

Canvas Data Microservice

Setup Local Development Environment

1. Add [Canvas Access Token](#) to the file PrivateToken.txt
2. Install all dependencies using `npm i`
3. Run the project with `npm run start:dev`

Student

Student Canvas Id (user_id)

Gets Canvas Id (user_id) of student which authorization token is used

`http://localhost:{port}/student/selfId` - API Route

Example

`http://localhost:{port}/student/selfId` - returns student's Canvas Id (user_id)

Student Data by Token

Gets information about student which authorization token is used

`http://localhost:{port}/student/self` - API Route

Example

`http://localhost:{port}/student/self` - returns StudentObject

Student Data by Canvas Id (user_id)

Gets information about student with certian ID

`http://localhost:{port}/student/:studentCanvasId` - API Route

Example

`http://localhost:{port}/student/1234` - returns StudentObject with given ID

| Parameter | Type | Description |
|------------------|------|-----------------------------|
| <i>studentId</i> | int | Student Canvas Id (user_id) |

Canvas Courses

List of Canvas Courses

Returns the list of active courses for the student, by his Student Canvas Id (user_id).

`http://localhost:{port}/course/student/:studentId` - API Route

Example

`http://localhost:{port}/course/student/1234` - returns all active courses of student with Student Canvas Id (user_id) 1234

| Parameter | Type | Description |
|------------------|------|-----------------------------|
| <i>studentId</i> | int | Student Canvas Id (user_id) |

Canvas Assignments

List of Canvas Assignments

Returns the list of assignments of the student in the course, by student's Student Canvas Id (user_id) and course id.

`http://localhost:{port}/assignment/course/:courseId/student/:studnetId` - API Route

Example

`http://localhost:{port}/assignment/course/1234/student/1234` - returns all assignments for active course with id 1234 of student with Student Canvas Id (user_id) 1234

| Parameter | Type | Description |
|------------------|------|-----------------------------|
| <i>studnetId</i> | int | Student Canvas Id (user_id) |
| <i>courseId</i> | int | id of the course |

Specific Canvas Assignment

Returns the assignment with certian id from certian course

`http://localhost:{port}/assignment/course/:courseId/assignment/:assignmentId` - API Route

Example

`http://localhost:{port}/assignment/course/1234/assignment/1234` - returns assignment with id 1234 for active course with id 1234

| Parameter | Type | Description |
|---------------------|------|----------------------|
| <i>courseId</i> | int | id of the course |
| <i>assignmentId</i> | int | id of the assignment |

Submission

Specific Submission

Gets information about subbmision with certian Id from certian course and student also specified by Id

`http://localhost:`

`{port}/subbmision/course/:courseId/student/:studentId/assignment/:assignmentsId` - API Route

Example

`http://localhost:{port}/subbmision/course/1234/student/1234/assignment/1234` - returns Submission with 1234 ID from course of 1234 id and student of 1234 id

| Parameter | Type | Description |
|---------------------|------|-----------------------------|
| <i>courseId</i> | int | id of the course |
| <i>assignmentId</i> | int | id of the assignment |
| <i>studnetId</i> | int | Student Canvas Id (user_id) |

Graded Submissions

List of Submissions with corresponding Assignments and max amount of points

Gets submissions objects with assignment and max amount of points for certian student and course defined by Id's

`http://localhost:{port}/submissionObjects/course/:courseId/student/:studentId` - API Route

Example

`http://localhost:{port}/submissionObjects/course/1234/student/1234` - returns array of submissionObjects form course of Id 1234

| Parameter | Type | Description |
|------------------|------|-----------------------------|
| <i>courseId</i> | int | id of the course |
| <i>studnetId</i> | int | Student Canvas Id (user_id) |

Specific Grade

Gets grade from certian student, assignment and course defined by Id

`http://localhost:`

`{port}/grade/gradeObject/course/:courseId/assignment/:assignmentId/student/:studentId` -

API Route

Example

`http://localhost:{port}/grade/gradeObject/course/1234/assignment/1234/student/1234` - returns gradeObject form course of Id 1234, assignment of Id 1234 from student with Id 1234

| Parameter | Type | Description |
|---------------------|------|-----------------------------|
| <i>courseId</i> | int | id of the course |
| <i>assignmentId</i> | int | id of the assignment |
| <i>studentId</i> | int | Student Canvas Id (user_id) |

Specific Submission with max amount of points

Gets submission object with max amount of points from certian student, assignment and course defined by Id

`http://localhost:{port}/grade/submissionObject/course/:courseId/assignment/:assignmentId/student/:studentId` - API Route

Example

`http://localhost:{port}/grade/submissionObject/course/1234/assignment/12/student/1234` - returns submissionObject form course of Id 1234, assignment of Id 1234 from student with Id 1234

| Parameter | Type | Description |
|---------------------|------|-----------------------------|
| <i>courseId</i> | int | id of the course |
| <i>assignmentId</i> | int | id of the assignment |
| <i>studentId</i> | int | Student Canvas Id (user_id) |

Weather Data Microservice

Setup Local Development Environment

1. Install all dependencies using `npm i`
2. Run the project with `npm run start:dev`

Get daily weather

Gets daily weather in Eindhoven for specified period of time.

<http://localhost:7000/dailyWeather?startDate={startDate}&endDate={endDate}> - API Route

| Parameter | Type | Description |
|-------------------------------------|--------|---|
| <i>startDate</i> Required | string | Specifies date of the first daily weather forecast. |
| <i>endDate</i> | string | Specifies date of the last daily weather forecast. Default is the date of sending request. |

Example

<http://localhost:7000/dailyWeather?startDate=2022-04-07&endDate=2022-06-15> - returns daily weather for time period between 2022-04-07 and 2022-06-15 <http://localhost:7000/dailyWeather?startDate=2022-06-07> - returns daily weather for time period between 2022-06-07 and date of sending request

Analyse Microservice

Setup Local Development Environment

1. Install all dependencies using `npm i`
2. Run the project with `npm run start:dev`

Canvas Weather Graph Data

Returns bundled daily weather data and student submissions form selected course in the specified time span by course id. <http://localhost:{port}/graphCanvasWeather/course/:courseId?startDate={startDate}&endDate={endDate}> - API Route

Example <http://localhost:{port}/graphCanvasWeather/course/1234?startDate=2023-01-01&endDate=2023-10-10> - returns bundled daily weather data and student submissions form course with id 1234 for time period between 2023-01-01 and 2023-10-10 <http://localhost:{port}/graphCanvasWeather/course/1234?startDate=2023-01-01> - returns bundled daily weather data and student submissions form course with id 1234 for time period between 2023-01-01 and date of sending request

startDate - **Required**, specifies date of the first daily weather forecast *endDate* - **Default is the date of sending request**, specifies date of the last daily weather forecast

| Parameter | Type | Description |
|-------------------------------------|--------|---|
| <i>courseId</i> | int | Student Canvas Id (user_id) |
| <i>startDate</i> Required | string | Specifies date of the first daily weather forecast. |

| Parameter | Type | Description |
|----------------|--------|--|
| <i>endDate</i> | string | Specifies date of the last daily weather forecast. Default is the date of sending request. |