SZLParameter": {
    "Value": "12345678"
  }
}]
```

USM Service

```
"ServiceList": [{
  "Code": "USM",
  "USMParameter": {
    "Content": null,
    "ParcelPropertyList" : [{ // Data of the return parcel
      "Content" : "Rossz mosógép",
      "Height" : 20,
      "Length" : 10,
      "Weight" : 30,
      "Width" : 80,
      "PackageType": 2
    }],
    "DeliveryAddress" : { // DeliveryAddress must be the same like PickupAddress!!!
      "City": "Teszt varos",
      "ContactEmail": "example@gmail.com",
      "ContactName": "Teszt felado",
      "ContactPhone": "+36202222222",
      "CountryIsoCode": "HU",
      "HouseNumber": "15",
      "Name": "Teszt felado",
      "Street": "Teszt Utca",
      "ZipCode": "6000"
    },
    "ReturnAddressType": 1 // UsmServiceParameter enum (Adress: 1, GLS: 2)
  }
}]
```

Code Examples

PrintLabels and PrepareLabels requests with ParcelPropertyList class

```
{
  "ClientNumberList": [],
  "Password": [password in byte array],
  "Username": "test@test.com",
  "ParcelList": [
    {
      "CODAmount": null,
      "CODCurrency": "",
      "CODReference": "Content",
      "ClientNumber": 100000004, //use your own ClientNumber
      "ClientReference": "ClientReference",
      "Content": "CodReference",
      "Count": 1,
      "DeliveryAddress": {
        "City": "Test City",
        "ContactEmail": "example@gmail.com",
        "ContactName": "test consignee",
        "ContactPhone": "+36201234567",
        "CountryIsoCode": "HU",
        "HouseNumber": "15",
        "HouseNumberInfo": "First floor 11.",
        "Name": "consignee name",
        "Street": "test address",
        "ZipCode": "6000"
      },
      "ParcelPropertyList" : [
        {
          "Content" : "furniture",
          "Height" : "100",
          "Length" : "30",
          "PackageType" : "2",
          "Weight": "55",
          "Width" : "30"
        }
      ],
      "ParcelNumber": null,
      "PickupAddress": {
        "City": "test city",
        "ContactEmail": "example@gmail.com",
        "ContactName": "test sender",
        "ContactPhone": "+36201234567",
        "CountryIsoCode": "HU",
        "HouseNumber": "15",

```

MyGLS API for XXL parcels

```

        "HouseNumberInfo": "last door",
        "Name": "sender name",
        "Street": "test street",
        "ZipCode": "6000"
    },
    "PickupDate": "/Date(1683702469000+0200)/",
    "ServiceList": []
}
},
"PrintPosition": 1,
"ShowPrintDialog": false,
"TypeOfPrinter": ""
}

```

ParcelPropertyList class

Property	DataType	Description
Content	String	Content of the parcel
Height	Integer	
Lenght	Integer	
PackageType	PackageType	Package types can be: 1 – Colli; 2 – Box; 3 – Roll; 4 – Can; 5 – Case; 6 – Reel; 7 – Sack;
Weight	Decimal	Maximum 80kg/parcel
Width	Integer	

UsmServiceParameter enum

Value	Description
1: Address	The return package will be delivered to the specified address.
2: GLS	The return package will be delivered to the GLS depot and from there handed over for destruction.

GetParcelListToDispatchList

Request

```
{
  "ClientNumberList": [],
  "Password": [password in byte array],
  "Username": "test@test.com",
  "ClientNumber": 100000004, // user your own ClientNumber
  "OnlyNotAddedToDispatch": true,
  "PrintDateFrom": "/Date(1683669600000+0200)/",
  "PrintDateTo": "/Date(1683669600000+0200)/"
}
```

Response

```
{
  "ParcelInfoList": [
    {
      "ClientReference": "ClientReference",
      "DeliveryAddressName": "Test consignee",
      "ParcelId": 3135399, // use this ID in SetDispatchList request
      "ParcelNumber": 3240118604,
      "PickupAddressName": "Test sender"
    }
  ],
  "ParcelListToDispatchListErrors": []
}
```

SetDispatchList

Request

```
{
  "ClientNumberList": [],
  "Password": [password in byte array],
  "Username": "test@test.com",
  "ClientNumber": 100000004, // use your own ClientNumber
  "ParcelIdList" : [3135399]
}
```

Response

```
{
  "DispatchListErrorList": [],
  "DispatchListReportPdf": [pdf data],
  "FormattedId": "FL-103004227-0000045"
}
```

GetDispatchList

Request

```
{
  "ClientNumberList": [],
  "Password": [password in byte array],
  "Username": "test@test.com",
  "ClientNumber": 100000004, // use your own ClientNumber
  "DateFrom": "/Date(1683669600000+0200)/",
  "DateTo": "/Date(1683669600000+0200)/"
}
```

Response

```
{
  "DispatchListErrorList": [],
  "DispatchListResultItemList": [
    {
      "ClientNumber": 100000004,
      "FormattedId": "FL-103004227-0000045"
    }
  ]
}
```

GetDispatchListReport

Request

```
{
  "ClientNumberList": [],
  "Password": [password in byte array],
  "Username": "test@test.com",
  "ClientNumber": 100000004, // use your own ClientNumber
  "FormattedId": "FL-103004227-0000045"
}
```

Response

```
{
  "DispatchListReportErrorList": [],
  "DispatchListReportPdf": [pdf data]
}
```