TrainingApplication - Mobile API's

Conventions

Result codes

All GET requests supports the following result codes:

- 200 OK
- 401 Authorization required (invalid/expired or missing ticket)
- 403 Access denied
- 404 Not found (invalid id or url)
- 422 Invalid according to business process

Any other result codes specific for a request is described for each request.

Parameters

Parameters in the Url are marked as <parameter-name>. The following common Url parameters are used: <country> Can have one of the values "se", "no" or "fi". (note that this project assignment uses only se)

Some requests use query string parameters. These are URL-encoded and sent as individual query string parameters. I.e. "/path?param1=x&param2=y".

Parameters in POST and PUT requests are sent as standard json encoded data in the body part of the POST message.

Result

Results are sent as standard json with the described structure. All varying field content is marked as <varying content>. Strings are assumed for everything enclosed in citation. The following other types are used:

<int>	Integer
<float>	Float
<boolean>	Boolean
<datetime>	Date and time, Example 2012-12-11T12:08:00+01:00

Unknown fields in the json data should be silently ignored without creating any error. This is important for compatibility between versions.

API Endpoints

Class types and categories

List Url: https://api2.sats.com/v1.0/se/classtypes
Method: GET
Result:

```
{
url : "https:\/\/api2.sats.com\/v1.0\/se\/classTypes",
userId : "<>",
classTypes: [
        {
                id: "<id-1>",
                name: "<classtype 1>",
                videoUrl: "<classtype url>",
                description: "<description text>",
                classCategories: ["<cat-id-1>", "<cat-id-2>", "<cat-id-3>"],
                profile: [
                        {id: "<id-1>", name: "Kondition", value: <int>}
                        {id: "<id-2>", name: "Styrka", value: <int>}
                        {id: "<id-3>", name: "Rörlighet", value: <int>}
                        {id: "<id-4>", name: "Balans", value: <int>}
                        {id: "<id-5>", name: "Spänst", value: <int>}
                ] },
        {
                id: "<id-2>",
                name: "<classtype 2>",
                ...
        },
        ...
        ],
        classCategories: [
        {
                id: "cat-id-1",
                name: "<class cat 1>",
        },
        {
                id: "cat-id-2",
                name: "<class cat 2>",
        }, ...
        ]
}
```

Field Name	Description	API Relations
ClassType.id	ClassType.id (PK)	
ClassType.name	Name of type	
ClassType.videoUrl	Url	
ClassType.description	Description	
ClassType.classCategories	Array of class categories (ref to ClassCategory.id)	
ClassType.profile	Klassens profil/fokus, Integer 0-100	
ClassCategory.id	ClassCategory.id (PK)	
ClassCategory.name	Category name	

Centers and regions

List
Url: https://api2.sats.com/v1.0/se/centers
Method: GET
Result:

```
{ "regions" : [
            {
            "id" : "<region-id-1>",
            "name" : "<region 1>",
            "centers" : [ { "availableForOnlineBooking" : <boolean>,
                        "description" : "<string>",
                        "filterId" : "<filter-id-1>",
                        "id" : "<id-1>",
                        "lat" : <float>,
                        "long" : <float>,
                        "name" : "<center 1>",
                        "regionId" : "<region-id-1>",
                        "url" : "<center 1 page url>"
            } ],
      } ],
  "url" : "https://api2.sats.com/v1.0/se/centers",
  "userId" : "506p54221"
}
```

Info
Url: https://api2.sats.com/v1.0/se/centers/<id>
Method: GET
Result:

```
{
            "url" : "https://api2.sats.com/v1.0/se/centers",
            "userId" : "506p54221"
            "centers" : {
                        "availableForOnlineBooking" : <boolean>,
                        "description" : "<string>",
                        "filterId" : "<filter-id-1>",
                        "id" : "<id-1>",
                        "lat" : <float>,
                        "long" : <float>,
                        "name" : "<center 1>",
                        "regionId" : "<region-id-1>",
                        "url" : "<center 1 page url>"
            },
}
```

