



Select a spec

default

# plannaplan <sup>1.0</sup>

[ Base URL: 192.168.0.223:1285/ ]

<http://192.168.0.223:1285/v2/api-docs>

Aplikacja do zapisów na zajęcia UAM.

[Terms of service](#)

## Assignments ▼

Assignment is representation of student willing to join given group (lecture or calss). This controller is depreaceted. Use commission instead

<b>GET</b>	<code>/api/v1/assignments/user</code>	Return user current assignemts (from newestt commision). STUDENT Token needs to be provided. This method is deprecated. Use <code>`/api/v1/commisions/user/schedule`</code> instead.
------------	---------------------------------------	--

## Commisions ▼

Commision is representation of student selected assignments at time. All assignments are attached to some commision so we can see current assignments and also browse history of changes for given user

<b>GET</b>	<code>/api/v1/commisions/export/csv</code>	Export acceptes assignmetns as csv file to import to usos. You need to provide ADMIN token in order to get access to this data.
------------	--	---

### Parameters

Try it out

Name

Description

Authorization

string  
(header)

### Responses

Response content type

\*/\*

Code	Description
200	<i>OK</i>
401	<i>Unauthorized</i>
403	<i>Forbidden</i>
404	<i>Not Found</i>

**GET** /api/v1/commissions/user Return list of user all commissions (history of schedules)

Parameters

Try it out

Name	Description
------	-------------

Authorization

string  
(header)

extralInfo Boolean if we want to display extra info about listed groups. Works only if we have groups set top true

boolean  
(query)  
Default value : false

groups Boolean if we want to display wiht commision's group ids

boolean  
(query)  
Default value : false

Responses

Response content type

\*/\*

Code	Description
------	-------------

200  
*OK*

Code	Description
------	-------------

**Example Value**    Model

```
[
  {
    "commisionDate": "string",
    "commiter": {
      "email": "string",
      "id": 0,
      "name": "string",
      "surname": "string"
    },
    "id": 0,
    "owner": {
      "email": "string",
      "id": 0,
      "name": "string",
      "surname": "string"
    }
  }
]
```

401 *Unauthorized*

403 *Forbidden*

404 *Not Found*

**POST**    /api/v1/commissions/user    Create commision with assignents to given groups. If group doesn't exist error will be thrown

**Parameters**

[Try it out](#)

Name	Description
------	-------------

Authorization

string  
(header)

groups    List of groups ids user want to assign to. If group doesnt existit error will be thrown

(body)

**Example Value**    Model

```
[
  0
]
```

Name Description

]

Parameter content type

application/json

id id  
integer(\$int64)  
(path)

Responses

Response content type

\*/\*

Code

Description

200

OK

Example Value Model

"string"

201

Created

401

Unauthorized

403

Forbidden

404

Not Found

GET

/api/v1/commissions/user/{id}

Return list of commissions for given user. To be able to access this data u need to provide DEANERY token

Parameters

Try it out

Name

Description

Name	Description
Authorization string (header)	
groups boolean (query)	Boolean if we want to display wiht commision's group ids  Default value : false
<b>id</b> * required integer(\$int64) (path)	id

Responses

Response content type

\*/\*

Code	Description
200	<p><i>OK</i></p> <p><b>Example Value</b>    Model</p> <pre>[   {     "commisionDate": "string",     "commiter": {       "email": "string",       "id": 0,       "name": "string",       "surname": "string"     },     "id": 0,     "owner": {       "email": "string",       "id": 0,       "name": "string",       "surname": "string"     }   } ]</pre>
401	<i>Unauthorized</i>
403	<i>Forbidden</i>

Code	Description
404	<i>Not Found</i>

**POST** /api/v1/commissions/user/{id} Create commission with assignments to given groups. If group doesn't exist error will be thrown

Parameters

Try it out

Name	Description
------	-------------

Authorization  
string  
(header)

groups List of groups ids user want to assign to. If group doesnt exist error will be thrown  
(body)

Example Value Model

```
[
  0
]
```

Parameter content type

application/json

**id** \* required id  
integer(\$int64)  
(path)

Responses

Response content type

\*/\*

Code	Description
------	-------------

200	OK
-----	----

Example Value Model

"string"

Code	Description
201	<i>Created</i>
401	<i>Unauthorized</i>
403	<i>Forbidden</i>
404	<i>Not Found</i>

**GET** `/api/v1/commissions/user/{id}/schedule` Return given user current assignemts (from newest commision). DEANERY Token needs to be provided.

