Description REST-API v2 for Key Performance Indicator

ValueStreamer GmbH
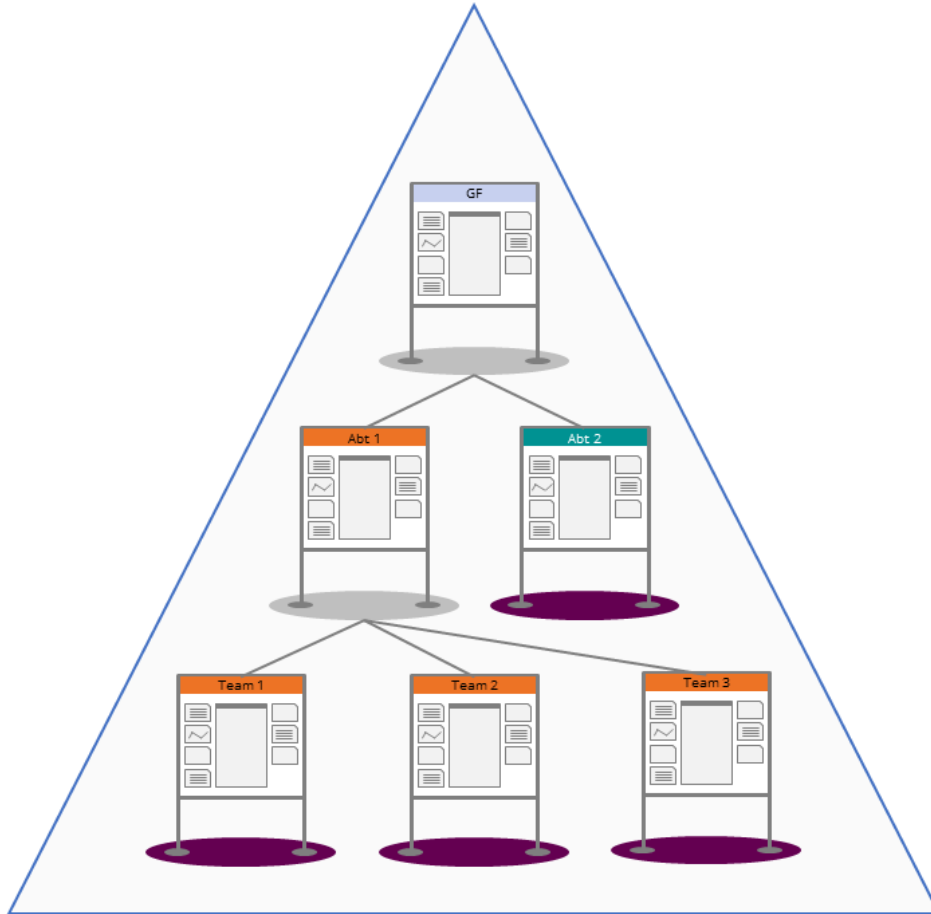
Content

valueStreamer

Example screenshots of KPI charts
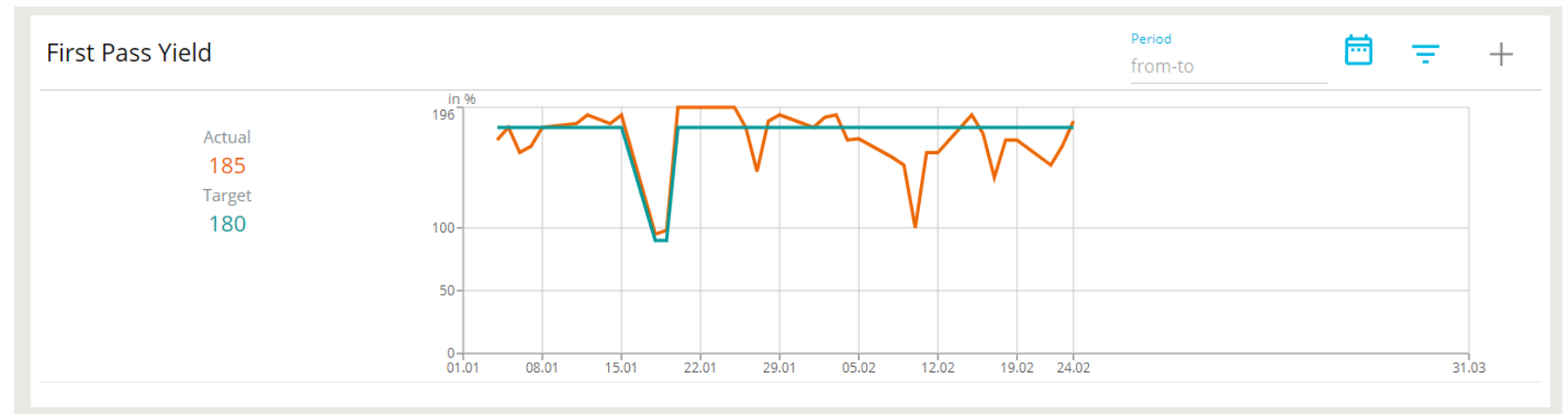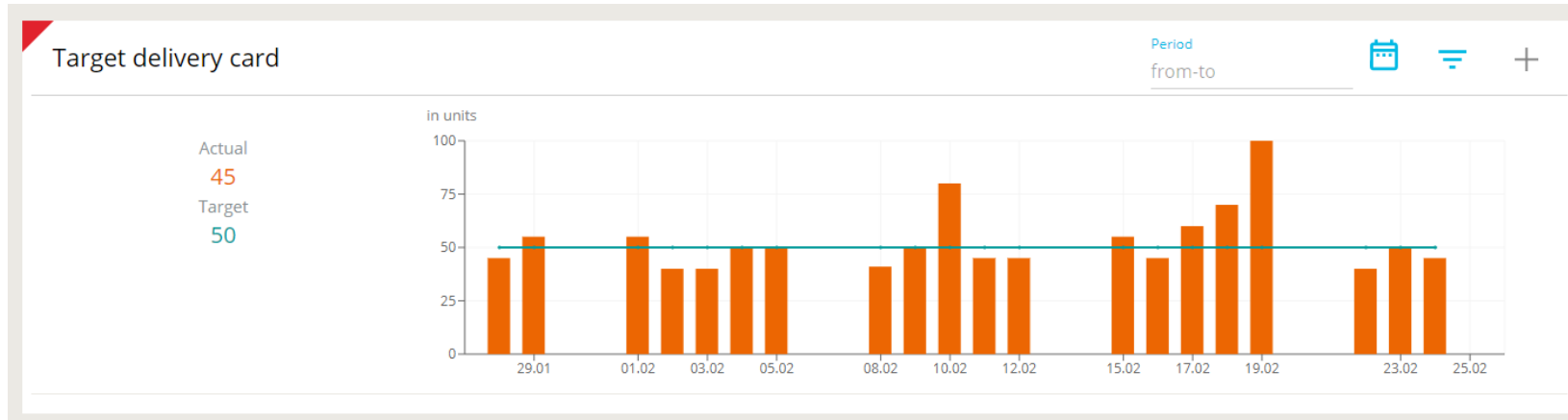
KPI and "Use of the Team Cascade"

Explanations:

- KPIs are defined globally and are available company-wide
- Team-Admin can make the created KPIs available to his team
- Entering of KPI data values:
  - If a KPI is not aggregated, each team can record values for a KPI
  - If a KPI is aggregated, values can only be entered at the lowest available leaf team level, which displays the KPI. The values of the teams above are aggregated/calculated by ValueStreamer
- If a KPI is aggregated, ValueStreamer aggregates/calculates the values from the lowest leaf team level along the team cascade from bottom to top. Parent teams above the lowest leaf team can not enter values.

