WebUntis JSON-RPC API

an API for accessing WebUntis functionality.

based on JSON-RPC 2.0
(http://www.jsonrpc.org/specification)

Service-URL: http(s)://<SERVER>/WebUntis/jsonrpc.do

requests must use POST
Content-Type should be set to text/plain or application/json

An API request has the general format

```
{"id":"ID","method":"METHOD NAME","params":{PARAMS},"jsonrpc":"2.0"}
```

with
ID   identifies the request, is repeated in the response
METHOD NAME   name of the method
PARAMS   method parameters as described in the section of the method
                            for simple functions PARAMS can be empty

A response has the general format

```
{"jsonrpc":"2.0","id":"ID","result":RESULT}
```

with
ID   request identifier
RESULT   result of the method as described in the section of the method

remarks:
  – character set: utf-8
  – date format: YYYYMMDD
  – time format: HHMM
  – color format: RRGGBB
  – fields in the result may be omitted if empty (e.g. foreColor and backColor will be omitted if
    not set)

Each section defines a method (function) of the API and describes

purpose
method name
needed right for the user
parameters
result

Additionally there are often given examples and sometimes special remarks.

1) Authentication

Authenticate the given user and start a session

method name	authenticate
right	-

mandatory parameter: ?school=SCHULNAME

```
{"id":"ID","method":"authenticate","params":{"user":"ANDROID",
"password":"PASSWORD", "client":"CLIENT"},"jsonrpc":"2.0"}
```

The parameter CLIENT is a unique identifier for the client app. The parameter client will be mandatory in the future.

```
result:
{"jsonrpc":"2.0","id":"ID","result":
{"sessionId":"644AFBF2C1B592B68C6B04938BD26965","personType"=2,"personId"=17}
```

```
personType                          type of person 2 = teacher, 5 = student
personId                            ID of person
```

All other methods require the result from authentication (= sessionId) per cookie (RequestHeader):

```
JSESSIONID=644AFBF2C1B592B68C6B04938BD26965
```

(The alternative method per pathparameter ;jsessionid=644AFBF2C1B592B68C6B04938BD26965 is deprecated due to security reasons and will no longer be supported by WebUntis 2019 and later).

2) Logout

End the session

method name	logout
right	-

```
{"id":"ID","method":"logout","params":{},"jsonrpc":"2.0"}
```

An application should always logout as soon as possible to free system resources on the server.

3) *Request teachers*

Get list of teachers

method name	getTeachers
right	masterdata teachers read for all
parameters	none

```
{"id":"ID","method":"getTeachers","params":{},"jsonrpc":"2.0"}

{"jsonrpc":"2.0","id":"ID","result":[
{"id":1,"name":"Bach","foreName":"Ingeborg","longName":"Bachmann","foreColor":"0
00000","backColor":"000000"},
{"id":2,"name":"Foss","foreName":"Dian","longName":"Fossey","foreColor":"000000"
,"backColor":"000000"},
...]}
```

4) *Request students*

Get list of students

method name	getStudents
right	masterdata students read for all
parameters	none

```
{"id":"ID","method":"getStudents","params":{},"jsonrpc":"2.0"}

{"jsonrpc":"2.0","id":"1","result":[
{"id":1,"key":"1234567","name":"MüllerAle","foreName":"Alexander","longName":"Mü
ller","gender":"male"},
{"id":2,"key":"7654321","name":"SchmidAme","foreName":"Amelie","longName":"Schmi
dt","gender":"female"},
...]}
```

5) *Request base classes*

Get list of Klassen (base class) for schoolyear

method name	getKlassen
right	masterdata Klassen read for all
parameters:	