**Parameters** Try it out

Name	Description
Authorization string (header)	
<b>id</b> * required integer(\$int64) (path)	id

**Responses** Response content type

Code	Description
200	<i>OK</i>

Example Value    Model

Code	Description
	<pre>[   {     "classes": {       "capacity": 0,       "day": 0,       "endTime": "string",       "grNr": 0,       "id": 0,       "isAccepted": true,       "lecturer": "string",       "room": "string",       "takenPlaces": 0,       "time": "string",       "type": "LECTURE"     },     "id": 0,     "lecture": {       "capacity": 0,       "day": 0,       "endTime": "string",       "grNr": 0,       "id": 0,       "isAccepted": true,       "lecturer": "string",       "room": "string",       "takenPlaces": 0     }   } ]</pre>
401	<i>Unauthorized</i>
403	<i>Forbidden</i>
404	<i>Not Found</i>

**GET** /api/v1/commissions/user/assignments Return list of latest user commission assignments. User is recognized via token

**Parameters** Try it out

Name	Description
Authorization	
string	
(header)	

**Responses** Response content type \*/\*



Code	Description
200	<i>OK</i>
	<p><b>Example Value</b> Model</p> <pre>[   {     "day": 0,     "endTime": "string",     "id": 0,     "lecturer": "string",     "name": "string",     "time": "string",     "type": "string"   } ]</pre>
401	<i>Unauthorized</i>
403	<i>Forbidden</i>
404	<i>Not Found</i>

**GET** /api/v1/commissions/user/schedule Return user current assignemts (from newest commission). STUDENT Token needs to be provided.

---

**Parameters** Try it out

Name	Description
Authorization	
string (header)	

---

**Responses** Response content type \*/\*

Code	Description
------	-------------

**Code** **Description**

200 *OK*

**Example Value** Model

```
[
  {
    "classes": {
      "capacity": 0,
      "day": 0,
      "endTime": "string",
      "grNr": 0,
      "id": 0,
      "isAccepted": true,
      "lecturer": "string",
      "room": "string",
      "takenPlaces": 0,
      "time": "string",
      "type": "LECTURE"
    },
    "id": 0,
    "lecture": {
      "capacity": 0,
      "day": 0,
      "endTime": "string",
      "grNr": 0,
      "id": 0,
      "isAccepted": true,
      "lecturer": "string",
      "room": "string",
      "takenPlaces": 0
    }
  }
]
```

**ConfigController** /

**POST** /api/v1/config

**Parameters**

**Name** **Description**

Authorization  
string  
(header)

ticket Ticket for validation.  
string  
(query)

**Responses**

Response content type

**Code** **Description**

200 *OK*

Code	Description
	<p><b>Example Value</b> Model</p> <p>"string"</p>
201	<i>Created</i>
401	<i>Unauthorized</i>
403	<i>Forbidden</i>
404	<i>Not Found</i>

**GET** ~~/api/v1/configurator/admin/tour~~ It will return what tour is currently on. You need to provide admin token. It is deprecated use /configurator/config/tour is instead.

