

# Java spring server endpoints

V-1.1.2 Hugo Ekstrand

## 1 About

This document contains information about the Java spring servers HTTP endpoints. They can be reached when the server is running on [`http://localhost:8080/`](http://localhost:8080/) locally or on [`http://vattenskoter.fredin.org:8080/`](http://vattenskoter.fredin.org:8080/) from when running on the development server.

## 2 List of mappings

This section contains a list of all head mappings, the topmost different sections in the spring server.

Empty cells indicate empty/null arguments, bodies, or payloads.

Dates are in ISO8601 yyyy-MM-dd'T'HH:mm:ss.SSSXXX.

**OK** = Exists on server now.

**TBD** = Has yet to be implemented (To Be Done).

### 2.1 Head mappings

Mapping	Subsection	Description
/students	2.2	Endpoints for: GET student names, email, and educational moments  POST student educational moments
/instructors	2.3	Endpoint for: GET instructor names
/events	2.4	Endpoints for: GET events POST events

### 2.2 Submappings /students

#### 2.2.1 GET

Status	Type	Mapping	Args	Body	Description
OK	GET	/students		{ name: string, email: string, id: UUID }	Returns a list of all students on the database.

OK	GET	/students/{uuid}	uuid: UUID	{ name: string, email: string, id: UUID }	Returns a specific student, given their unique email.
OK	GET	/students/{uuid}?property={prop}	uuid: UUID, prop: string	{ {prop}: string }	Returns a specific property of a student, such as its name, given the property name and the student's unique email.
OK	GET	/students/{uuid}/moments	uuid: UUID	[{ name: string, description: string, completed: bool }]	Returns a list of all educational moments for a student with a completion status.
OK	GET	/students/{uuid}/moments?moment={moment}	uuid: UUID, moment: string (no blankspaces. Replace blankspace with underscore _)	{ name: string, description: string, completed: bool }	Returns a specific educational moments for a student with a completion status.

## 2.2.2 POST

Status	Type	Mapping	Args	Out type	Description
OK	POST	/students/{uuid}/updatemoment	uuid: UUID,	{ name: string, completed: bool }	Sets the completion status of an educational moment of a student, given the student's unique email, the name of the educational moment being set, and the boolean status (where completed = true).

## 2.3 Submappings /instructor

### 2.3.1 GET

Status	Type	Mapping	Args	Body	Description
OK	GET	/instructors		[{	Returns a list of all

				name: string, id: UUID }]	instructors' names.
--	--	--	--	------------------------------------	---------------------

### 2.3.2 POST

Status	Type	Mapping	Args	Body	Description
--------	------	---------	------	------	-------------

## 2.4 Submapping /events

### 2.4.1 GET

Status	Type	Mapping	Args	Body	Description
OK	GET	/events		{ id: uint, title: string, from: date, to: date, instructors: [ { name: string, id: UUID } ], participants: [ { name: string, email: string, id: UUID } ] }	Returns a list of events stored in the database.
OK	GET	/events?from={from}&to={to}	from: date, to: date	{ id: uint, title: string, from: date, to: date, instructors: [ { name: string, id: UUID } ], participants: [ {	Returns a list of all events stored in the database between two given dates.

				name: string, email: string, id: UUID } ]  }]	
TBD	GET	/events/{id}	id: uint	{ id: uint, title: string, from: date, to: date, instructors: [ { name: string, id: UUID } ], participants: participants: [ { name: string, uuid: UUID } ] } }	Returns the event with the given id.

## 2.4.2 POST

Status	Type	Mapping	Args	Payload	Description
TBD	POST	/events/newsession		[{ title: string from: date, to: date, instructors: [ { name: string, id: UUID } ], participants: participants: [ { name: string, uuid: UUID } ] }]	Creates an event given a JSON payload event.
TBD	POST	/events/{id}/updatesession	id: uint	[{ title: string from: date, 	Overwrites the data of the event with the same id on the

				to: date, instructors: [ { name: string, id: UUID } ], participants: participants: [ { name: string, uuid: UUID } ] }]	database to the new given JSON payload.
OK	POST	/events/{id}/deletesession	id: uint		Deletes the event with the given id from the database.