- schoolyearId   optional, default: current schoolyear

e.g.
```
{"id":"ID","method":"getKlassen","params":
{"schoolyearId":"SCHOOLYEARID"},"jsonrpc":"2.0"}
```

```
{"jsonrpc":"2.0","id":"ID","result":[
{"id":71,"name":"1A","longName":"Klasse
1A","foreColor":"000000","backColor":"000000",did:2},
{"id":72,"name":"1B","longName":"Klasse
1B","foreColor":"000000","backColor":"000000", "teacher1": 17, "teacher2": 21},
...]}
```

6) Request subjects

Get list of subjects

method name	getSubjects
right	
parameters	none
	-

```
{"jsonrpc":"2.0","id":"ID","result":[
{"id":1,"name":"RK","longName":"Kath.Religion","foreColor":"000000","backColor":
"000000"},
{"id":2,"name":"RE","longName":"Evang.
Religion","foreColor":"000000","backColor":"000000"},
...]}
```

7) Request rooms

Get list of rooms

method name	getRooms
right	-
parameters	none

```
{"jsonrpc":"2.0","id":"ID","result":[
{"id":1,"name":"R1A","longName":"1A","foreColor":"000000","backColor":"000000"},
{"id":2,"name":"R1B","longName":"1B","foreColor":"000000","backColor":"000000"},
...]}
```

8) Request departments

Get list of departments

method name	getDepartments
right	-
parameters	none

```
{"jsonrpc":"2.0","id":"ID","result":[
{"id":1,"name":"A1","longName":"AAA1"},
{"id":2,"name":"A2","longName":"AAA2"}
...]}
```

9) Request holidays

Get list of holidays

method name	getHolidays
right	-
parameters	none

```
{"jsonrpc":"2.0","id":"ID","result":[
{"id":44,"name":"Natio","longName":"Nationalfeiertag","startDate":20101026,"endD
ate":20101026},
{"id":42,"name":"Allerheiligen","longName":"Allerheiligen","startDate":20101101,
"endDate":20101101},
...]}
```

10) Request timegrid

Get timegrid

method name	getTimegridUnits
right	-
parameters	none

```
{"jsonrpc":"2.0","id":"ID","result":[
{"day":0,"timeUnits":[{"startTime":800,"endTime":850},
{"startTime":855,"endTime":945},{"startTime":1000,"endTime":1050},
...],
{"day":1,"timeUnits":[{"startTime":800,"endTime":850},
...],
...]}
```

1 = sunday, 2 = monday, ..., 7 = saturday

11) Request status data

Information about lesson types and period codes and their colors

method name	getStatusData
right	-
parameters	none

Response

```
{"jsonrpc":"2.0","id":"req-002",
"result":{
"lstypes":[
{"ls":{"foreColor":"000000","backColor":"ee7f00"}},
{"oh":{"foreColor":"e6e3e1","backColor":"250eee"}},
{"sb":{"foreColor":"000000","backColor":"1feee7"}},
{"bs":{"foreColor":"000000","backColor":"c03b6e"}},
```

```
{"ex":{"foreColor":"000000","backColor":"fdc400"}}
],
"codes":[
{"cancelled":{"foreColor":"000000","backColor":"b1b3b4"}},
{"irregular":{"foreColor":"e3e33b","backColor":"77649a"}}
]}}
```

12) Request current schoolyear

Data for the current schoolyear

method name	getCurrentSchoolyear
right	-
parameters	none

Response

```
{"jsonrpc":"2.0","id":"req-002","result":
[{"id":10,"name":"2010/2011","startDate":20100830,"endDate":20110731}]}
```

13) Request schoolyears

List of all available schoolyears

method name	getSchoolyears
right	-
parameters	none

Response

```
{"jsonrpc":"2.0","id":"req-002","result":
[{"id":10,"name":"2010/2011","startDate":20100830,"endDate":20110731},
{"id":11,"name":"2011/2012","startDate":20110905,"endDate":20120729}]}
```