**Warning: Deprecated**

**Parameters** Try it out

Name	Description
Authorization string (header)	

**Responses** Response content type \*/\*

Code	Description
200	<i>OK</i>

Code	Description
	<p><b>Example Value</b> Model</p> <pre>{   "currentTour": "string" }</pre>
401	<i>Unauthorized</i>
403	<i>Forbidden</i>
404	<i>Not Found</i>

**POST** `/api/v1/configurator/config` Imports data to system. To call you need to provide ADMIN token

#### Parameters

[Try it out](#)

Name	Description
Authorization	
string (header)	
file file (formData)	file .xlsx that contains courses and groups with appointed rules
firstTourBegin (formData)	Date when first tour begin in format dd.MM.yyyy
firstTourEnd (formData)	Date when first tour ends in format dd.MM.yyyy
secondTourBegin (formData)	Date when second tour begin in format dd.MM.yyyy
secondTourEnd (formData)	Date when second tour ends in format dd.MM.yyyy

Responses		Response content type
		<input type="text" value="*/"/>
Code	Description	
200	<i>OK</i>	
	<b>Example Value</b> Model	
	<code>"string"</code>	
201	<i>Created</i>	
401	<i>Unauthorized</i>	
403	<i>Forbidden</i>	
404	<i>Not Found</i>	

**POST** /api/v1/configurator/config/courses Imports data to system. To call you need to provide ADMIN token

**Parameters** Try it out

Name	Description
Authorization	
string (header)	
file	file .xlsx that contains courses and groups with appointed rules
file (formData)	

**Responses** Response content type

Code	Description
200	<i>OK</i>
	<p><b>Example Value</b> Model</p> <p>"string"</p>
201	<i>Created</i>
401	<i>Unauthorized</i>
403	<i>Forbidden</i>
404	<i>Not Found</i>

**GET** `/api/v1/configurator/config/tour` It will return what tour is currently on.

**Parameters** [Try it out](#)

Name	Description
Authorization string (header)	

**Responses** Response content type

Code	Description
------	-------------

Code	Description
200	<i>OK</i>
	<p><b>Example Value</b> Model</p> <pre>{   "currentTour": "string" }</pre>
401	<i>Unauthorized</i>
403	<i>Forbidden</i>
404	<i>Not Found</i>

**POST**`/api/v1/configurator/config/tours`

Set tours dates. To call you need to provide ADMIN token

**Parameters**[Try it out](#)

Name	Description
Authorization string ( <i>header</i> )	
firstTourBegin string(\$date-time) ( <i>query</i> )	Date when first tour begin in format dd.MM.yyyy
firstTourEnd string(\$date-time) ( <i>query</i> )	Date when first tour ends in format dd.MM.yyyy
secondTourBegin string(\$date-time) ( <i>query</i> )	Date when second tour begin in format dd.MM.yyyy

Name	Description
secondTourEnd string(\$date-time) (query)	Date when second tour ends in format dd.MM.yyyy

Responses

Response content type

\*/\*

Code	Description
------	-------------

200

OK

Example Value Model

"string"

## Course

201

Endpoints to deal with both.

Created

**GET** /api/v1/courses/all Return all courses

Parameters

Try it out

Name	Description
------	-------------

Authorization

string  
(header)

groups Boolean if you want to have response with associated groups or without

boolean  
(query)

Default value : false

takenPlaces Boolean if we want to have response with information about taken places by other students. Needs to be set groups true first

boolean  
(query)

Default value : false



Responses		Response content type
		<input type="text" value="*/"/>
Code	Description	
200	<i>OK</i>	
	<p><b>Example Value</b>    Model</p> <pre>[   {     "id": 0,     "name": "string",     "symbol": "string"   } ]</pre>	
401	<i>Unauthorized</i>	
403	<i>Forbidden</i>	
404	<i>Not Found</i>	

## DeveloperController All endpoints to use in development time to help app testing ▼

**POST**    /api/v1/developer/algorithm/accept    performAcceptAlgorithm

---

**Parameters** Try it out

---

Name	Description
Authorization string (header)	

Responses		Response content type
		<input type="text" value="*/"/>
Code	Description	
200	<i>OK</i>	
	<b>Example Value</b> Model	
	"string"	
201	<i>Created</i>	
401	<i>Unauthorized</i>	
403	<i>Forbidden</i>	
404	<i>Not Found</i>	

