LeaveDays – API

Getting started on the LeaveDays REST – API

Description

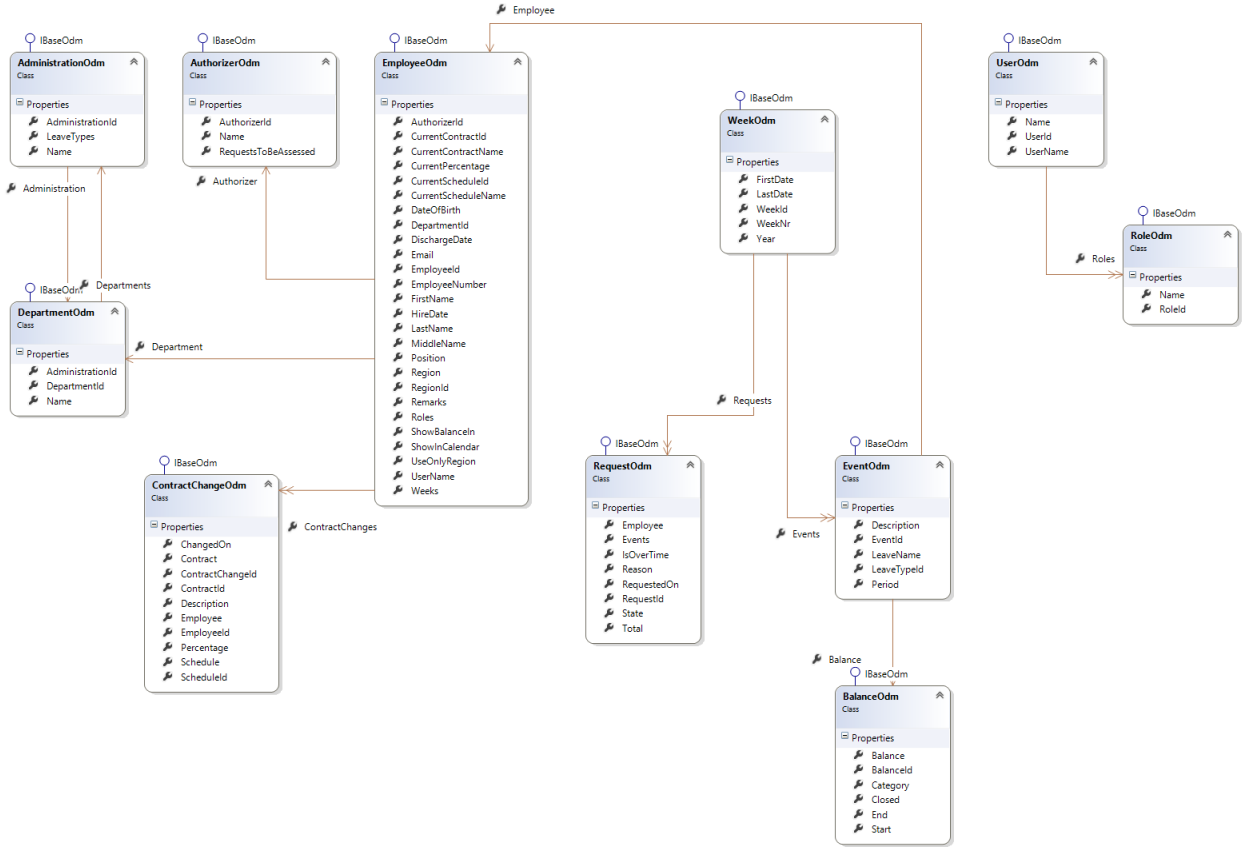
This document describes the currently supported functions and queries of the LeaveDays REST-API implementation and it's protocols. This is a short summary of the full implementation and a work in progress. During the implementation things can change without notice. After release to production the interface will be frozen and versioning will take effect.

Purpose

The purpose of the document is to get started on developing a client implementation that connects to the LeaveDays REST-API and can query data and update resources.

Resources

The available resources in the API at the moment are:



The Odm postfix will be omitted in the rest of this document.

Not all resources are shown in this model.

General Functionality

The model can be queried using much of the Odata v4 protocol.