14) *Request timetable for an element (simple)*

Get timetable for klasse, teacher, student, room, subject

method name	getTimetable
right	timetable view for the requested element
parameters:	
- id	element id, mandatory
- type	type of element, mandatory
	1 = klasse, 2 = teacher, 3 = subject, 4 = room, 5 = student
- startDate	optional, default: actual date
- endDate	optional, default: actual date

Response

„result": array of period objects

period object:

„id":	ID of period
„date":	
„startTime":	
„endTime":	
„kl":	array of klassen ids
„te":	array of teacher ids
„su":	array of subject ids
„ro":	array of room ids
„lstype":	„ls" (lesson) | „oh" (office hour) | „sb" (standby) | „bs" (break supervision) | „ex" (examination)
	omitted if lesson
„code":	„" | „cancelled" | „irregular"
	omitted if empty
„lstext":	text of the lesson, omitted if empty
„statflags":	statistical flags of the lesson, omitted if empty
„activityType":	activity type of the lesson, omitted if empty

e.g.

```
{"id":"req-002","method":"getTimetable","params":
{"id":71,"type":1},"jsonrpc":"2.0"}

{"jsonrpc":"2.0","id":"req-002","result":[
{"id":125043,"date":20110117,"startTime":800,"endTime":850,
"kl":[{"id":71}],"te":[{"id":23}],"su":[{"id":13}],"ro":[{"id":1}]},
{"id":125127,"date":20110117,"startTime":1055,"endTime":1145,
"kl":[{"id":71}],"te":[{"id":41}],"su":[{"id":19}],"ro":[{"id":31}]},
...]}
```

15) Request timetable for an element (customizable)

Get a customizable timetable for klasse, teacher, student, room, subject..

method name	getTimetable
right	timetable view for the requested element
parameters	

- options	a parameter object encapsulating the following fields:
- element	id object encapsulating the following fields:
- id	element id, either the internal id, name, or external key of an element (see „keyType"), mandatory
- type	element type, mandatory 1 = klasse, 2 = teacher, 3 = subject, 4 = room, 5 = student
- keyType	„id", „name", or„externalkey", optional (default: „id")
- startDate	number, format: YYYYMMDD, optional (default: current date)
- endDate	number, format: YYYYMMDD, optional (default: current date)
- onlyBaseTimetable	boolean, returns only the base timetable (without bookings etc.) (default:false)
- showBooking	boolean, returns the period's booking info if available (default: false)
- showInfo	boolean, returns the period information if available (default: false)
- showSubstText	boolean, returns the Untis substitution text if available (default: false)
- showLsText	boolean, returns the text of the period's lesson (default: false)
- showLsNumber	boolean, returns the number of the period's lesson (default: false)
- showStudentgroup	boolean, returns the name(s) of the studentgroup(s) (default: false)
- klasseFields	array, optional, values: „id", „name", „longname", „externalkey"
- roomFields	array, optional, values: „id", „name", „longname", „externalkey"
- subjectFields	array, optional, values: „id", „name", „longname", „externalkey"
- teacherFields	array, optional, values: „id", „name", „longname", „externalkey"

If a *Fields parameter (e.g. „teacherFields") is not set explicitly, the return object will contain the internal id („keyType": „id") of the references (e.g. the teachers of a period) by default.

Response

„result": array of period objects

period object

- id	number, ID of the period
- date	date
- startTime	time
- endTime	time
- lstype	„ls" (lesson) | „oh" (office hour) | „sb" (standby) | „bs" (break supervision) | „ex"(examination), omitted if the type is lesson
- code	„" | „cancelled" | „irregular", omitted if empty
- info	string, period information
- substText	string, Untis substitution text
- lstext	string, text of the period's lesson

- lsnumber	number, number of the period's lesson
- statflags	string, statistical flags
- activityType	string, activity type of the lesson, omitted if empty
- sg	string, name of the period's student group
- bkRemark	string, remark of the period's booking (optional)
- bkText	string, text of the period's booking (optional)
- kl	array of klassen, containing the fields specified in „ klasseFields"
- te	array of teachers, containing the fields specified in „teacherFields"
- su	array of subjects, containing the fields specified in „subjectFields"
- ro	array of rooms, containing the fields specified in „roomFields"