Details for "Clustering on the lowest available layer" (Sub-tiles)
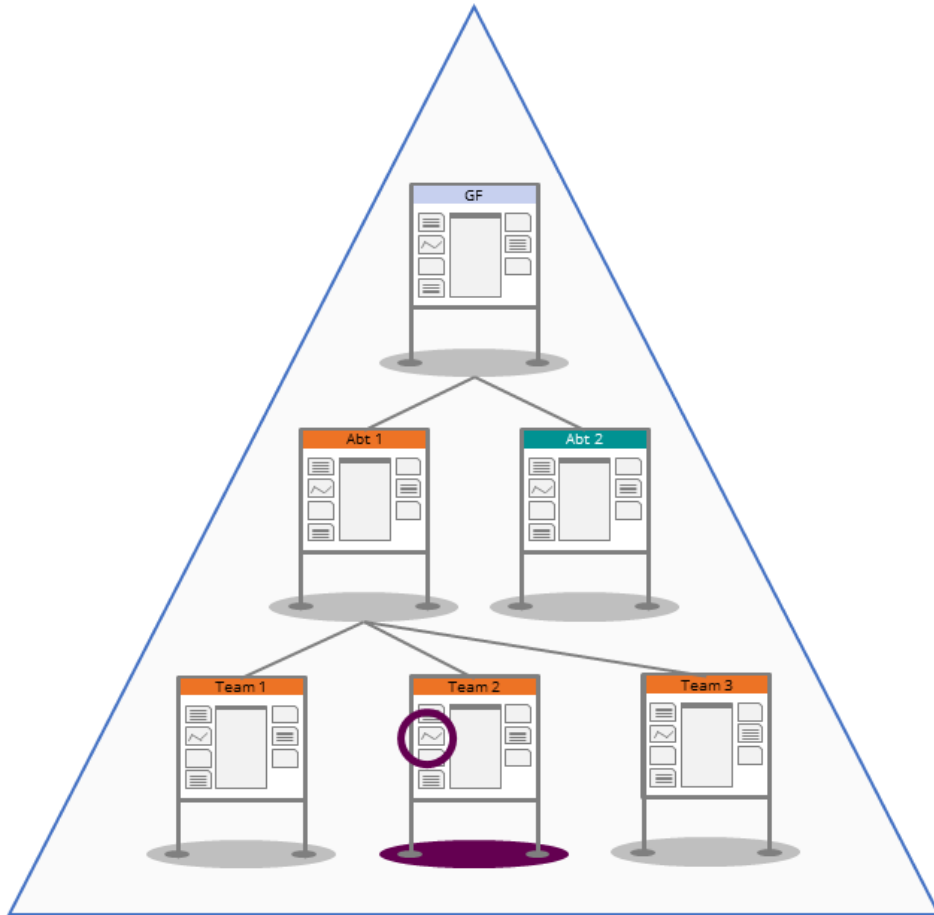
Explanations:

- In addition to the explanations on the slide before, the following points apply
- At the lowest level, sub-tiles can be linked to an added KPI to increase the granularity
- The sub-tiles correspond 1:1 to the attributes (diagram type, aggregation mechanism, ...) of the added KPIs and are permanently linked to them.
- Only the number and title of the sub-tiles are flexible

valueStreamer

OpenAPI 3 specification

- The specification contains all endpoints and json formats in the .yaml file as OpenAPI 3 format
- We recommend viewing the file via Swagger UI / Editor. Advantages: Clear and user-friendly UI with examples (see on the right)
- RESTful API, JSON
- Meta Endpoints:
  - GET: Get the list of all KPI
  - GET: Get the KPI details and recording teams
- Data Endpoints for KPI and Sub-tiles:
  - GET: Get a list of records for a KPI
  - GET: Get a specific record
  - PUT: Create or update a record
 - DELETE: Delete a record
- Authentication via Basic Auth

Meta

| GET | /exchange/kpi | Returns the list of tenant-specific KPI. |
| GET | /exchange/kpi/{kpi-id} | Returns the details of a KPI by UUID. |

KPI

GET	/exchange/kpi-data	Returns a list of filterable data records for a KPI.
GET	/exchange/kpi-data/{date}/{team-id}/{kpi-id}	Query a specific record.
PUT	/exchange/kpi-data/{date}/{team-id}/{kpi-id}	Create or update a record.
DELETE	/exchange/kpi-data/{date}/{team-id}/{kpi-id}	Delete a specific record.

valueStreamer

Example: Create kpi data points for a specific KPI

- **KPI:** First Pass Yield
- **Team:** Demo Team
- **Sub-tile:** -
- **Values:** Actual (177) and target (190)
- **Method:** PUT to create the record with the data:

First Pass Yield

Period
01 Feb 21 - 23 Feb 21 x

in %

Actual
185
Target
180

190

190

177
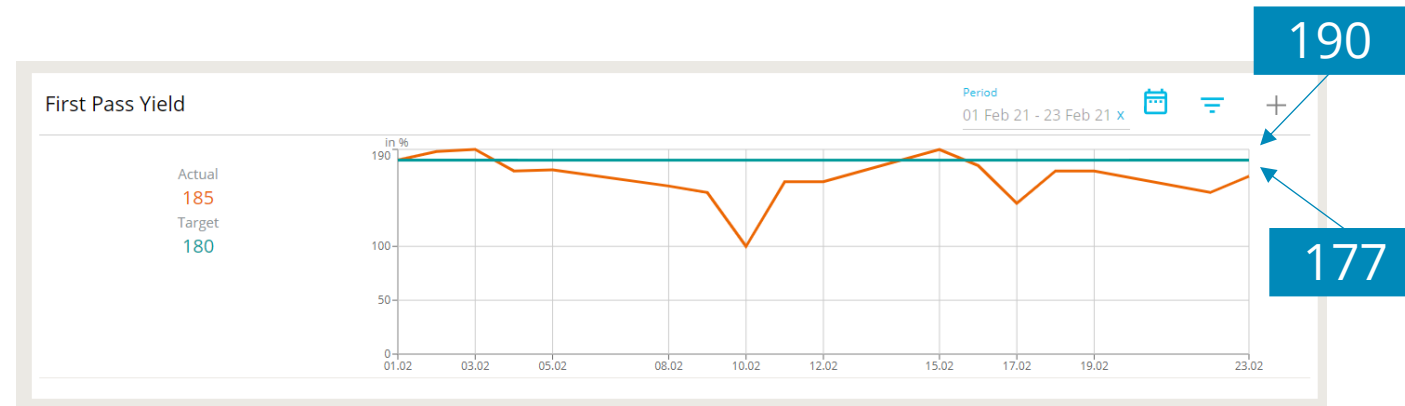
Property	Explanation	Example Actual	Example Target
Date	Date	2020-09-15	2020-09-15
teamId	UUID of the team	8dca904d-9134-3bb0-95b7-3f72ae3dfacf	8dca904d-9134-3bb0-95b7-3f72ae3dfacf
kpiid	UUID of the kpi	c3ba32c2-a13a-0000-0000-000000000000	c3ba32c2-a13a-0000-0000-000000000000
subTiledId	UUID of the sib-tile		
kpiValueId	UUID of the actual or target	79cf2f74-0e9c-45b5-bc4d-f437079dde7c	fd2c7750-79e1-4514-89a2-e9c103db4d6b
value	Value	177	190

HTTP PUT example to create a record:

PUT /api/exchange/kpi-data/2020-09-15/8dca904d-9134-3bb0-95b7-3f72ae3dfacf/c3ba32c2-a13a-0000-0000-00000000000 HTTP/1.1
Host: https://api-tenant.valuestreamer.de
Content-Type: application/vs.v2.0+json
Authorization: Basic <tobeinserted>

```
{
  "values": [
    {"kpiValueId":"79cf2f74-0e9c-45b5-bc4d-f437079dde7c",    "value": 177 },
    {"kpiValueId":"fd2c7750-79e1-4514-89a2-e9c103db4d6b",   "value": 190 }
  ]
}
```
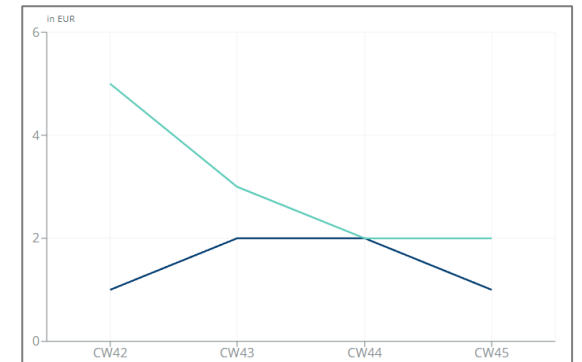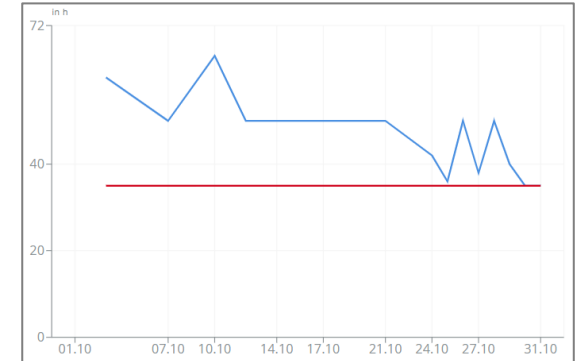
Recording interval (DAY, WEEK)

