

1. Get translation keys

URL: /get/[apiToken]/[start]/[offset]

Required params: apiToken

Optional params: start, offset

Get all the translation keys for target deployment. If optional parameters [start] and [offset] are set, then it only gets [offset] keys starting key number: [start]

JSON result:

```
{
  deployment: "test.deployment",
  metadata: {
    offset: 6,
    start: 0,
    total: 6
  },
  tks: [
    {
      id: 1833,
      name: "adm_access",
      value: "User",
      deployment: 1,
      language: 22,
      lastChanged: "2013-06-20 12:06:11"
    },
    {
      id: 306883,
      name: "adm_access",
      value: "User",
      deployment: 1,
      language: 15,
      lastChanged: "2013-06-21 16:06:44"
    },
    ...
  ]
}
```

In case of error, the JSON result will contain an error attribute:

```
{
  error: 'invalid apiToken'
}
```

2. Bulk Create/update translation keys

URL: /set/[apiToken]

Required params: apiToken

POST params: file (can be gzipped) of a json encoded array[tkObjects]

each tkObject element must have this attributes: name, value, language, lastChanged.

Create/update translation keys for target deployment.

Example request:

file.json:

```
[
{"name":"adm_access","value":"User","language":15,"lastChanged":"2013-06-19 11:06:25"},
{"name":"adm_accounting_t","value":"Branch","language":15,"lastChanged":"2013-06-19 11:06:25"},
{"name":"adm_accounting_t","value":"Branch2","language":13,"lastChanged":"2013-06-19 11:06:26"},
...
]
```

CURL:

```
curl -X POST --data-binary @file.json(.gz) http://tks.dev-emiliano.lan/set/[apiToken]
```

JSON result:

```
{
  updates: 25,
  inserts: 122
}
```

In case of error, the JSON result will contain an error attribute:

```
{
  error: 'invalid json parameters'
}
```

3. Get deployment stats

URL: /stats/[apiToken]

Required params: apiToken

Get stats for target deployment.

JSON result:

```
{
  deployment: "vw.dw",
  totalKeys: 4110,
  lastChanged: "2013-02-03 13:12:00"
  danish: 2330,
  english: 1000,
  german: 980,
  icelandic: 0,
  dutch: 0
}
```

In case of error, the JSON result will contain an error attribute:

```
{
  error: 'invalid apiToken'
}
```