Ex.

```
{"id": "1", "method": "getTimetable", "params": {"options":
  {
     "element": {"id": "1a_ext", "type": 1, "keyType": "externalkey"},
     "showStudentgroup": true,
     "showLsText": true,
     "showLsNumber": true,
     "showInfo": true,
     "klasseFields": ["id", "name", "longname", "externalkey"],
     "teacherFields": ["id", "name"]
  }
}, "jsonrpc": "2.0"}

{"id" : "1", "jsonrpc" : "2.0", "result" [
  {
     "id": 74,
     "date": 20150515,
     "startTime": 855,
     "endTime": 940,
     "kl": [{"id": 1, "name": "1a", "longname": "Klasse 1a (Gauss)",
           "externalkey": "1a_ext"}],
     "te": [{"id": 7, "name": "Nobel"}],
     "su": [{"id": 3}],
     "ro": [{"id": 15}],
     "statflags": "@",
     "lsnumber": 700,
     "lstext": "Sample Lesson Text.",
     "info": "Sample Period Info as entered in WebUntis",
     "sg": "Rel_1a"
  },
  ...
```

16)

(no longer supported, function will be removed)

17) *Request last import time*

Import time of the last lesson/timetable or substitution import from Untis

method name   getLatestImportTime
right   none
parameters   none

Response

```
{"jsonrpc":"2.0","id":"ID","result":TIMESTAMP}

TIMESTAMP               Unix timestamp
```

18) Search personId

Get Id of the person (teacher or student) from the name

method name   getPersonId
right   none
parameters:

- type   type of person, mandatory
                            2 = teacher or 5 = student
- sn   surname, mandatory
- fn   forename, mandatory
- dob   birthdata, mandatory, use 0 if unknown

result: Id of person or 0 if no fitting person found

```
e.g.
Request
{"id":"req-002","method":"getPersonId","params":
{"sn":"Talisker","dob":0,"type":5,"fn":"James"},"jsonrpc":"2.0"}

Response
{"jsonrpc":"2.0","id":"req-002","result":0}
```

19) Request substitutions

Request substitutions for the given date range

method name	getSubstitutions
right	any timetable view
parameters:	

- startDate	mandatory
- endDate	mandatory
- departmentId	mandatory, use 0 for all departments or if not applicable

result
array of substitution objects

Substitution object

- type   type of substitution

cancel	cancellation
subst	teacher substitution
add	additional period
shift	shifted period
rmchg	room change
rmlk	locked period
bs	break supervision
oh	office hour
sb	standby
other	foreign substitutions
free	free period
exam	exam
ac	activity
holi	holiday
stxt	substitution text

- lsid	id of the concerned lesson
- date	
- startTime	
- endTime	
- kl	array of klassen id objects
- te	array of teacher id objects
- su	array of subject id objects
- ro	array of room id objects
- txt	substitution text
- reschedule	reschedule object, used with types cancel and shift
	shift: date and time of the original period that was shifted
	cancel: date and time of the new period

id object
- id   id of element
- name   name of the element
- externalkey   external key of the element (if specified)
- orgid   id of original element, used if the current element replaces
                          (substitutes) this original element
- orgname   name of original element
- orgexternalkey   external key of the original element (if specified)