Create a HTTP GET request with following url to retrieve a list of resources in JSON:

https://<hostname>/odata/<resourcename in plural>

For instance: To retrieve a list of request resources you would use the following syntax:

GET /odata/requests HTTP/1.1
Host: <hostname>
Authorization: Basic a3dhcm5hYXI6d2Vsa29t
Content-Type: application/json
Cache-Control: no-cache

(please add the appropriate headers for https transport if needed)

The authorization header should be created using basic authentication and a user / password combination in the target website.

To retrieve a single instance of a resource you would add an Id to the end of the get request url like so:

GET /odata/requests('<id>')

The id's are normally found by querying the list of resources and use the Id field.

You can the further enhance the output by adding commands using the odata format:

GET /odata/requests('<id>')$expand=Events

or

GET /odata/requests$filter=startswith(Employee, 'SomeName')

Please checkout the Odata format (v4) specification for more detailed descriptions. Also please note that not all of the specification has been implemented so it is a bit of trial and error.

Please note that the resource list names are lowercase and the attribute names (fieldnames) used in the second halve of the request are CamelCased.

An update to a resource is performed by using a POST request:

POST /odata/requests('67!390162')/RequestService.Approve HTTP/1.1
Host: localhost:49199
Authorization: Basic aXNjaHV1cnM6d2Vsa29t
Content-Type: application/json
Cache-Control: no-cache

{ "Comment":"", "ManagerComment":"" }

This will perform the approve function on the request with the mentioned id. Please note that the exact headers will differ from this example.

As a further example this query will retrieve all the events for an employee in the given week:

http://localhost:49199/odata/employees('92!1997')/Weeks('2014-30')

We also support a few singleton resources like:

- activeowner
- activeauthorizer

The singletons will support several queries like PendingRequest and WeekEvents.

Description of the Resources

User

Identifies the users in the system.

Role

The role the user has in the system. Following are the valid values: Employee, Authorizer, Administrator, SystemAdminstrator. (or there Dutch equivalent).

Owner

Every Event has at most one owner. In the current implementation the Owner Id is the same as the Employee ID and can be used to retrieve one or the other.

Employee

The employee makes leave and overtime requests based on the Work schedule and availability of balance(s)

Authorizer

Authorizes the leave request the employee has entered in the system. Every employee has at least one Authorizer connected.

Request

Describes the leave or overtime request. Holds one or more Events . The events are associated with a balance of a specified leave type.

Event

Describes the leave or overtime period requested. The value of the event is the actual number of minutes that will be deducted or added to the balance. This can be different from the timespan between start and end of the period because of lunch breaks or overtime bonus percentage.

Balance

The balance holds all the vents for a specified leave type. Gives the actual balance for all authorized and pending leave and overtime requests.

ContractChange

Every employee should have at least one contract change at the start of the contract. It holds the information on percentage part-time, work schedule and contract type.

Department

The department the employee is assigned to

Administration

The administration the employee is working for

Week

A holder for requests and events to enable easy querying. The id of a week is Year-Weeknr like: 2014-25.

Please send email to support@vrijedagen.nl when further info is needed or it is not working as expected.

Addendum A

Basic Authorization

GET /odata/requests HTTP/1.1
Host: <hostname>
Authorization: Basic a3dhcm5hYXI6d2Vsa29t
Content-Type: application/json
Cache-Control: no-cache

(please add the appropriate headers for https transport if needed)

The authorization header should be created using basic authentication and a user / password combination in the target website.

I have added here an explanation from WikiPedia for creating the basic authorization headers:

Client side[edit]

When the user agent wants to send the server authentication credentials it may use the *Authorization* header.[7]

The *Authorization* header is constructed as follows:[8]

1. Username and password are combined into a string "username:password"
2. The resulting string is then encoded using the RFC2045-MIME variant of Base64, except not limited to 76 char/line[9]
3. The authorization method and a space i.e. "Basic " is then put before the encoded string.

For example, if the user agent uses 'Aladdin' as the username and 'open sesame' as the password then the header is formed as follows:

```
Authorization: Basic QWxhZGRpbjpvcGVuIHNlc2FtZQ==
```