

WebUntis JSON-RPC API

an API for accessing WebUntis functionality.

based on JSON-RPC 2.0

(<http://groups.google.com/group/json-rpc/web/json-rpc-2-0>)

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

requests must use POST

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:

- Ě character set: utf-8
- Ě date format: YYYYMMDD
- Ě time format: HHMM
- Ě color format: RRGGBB
- Ě 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), either per pathparameter

; jsessionid=644AFBF2C1B592B68C6B04938BD26965

or per cookie (RequestHeader)

JSESSIONID=644AFBF2C1B592B68C6B04938BD26965

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	-
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":"000000","backColor":"000000"},  
{"id":2,"name":"Foss","foreName":"Dian","longName":"Fossey","foreColor":"000000","backColor":"000000"},  
...]}
```

4) Request base classes

Get list of Klassen (base class) for schoolyear

method name	getKlassen
right	-
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"},  
...]}
```

5) 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"
    }
  ]
}
```

6) 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"
    }
  ]
}
```

7) 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"
    }
  ]
}
```

8) 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,
      "endDate": 20101026
    },
    {
      "id": 42,
      "name": "Allerheiligen",
      "longName": "Allerheiligen",
      "startDate": 20101101,
      "endDate": 20101101
    }
  ]
}
```

9) 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

10) 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"}}
]}}
```

11) 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}]}
```

12) 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}]}
```

13) Request timetable for an element

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

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}]},  
...]
```

14) classregister data for period

List of students that should attend a lesson and student absences

method name getStudentsForPeriod
right classregister
parameters:

- „ttid“: ID of timetable period

Response

```
{"jsonrpc":"2.0","id":"ID","result":{"students": [STUDENTS], "periods":  
[PERIODS]}}
```

STUDENTS array of students
"id" ID of student
"longName" longname
"foreName" forename
"absent" true (if student is absent in period) | false
"absences" ABSENCES: absences of student that concern the period

ABSENCES array of absences
"id" ID of absence
"startDate"
"endDate"
"startTime"
"endTime"
"reasonName" shortname of absencereason

PERIODS array of timetable ids
multiple timetable ids if period is part of a block

e.g.

```
{"id":"ID","method":"getStudentsForPeriod","params":{"ttid":124298},  
"jsonrpc":"2.0"}
```

```
{
  "jsonrpc": "2.0",
  "id": "req-002",
  "result": {
    "students": [
      {
        "id": 57,
        "longName": "Seppl",
        "foreName": "",
        "absent": "false"
      },
      {
        "id": 56,
        "longName": "Chef",
        "foreName": "",
        "absent": "true",
        "absences": [
          {
            "id": 420,
            "startDate": 20110119,
            "endDate": 20110121,
            "startTime": 800,
            "endTime": 1840,
            "reasonName": "krank"
          }
        ]
      },
      {
        "id": 155,
        "longName": "Arran",
        "foreName": "Fionna",
        "absent": "false"
      },
      ...
    ],
    "periods": [
      {
        "id": 124298
      }
    ]
  }
}
```

15) 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
-----------	----------------

17) 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
}
```


18) 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
- 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
- orgid id of original element, used if the current element replaces
 (substitutes) this original element

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}]}}]
```

19) 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"}]
```