reschedule object
- date
- startTime
- endTime

e.g.

```
Request
{"id":"req-002","method":"getSubstitutions","params":
{"startDate":20111107,"endDate":20111116,"departmentId":0},"jsonrpc":"2.0"}
Response
{"jsonrpc":"2.0","id":"req-002","result":
[{"type":"cancel","lsid":3029,"date":20111110,"startTime":855,"endTime":945,"kl"
:[{"id":119}],"te":[{"id":7}],"su":[{"id":5}],"ro":[{"id":2}]},
{"type":"add","lsid":3064,"date":20111114,"startTime":1800,"endTime":1900,"kl":
[{"id":122}],"te":[{"id":28}],"su":[{"id":8}],"ro":[]},
{"type":"shift","lsid":3087,"date":20111116,"startTime":1440,"endTime":1530},
{"reschedule":{"date":20111115,"startTime":1155,"endTime":1245}},{"kl":
[{"id":124}],"te":[{"id":3}],"su":[{"id":12}],"ro":[{"id":9}]},
{"type":"cancel","lsid":3087,"date":20111115,"startTime":1155,"endTime":1245},
{"reschedule":{"date":20111116,"startTime":1440,"endTime":1530}},{"kl":
[{"id":124}],"te":[{"id":3}],"su":[{"id":12}],"ro":[{"id":9}]},
{"type":"cancel","lsid":3091,"date":20111115,"startTime":1000,"endTime":1050,"kl
":[{"id":124}],"te":[{"id":5}],"su":[{"id":20}],"ro":[{"id":34}]},
{"type":"subst","lsid":3111,"date":20111115,"startTime":800,"endTime":850,"kl":
[{"id":126}],"te":[{"id":36,"orgid":5}],"su":[{"id":20}],"ro":[{"id":34}]},
{"type":"subst","lsid":3163,"date":20111110,"startTime":1000,"endTime":1050,"kl"
{"type":"rmchg","lsid":3238,"date":20111115,"startTime":1000,"endTime":1050,"txt
":"Raumänderung","kl":[{"id":132},{"id":131}],"te":[{"id":3}],"su":
[{"id":3}],"ro":[{"id":39,"orgid":26}]}]}
```

20) Request classregevents

Request classregevents for the given date range

method name: getClassregEvents
right: classregevents read for all
parameters:

- startDate   mandatory
- endDate   mandatory

result
array of classregevent objects

e.g.
```
Request
{
  "id":"req-002",
  "method":"getClassregEvents",
  "params": {
    "startDate": 20121018,
    "endDate": 20121018
  },
  "jsonrpc":"2.0"
}

Response
{
  "jsonrpc":"2.0",
  "id":"req-002",
  "result": [
    {
      "studentid": "100010",
      "surname": "Oban",
      "forname": "Tom",
      "date": 20121018,
      "subject": "",
      "reason": "",
      "text": "eats during lesson",
      "categoryId": 12                      // Id of the remark category
    }
  ]
}
```

21) Request Exams

Request Exams

method name: getExams
right: examinations read for all
parameters:

- – examTypeId   mandatory
- – startDate   mandatory
- – endDate   mandatory

result
Array of Exam Objects

```
e.g.
Request
{ "id":"1", "method":"getExams", "params":{ "examTypeId":"1",
"startDate":"20140101", "endDate": "20141231" }, "jsonrpc":"2.0" }

Response
{ "jsonrpc": "2.0", "id": "1", "result":
[
{ "id": 1, "classes": [ 1 ], "teachers": [ 4 ], "students": [ 15, 23, 7, 8, 24,
16, 25, 1, 17, 9, 73, 26, 18, 2, 11, 3, 19, 28, 20, 21, 5, 6, 14, 22 ],
"subject": 14, "date": 20140729, "startTime": 1425, "endTime": 1510 } ] }
```

22) Request ExamTypes

Request ExamTypes

method name: getExamTypes
right: examtypes read for all
parameters: none

result
Array of ExamType Objects

```
e.g.
Request
{ "id":"1", "method":"getExamTypes", "params":{}, "jsonrpc":"2.0" }
```

23) Student timetable with absence information

Retrieves the timetable for all students together with absene information for a given daterange.