The KPI editor allows you to set whether the values for the KPI are entered per day, per calendar week or per month. Depending on this, for example, a series of days, calendar weeks or months is displayed on the X-axis. This has the following impact on the KPI API:

Recording interval: DAY

▪ Dates must be send / read in the format yyyy-MM-dd, e.g. 2022-11-01

Recording interval: WEEK

▪ Dates must be send / read in the format yyyy-MM-dd, e.g. 2022-11-01
▪ To create an entry for a calendar week, send the date for Monday
▪ E.g. to create an entry for CW 52/2022, send 2022-12-26
▪ E.g. to create an entry for CW 01/2023, send 2022-01-02

valueStreamer

Recording interval (MONTH)

The KPI editor allows you to set whether the values for the KPI are entered per day, per calendar week or per month. Depending on this, for example, a series of days, calendar weeks or months is displayed on the X-axis. This has the following impact on the KPI API:

Recording interval: MONTH
- Dates must be send / read in the format yyyy-MM-dd, e.g. 2022-11-01
- To create an entry for a month, send the first day of the month
- E.g. to create an entry for January 2022, send 2022-01-01
- E.g. to create an entry for December 2022, send 2022-12-01

valueStreamer
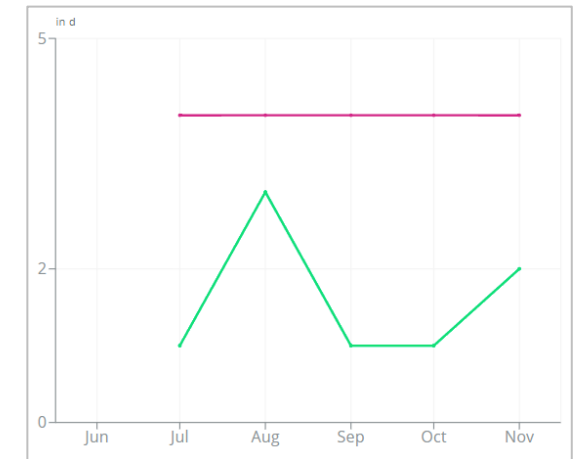
Notes for creating / updating KPI data points (1/2)

If creating/updating data points via PUT endpoint for a specific KPI+team+date at least one kpiValue must be provided in the request. The following cases are possible:

1. A complete set of kpiValues (one or more actual and target values, depending on the KPI) is provided in the request. If a data point already exists for the combination of KPI+team+date, the existing data point will be updated

2. Only a target value for a date > today (forecast) is provided in the request. The „missing" actual value(s) for the combination of KPI+team+date can be updated at a later time with a separate request

valueStreamer

Notes for creating / updating KPI data points (2/2)

3. Not all actual values are provided in the request →The API checks if data exists for the actual values that were not provided in the request for the combination of KPI+team+date:

- If yes, the existing value(s) will not be updated/changed
- If no, the API will create new records with the value „0" for the kpiValues that were not provided in the request

4. Only actual value(s), NOT the target value are provided in the request → The API checks if data exists for the target value that was not provided in the request for the combination of KPI+team+date:

- If yes, the existing value for the target value will not be updated/changed
- If no, the API checks if in the past a data point for the target value exists for the combination of KPI+team and will use that value for the current date. If no target value has been recorded yet, the API will create a data point with the value 0 for the target value that was not provided in the request

Hints for KPIs with unit „Time"

- KPIs that were configured to use the unit „Time" use a specific way of representing the time on the y-axis:

HTTP PUT example to create a record:

PUT /api/exchange/kpi-data/2025-10-15/8dca904d-9134-3bb0-95b7-3f72ae3dfacf/c3ba32c2-a13a-0000-0000-000000000000 HTTP/1.1
Host: https://api-tenant.valuestreamer.de
Content-Type: application/vs.v2.0+json
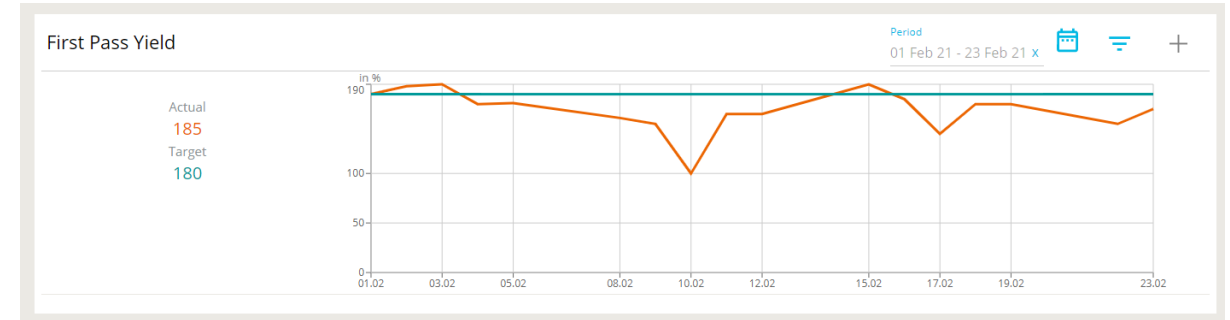Authorization: Basic <tobeinserted>

```
{
  "values": [
    {"kpiValueId":"79cf2f74-0e9c-45b5-bc4d-f437079dde7c",    "value": 883 }]
}
```

- In order to create/update a time datapoint, use the minutes since 0.00 o'clock in your request
- Example: To create a KPI datapoint for 2:43 p.m. (14:43 o'clock in 24h format) use the value: 14*60+43 = **883**

valueStreamer

Example: Get all records for a KPI

- **KPI:** First Pass Yield
- **Team:** Demo Team
- **Endpoint:** GET to /api/exchange/kpi-data returns a list of records
- **From/to:** Depending on whether is recorded per day/week, from/to must not be apart more than 31 days or 6 months
- **Query parameters:**

Property	Explanation	Example
kpi-id	UUID of the kpi (mandatory)	c3ba32c2-a13a-0000-0000-000000000000
from	Date (mandatory)	2020-09-01
to	Date (mandatory)	2020-09-30
team-id	UUID of the team (mandatory)	8dca904d-9134-3bb0-95b7-3f72ae3dfacf
sub-tile-id	UUID of the sub-tile (optional)	

First Pass Yield

Period
01 Feb 21 - 23 Feb 21 x

Actual
185
Target
180

in %
190
100
50
0
01.02   03.02   05.02   08.02   10.02   12.02   15.02   17.02   19.02   23.02

HTTP GET example:

GET /api/exchange/kpi-data?kpi-id=c3ba32c2-a13a-0000-0000-000000000000&team-id=8dca904d-9134-3bb0-95b7-3f72ae3dfacf&from=2020-09-01&to=2020-09-30 HTTP/1.1
Host: https://api-tenant.valuestreamer.de
Content-Type: application/vs.v2.0+json
Authorization: Basic <tobeinserted>

valueStreamer

Technical remarks

- The API consumer can insert or fetch data via REST application interface. The API itself does not push or pull data from or to external systems.

- The customer must exchange ValueStreamer UUIDs when interacting with the REST API, for example when creating a data point for a specific team + kpi + actual value + date combination.

- After the KPIs have been defined and created, the customer can fetch the possible UUID combinations of team + kpi + sub-tile (optional) + actual value + target value on the lowest leaf level in the cascade/hierarchy via REST metadata endpoint. With these combinations, kpi data points can be fetched, inserted or modified.

- The combination of team + kpi + date + sub-tile (optional) per kpi data record is unique. It is only possible to create one record per day for this combination.

- Authorization on the API endpoints works via basic auth. ValueStreamer provides username + password to the customer. When calling the API, the basic auth credentials must be passed base64 encoded in the authorization header like so: Authorization: Basic 123456

- The endpoints to create (PUT), read (GET), update (PUT), delete (DELETE) kpi data records are described using the OpenAPI 3 specification as .yaml file. We recommend to view the file via Swagger UI Preview. The specification describes per endpoint how the requests and responses are defined.