**POST** /api/v1/developer/algorithm/exchange performExchangeAlgorhythm

**Parameters**

Name	Description
Authorization	
string	
(header)	

**Responses**

Code	Description
------	-------------

Code	Description
200	<i>OK</i>
	<p><b>Example Value</b> Model</p> <p><code>"string"</code></p>
201	<i>Created</i>
401	<i>Unauthorized</i>
403	<i>Forbidden</i>
404	<i>Not Found</i>

**POST** /api/v1/developer/tours/first setFirstTour

**Parameters** Try it out

Name	Description
Authorization string (header)	

**Responses** \*/\*

Code	Description
200	<i>OK</i>
	<p><b>Example Value</b> Model</p>

Code	Description
	"string"
201	<i>Created</i>
401	<i>Unauthorized</i>
403	<i>Forbidden</i>
404	<i>Not Found</i>

**POST** /api/v1/developer/tours/no setNoTour

**Parameters** Try it out

Name	Description
Authorization string (header)	

**Responses** \*/\*

Code	Description
200	<i>OK</i>
	<b>Example Value</b> Model
	"string"

Code	Description
201	<i>Created</i>
401	<i>Unauthorized</i>
403	<i>Forbidden</i>
404	<i>Not Found</i>

**POST** /api/v1/developer/tours/second setSecondTour

**Parameters** Try it out

Name	Description
Authorization string (header)	

**Responses** \*/\*

Code	Description
200	<i>OK</i>
	<p><b>Example Value</b> Model</p> <p>"string"</p>
201	<i>Created</i>

**Exchange** Endpoint to exchange with accepted assignments. ∨

**POST** /api/v1/exchanges/exchange Creates exchange offer.

**Parameters**

Try it out

Name	Description
------	-------------

Authorization

string  
(header)

exchangeRequest

(body) Json object that contains assignment to trade and desired group.  
For example: { "assignment": 3, "group":32 }

Example Value Model

```

{
  "additionalProp1": 0,
  "additionalProp2": 0,
  "additionalProp3": 0
}

```

Parameter content type

application/json

**Responses**

Response content type

\*/\*

Code	Description
------	-------------

200 *OK*

Example Value Model

"string"

201 *Created*

401 *Unauthorized*

Code	Description
403	<i>Forbidden</i>
404	<i>Not Found</i>

**GET** /api/v1/exchanges/exchange/{id} Get exchange offer

### Parameters

Try it out

Name	Description
------	-------------

Authorization

string  
(header)

**id** \* required

integer(\$int64)  
(path)

id

### Responses

Response content type

\*/\*

Code	Description
------	-------------

200

*OK*

Example Value Model

```
{
  "courseName": "string",
  "desiredGroup": {
    "day": 0,
    "endTime": "string",
    "grNr": 0,
    "id": 0,
    "isAccepted": true,
    "lecturer": "string",
    "room": "string",
    "takenPlaces": 0,
    "time": "string",
```

Code	Description
	<pre> "type": "LECTURE" }, "id": 0, "ownedAssignment": {   "day": 0,   "endTime": "string",   "grNr": 0,   "id": 0,   "isAccepted": true,   "lecturer": "string",   "room": "string",   "takenPlaces": 0,   "time": "string" } </pre>
401	<i>Unauthorized</i>
403	<i>Forbidden</i>
404	<i>Not Found</i>

**DELETE** /api/v1/exchanges/exchange/{id} Delete exchange offer

#### Parameters

Try it out

Name	Description
Authorization string (header)	
<b>id</b> * required integer(\$int64) (path)	id

#### Responses

Response content type

\*/\*

Code	Description
200	<i>OK</i>



Code	Description
	<p><b>Example Value</b> Model</p> <p>"string"</p>
204	<i>No Content</i>
401	<i>Unauthorized</i>
403	<i>Forbidden</i>

**GET** /api/v1/exchanges/exchange/all Get exchange offers

**Parameters** Try it out

Name	Description
Authorization string (header)	