method name	getTimetableWithAbsences
right	Student absences

parameters

- options	a parameter object encapsulating the following fields:
- startDate	the start of the daterange to retrieve the absences for (YYYYMMDD)
- endDate	the end of the daterange to retrieve the absences for (YYYYMMDD)

result

array of period-with-absence objects

period-with-absence object

- date	date, date of the period
- startTime	time, startTime of the period
- endTime	time, endTime of the period
- studentId	string, external key of the student
- subjectId	string, external key of the period's subject
- teacherIds	array of string, external keys of the period's teachers
- studentGroup	string, the name of the period's studentgroup
- status	string, status of the period (only if the period is not a regular period, possible values: „irregular")
- absenceReason	string, the reason of absence (optional)
- absentTime	number, the number of minutes the student was absent (optional)
- excuseStatus	string, the status of excuse for a (possible) absence (optional)
- user	string, the external key of the user that checked the absences
- checked	boolean, true if the absences were checked in the period
- invalid	boolean, true if the field „notify" is set in the reason of absence, currently only available for Norway and Sweden (optional)

Example request

```
{
  "id": "1",
  "method": "getTimetableWithAbsences",
  "params": {
    "options": {
      "startDate": 20150105,
      "endDate": 20150111
    }
  },
  "jsonrpc": "2.0"
}
```

Example response

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "result": {
    "periodsWithAbsences": [
      {
        "date": 20150105,
        "startTime": 800,
        "endTime": 850,
        "studentId": "stud01_k01_ext",
        "subjectId": "subject01_ext",
        "teacherIds": [
          "teacher01_ext"
        ],
        "studentGroup": "sg01",
        "absenceReason": "Illness",
        "absentTime": 50,
        "user": "admin_ext",
        "checked": true,
        "invalid": true
      },
      ...
```

24) Request remark categories

Request all remark categories

method name: getClassregCategories
right: classregister
parameters: none

result
array of classregcategory objects

e.g.
```
Request
{
  "id":"1",
  "method":"getClassregCategories",
  "jsonrpc":"2.0"
}

Response
{
  "id": "1",
  "jsonrpc": "2.0",
  "result": [
    {
      "id": 1,
      "name": "disturbs",
      "longName": "disturbs"
    },
    {
      "id": 6,
      "name": "late",
      "longName": "late",
      "groupId": 2
    }
  ]
}
```

25) Request groups for remark categories

Request all groups for remark categories

method name: getClassregCategoryGroups
right: classregister
parameters   none

result
array of classregcategorygroup objects

e.g.
```
Request
{
  "id": "1",
  "method": "getClassregCategoryGroups",
  "jsonrpc": "2.0"
}

Response
{
  "jsonrpc": "2.0",
  "id": "1",
  "result": [
    {
      "id": 1,
      "name": "group1"
    },
    {
      "id": 2,
      "name": "group2"
    }
  ]
}
```

26) Request classregevents for a given element

Request all classregevents for a given student or klasse.

method name:	getClassregEvents
right:	classevent
parameters:	
- options	a parameter object encapsulating the following fields:
- startDate	(YYYYMMDD) , mandatory
- endDate	(YYYYMMDD) , mandatory
- element	
- id	id of the element, e.g. „Tobermory"
- type	type of the element, 1 = klasse, 5 = student
- keyType	„id", „name", or „externalkey", (default: „id")

result
array of classregevent objects

e.g.
```
Request
{
  "id": "req-123",
  "method": "getClassregEvents",
  "jsonrpc": "2.0",
  "params": {
    "options": {
      "startDate": 20170814,
      "endDate": 20180729,
      "element": {
        "id": "Tobermory",
        "type": 5,
        "keyType": "name"
      }
    }
  }
}

Response
{
  "jsonrpc": "2.0",
  "id": "req-123",
  "result": [
    {
      "studentid": "",
      "surname": "Tobermory",
      "forname": "",
      "date": 20180709,
      "subject": "",
      "reason": "",
      "text": "",
      "categoryId": 2
    }
  ]
}
```