Field Name	Description	API Relations
Region.id	Region.id (PK)	
Region.name	Name of region	
Region.Center.availableForOnlineBooking	Online booking possibility	
Region.Center.description	Description	
Region.Center.id	Center Id (PK)	
Region.Center.lat	Coordinate latitude	
Region.Center.long	Coordinate longitude	
Region.Center.name	Name of region	
Region.Center.regionId	Region.id (FK)	
Region.Center.url	Url	
Region.Center.isElixia	Not to be used	

Instructors

Url: https://api2.sats.com/v1.0/se/instructors
Method: GET
Result:

```
{ "instructors" : [
            {
                        "id" : "<id-1>",
                        "name" : "<instructor 1>"
            },
            {
                        "id" : "<id-2>",
                        "name" : "<instructor 2>"
            }
   ],
  "url" : "https://api2.sats.com/v1.0/se/instructors",
  "userId" : ""
}
```

Field Name	Description	API Relations
Instructor.id	ID (PK)	
Instructor.name	Name of instructor	

Training activities

List all training activities for the current user. This api endpoint does not exist.
Url: /<country>/training/activities/<fromDate>,<toDate>
Method: GET
Url parameters:

Name	Value	Mandatory
fromDate	yyyymmdd	Yes
toDate	yyyymmdd	Yes

Result:

```
{ "activities" : [ {
        "booking" : #booking(id = “<booking-id>”),
        "comment" : "<a training comment>",
        "date" : <datetime>,
        "distanceInKm":<int>,
        "durationInMinutes":<int>,
        "id" : “<id-1>”,
        "source" : SATS|OTHER|RUNKEEPER,
        "status" : PLANNED|COMPLETED,
        "subType" : "<training-type-id>",
        "type" : GROUP|GYM|OTHER
    }
  ]
}
```

Field Name	Description	API Relations
Activity.Booking	Booking object (nullable).	
Activity.comment	Activity comment	
Activity.date	Date and time for activity	
Activity.durationInKm	Distance in kilometers	
Activity.durationInMinutes	Duration of the activity in minutes	
Activity.id	Activity.id (PK)	
Activity.source	Source of the activity	
Activity.status	Planned or completed activity	
Activity.type	Type of activity; Gym = Styrketräning, Group is Gruppträning (bookings) and OTHERS is for example CrossTraining.	
Activity.subType	Training type id	/activities/types

Booking object

```
#booking(id) = {
        status: “queued|confirmed|done|other”,
        class: #class(id = “<class-id-1>”),
        center: “<class-type-1>”,
        id: “<booking-id-1>” ,
        positionInQueue: <int>
}
```

Field Name	Description	API Relations
Booking.id	Booking class id (PK)	
Booking.class	Class Object	
Booking.center	Center.id (FK)	/centers
Booking.status	Status of the booking; queued, confirmed, done or other	

Class object

```
#class(id) = {
        "centerId" : "<center-id-1>",
        "centerFilterId":"<center-filter-id-1>",
        "classTypeId" : "<class-type-id-1>",
        "durationInMinutes" : <int>,
        "id" : "id-1",
        "instructorId" : "<instructor-id-1>",
        "name" : "<class-1>",
        "startTime" : "<datetime>",
        "bookedPersonsCount" : <int>,
        "maxPersonsCount": <int>,
        "regionId":"<region-id-1>",
        "waitingListCount":<int>
        "classCategoryIds" : [ "<class-category-id-1>","<class-category-id-2>"]
}
```

Field Name	Description	API Relations
centerId	Center.id (FK)	/centers
classTypeId	ClassType.id (FK)	/classtypes
durationInMinutes	Length of class	
id	Class.id (PK)	
instructorId	Intructor.id (FK)	/instructors
name	Name of the class	
startTime	Time and date of class	
bookedPersonsCount	Count of bookings	
centerFilterId	Not used	
maxPersonsCount	Max persons in class	
waitingListCount	Number of users on waiting list	

Types

This api endpoint does not exist.

Url: /<country>/training/activities/types
Method: GET
Result:

```
{
"url": "https:\/\/api2.sats.com\/v1.0\/se\/training\/activities\/types",
"userId":"",
"types" : [
        {"name:"<Name of type>", "subType":"<training-type-id>", "type": GROUP|GYM|OTHER },
        ....
    ]
}
```