**Responses** Response content type

Code	Description
200	<p><i>OK</i></p> <p><b>Example Value</b> Model</p> <pre>[   {     "courseName": "string",     "desiredGroup": {       "day": 0,       "endTime": "string",</pre>

Code	Description
------	-------------

```

"grNr": 0,
"id": 0,
"isAccepted": true,
"lecturer": "string",
"room": "string",
"takenPlaces": 0,
"time": "string",
"type": "LECTURE"
},
"id": 0,
"ownedAssignment": {
  "day": 0,
  "endTime": "string",
  "grNr": 0,
  "id": 0,
  "isAccepted": true,
  "lecturer": "string",
  "room": "string",
  "takenPlaces": 0.

```

401 *Unauthorized*

## Group

Endpoints to deal with gorups. Group is related directly to course and can be either class and lecture



403 *Forbidden*

**PUT** /api/v1/groups/{id}/capacity Change capacity of given group. You need to provide DEANERY token to be ale to change capacity

### Parameters

Try it out

Name	Description
------	-------------

Authorization

string  
(header)

id id of group to change capacity

integer(\$int64)  
(path)

**newcapacity** \* required newcapacity

integer(\$int32)  
(query)

### Responses

Response content type

\*/\*

Code	Description
------	-------------

200

*OK***Example Value** Model*"string"*

201

*Created*

401

*Unauthorized*

403

*Forbidden*

404

*Not Found***GET**`/api/v1/groups/course/{id}` Return list of lectures and classes (if present) given course**Parameters****Try it out**

Name	Description
Authorization string (header)	
capacity boolean (query)	Boolean if we want to have capacity field in response <i>Default value</i> : true
<b>id</b> * required integer(\$int64) (path)	id
takenPlaces boolean (query)	Boolean if we want to have response with information about taken places by other students <i>Default value</i> : false

Responses

Response content type

\*/\*

Code	Description
200	<i>OK</i>

Example Value Model

```

{
  "classes": [
    {
      "day": 0,
      "endTime": "string",
      "grNr": 0,
      "id": 0,
      "isAccepted": true,
      "lecturer": "string",
      "room": "string",
      "takenPlaces": 0,
      "time": "string",
      "type": "LECTURE"
    }
  ],
  "lectures": [
    {
      "day": 0,

```

# StatisticsControll

GET /api/v1/statis

Parameters

Try it out

Name

```
"endTime": "string",
"grNr": 0,
"id": 0,
"isAccepted": true,
"lecturer": "string",
"room": "string",
```

Authorization

string  
(header)

Responses

Response content type

\*/\*

Code

Description

200

*OK*

Example Value Model

```
{
  "ammount": 0
}
```

401

*Unauthorized*

403

*Forbidden*

404

*Not Found*

GET

/api/v1/statistics/groups/full getGroupsFullAmmounts

Parameters

Try it out

Name	Description
------	-------------

Authorization  
string  
(header)

**Responses**

Response content type

\*/\*

Code	Description
------	-------------

200

*OK*

**Example Value** Model

```
{
  "ammount": 0
}
```

401

*Unauthorized*

403

*Forbidden*

404

*Not Found*

**GET** /api/v1/statistics/users/accepted getAcceptedAmmounts

**Parameters**

Try it out

Name	Description
------	-------------

Authorization  
string  
(header)

**Responses** Response content type

Code	Description
200	<i>OK</i>
	<p><b>Example Value</b> <small>Model</small></p> <pre>{   "ammount": 0 }</pre>
401	<i>Unauthorized</i>
403	<i>Forbidden</i>
404	<i>Not Found</i>

**GET** `/api/v1/statistics/users/accepted/partly` `getAcceptedPartlyAmmounts`

**Parameters** [Try it out](#)

Name	Description
Authorization string (header)	

**Responses** Response content type

Code	Description
200	<i>OK</i>

Code	Description
	<p><b>Example Value</b> Model</p> <pre>{   "ammount": 0 }</pre>
401	<i>Unauthorized</i>
403	<i>Forbidden</i>
404	<i>Not Found</i>

**GET** /api/v1/statistics/users/noregistered getNonCommisionsAmmounts

**Parameters** Try it out

Name	Description
Authorization string (header)	

**Responses** Response content type \*/\*

Code	Description
200	<i>OK</i>
	<p><b>Example Value</b> Model</p> <pre>{   "ammount": 0 }</pre>



Code	Description
401	<i>Unauthorized</i>
403	<i>Forbidden</i>
404	<i>Not Found</i>

**GET** /api/v1/statistics/users/registered getCommisionsAmmounts

**Parameters** Try it out

Name	Description
Authorization string (header)	

**Responses** Response content type

Code	Description
200	<i>OK</i>

**Token** Endpoints to get authorization. ▼

**Example Value**    Model

**GET** /token Endpoint to token we n { "ammount": 0 }

**Parameters** Try it out

Name	Description
------	-------------

Name	Description
------	-------------

Authorization

string  
(header)

ticket  
string  
(query)

Ticket get from CAS system. It should look like ST-1376572-wo41gty5R0JCZFKMMie2-cas.amu.edu.psl

Responses

Response content type

\*/\*

Code	Description
------	-------------

200

*OK*

Example Value Model

```
{
  "authorityRole": "string",
  "email": "string",
  "id": 0,
  "refreshToken": "string",
  "token": "string"
}
```

401

*Unauthorized*

403

*Forbidden*

404

*Not Found*

**GET**

/token/refresh

Endpoint to access new token based on refresh token. It's needed when request with provided token fail with code 403

Parameters

Try it out

Name	Description
------	-------------

Name	Description
------	-------------

Authorization

string  
(header)

refreshToken

string  
(query)

Refresh token obtained in /token request

Responses

Response content type

\*/\*

Code	Description
------	-------------

200

OK

Example Value Model

```
{
  "authorityRole": "string",
  "email": "string",
  "id": 0,
  "refreshToken": "string",
  "token": "string"
}
```

401

Unauthorized

# Users

Endpoints to deal with users.

403

Forbidden

**POST** /api/v1/users/admin Adds new admin user.

Parameters

Try it out

Name	Description
------	-------------

Authorization

string  
(header)

Name	Description
authority string (query)	USOS ID or E-mail. If user does not exist it should be USOS ID

**Responses**

Response content type

\*/\*

Code	Description
200	<i>OK</i>
	<p><b>Example Value</b> Model</p> <p>"string"</p>
201	<i>Created</i>
401	<i>Unauthorized</i>
403	<i>Forbidden</i>
404	<i>Not Found</i>

**POST** /api/v1/users/deanery Adds new deanery user.

**Parameters**

Try it out

Name	Description
Authorization string (header)	

Name	Description
authority string (query)	USOS ID or E-mail. If user does not exist it should be USOS ID

  

Responses	Response content type
	<input type="text" value="*/"/>

  

Code	Description				
200	<i>OK</i>				
	<table border="0"> <tr> <td><b>Example Value</b></td> <td>Model</td> </tr> <tr> <td>"string"</td> <td></td> </tr> </table>	<b>Example Value</b>	Model	"string"	
<b>Example Value</b>	Model				
"string"					
201	<i>Created</i>				
401	<i>Unauthorized</i>				
403	<i>Forbidden</i>				
404	<i>Not Found</i>				

<b>GET</b>	/api/v1/users/student/search	Search for user by providing query. If query is empty it will return all students. You need token with DEANERY role to call this
<b>Parameters</b>		<input type="button" value="Try it out"/>
Name	Description	

Name	Description
------	-------------

Authorization

string  
(header)

query  
string  
(query)

Query to filter all students. If empty will match everyone

**Responses**

Response content type

\*/\*

Code	Description
------	-------------

200

*OK*

Example Value Model

```
[
  {
    "email": "string",
    "id": 0,
    "name": "string",
    "surname": "string"
  }
]
```

401

*Unauthorized*

403

*Forbidden*

404

*Not Found*

**GET** /api/v1/users/students Gets all students. You need token with DEANERY role to call this

**Parameters**

Try it out

Name	Description
------	-------------

Name	Description
------	-------------

Authorization  
string  
(header)

**Responses**

Response content type

\*/\*

Code	Description
------	-------------

200

OK

Example Value Model

```
[
  {
    "email": "string",
    "id": 0,
    "name": "string",
    "surname": "string"
  }
]
```

**Models**

401	AssignmentDetailedResponse	One of sized
	day	integer(\$int32)
	endTime	string
403	id	integer(\$int64)
	lecturer	Forbidden
	name	string
	time	string
404	type	string
		Not Found

```
AssignmentResponse {
  description: Response shows information about given assignment to course.
  classes: GroupWithCapacityResponse {...}
  id: integer($int64)
  lecture: GroupWithCapacityResponse {...}
  name: string
}
```

```
CommisionResponse {
  description:      Response shows information about commision.
  commisionDate      string
                     Timestamp where the user commit the commision
  commiter            UserResponse {...}
  id                  integer($int64)
                     ID created by database
  owner              UserResponse {...}
}
```

```
ConfigTourResponse {
  currentTour        string
}
```

```
CourseWithGroupsResponse {
  description:      Response shows information about user exchanges
  courseName          string
  desiredGroup        GroupDefaultResponse {...}
  id                  integer($int64)
                     Database id
  ownedAssignment     GroupDefaultResponse {...}
}
```

```
CourseWithGroupsResponse«GroupDefaultResponse» {
  description:      Response shows information about groups to given
                     course.
  classes              [...]
  lectures             [...]
}
```

```
CoursesResponse {
  id                  integer($int64)
  name                string
  symbol              string
}
```



```
GroupDefaultResponse {
  description:      Response shows information about given group.
  day                integer($int32)
                    Value shows what day when the course is. Example 0 is
                    Monday.
  endTime           string
                    Value shows time when the course ends.
  grNr              integer($int32)
  id                integer($int64)
                    ID created by database.
  isAccepted        boolean
                    Used only in resposnes realted to user assignments. For
                    example in /api/v1/users/schedule.
  lecturer          string
                    Value shows degree, name and surname.
  room              string
                    Value shows room where the class takes.
  takenPlaces       integer($int32)
                    Value shows how many places is already taken by other
                    students.
  time              string
                    Value shows time when the course takes.
  type              string
                    Value shows kind of group. The types are LECTURE or
                    CLASS.
                    Enum:
                    Array [ 6 ]
}
```

```
GroupWithCapacityResponse  {
  description:           Response shows information about group with included
                           capacity.
  capacity                integer($int32)
  day                    integer($int32)
                           Value shows what day when the course is. Example 0 is
                           Monday.
  endTime                string
                           Value shows time when the course ends.
  grNr                   integer($int32)
  id                     integer($int64)
                           ID created by database.
  isAccepted             boolean
                           Used only in responses related to user assignments. For
                           example in /api/v1/users/schedule.

  lecturer               string
                           Value shows degree, name and surname.

  room                   string
                           Value shows room where the class takes.
  takenPlaces            integer($int32)
                           Value shows how many places is already taken by other
                           students.
  time                   string
                           Value shows time when the course takes.
  type                   string
                           Value shows kind of group. The types are LECTURE or
                           CLASS.
                           Enum:
                           Array [ 6 ]
}
```

```
StatisticSimpleNumberResponse  {
  amount                 integer($int32)
}
```

```
TokenResponse  {
  description:           Response shows information about logged user.
  authorityRole          string
                           user app role
  email                  string
                           user unviersity email
  id                     integer($int64)
                           user id in database
  refreshToken           string
                           user token needed to refresh
  token                  string
                           user token used to verify requests
}
```

```
UserResponse {  
  description:      Response shows information about user.  
  email              string  
  id                 integer($int64)  
  name               string  
  surname            string  
}
```