External Management

The TracTrac information system provides a set of web services that are used to manage the system from a third party application. All these services use the event manager authentication to avoid invalid accesses.

One event is managed in one server. At this moment the servers are not connected between them: each server has its own list of events and users. We are going to change this behaviour in a future centralizing the management of events/users in only one url.

If the user has a valid account in one server he can start to use the external services in this server. The base-url is the url of the event manager that always is composed by the name of the server with the prefix tracms:

Server	Base URL
traclive.dk	http://tracms.traclive.dk
secondary.traclive.dk	http://tracms.secondary.traclive.dk
skitrac.traclive.dk	http://tracms.skitrac.traclive.dk
tracdev.dk	http://tracms.tracdev.dk

All the services has two parameters to manage the authentication:

- username: a valid username in the event manager
- password: the password

The output of all the services is always a JSON with the attributes status and message.

```
{
   status: "",
   message: ""
}
```

The attribute status has one of these values:

Value	Description	Example
DONE	The request has been executed successfully	`{ status: "DONE", message: "Your credentials are valid" }`
FAILURE	The request has an error	`{ status: "FAILURE", message: "You do not have access to this service" }`

And the attribute message is a free text that gives human-readable information about the execution of the service (it is typically used to describe the errors).

With the base URL and a valid account, it is possible to access to the next services:

Validate credentials

This service is used to validate if a user has a valid credentials to use the external services. It can be called using a HTTP GET request adding the suffix validate_credentials to the base URL:

Server	URL
traclive.dk	http://tracms.traclive.dk/validate_credentials
secondary.traclive.dk	http://tracms.secondary.traclive.dk/validate_credentials
skitrac.traclive.dk	http://tracms.skitrac.traclive.dk/validate_credentials
tracdev.dk	http://tracms.tracdev.dk/validate_credentials

It has to be called using two parameters:

Parameter	Description	Required
username	A valid event manager user name	YES
password	The password of the user	YES

And the possible output values for the attribute status are:

Status	Description	Example

DONE	The user name exists and the password is correct	{
		status: "DONE",
		message: "Your credentials are valid"
		}

FAILURE	The user name doesn't exist or the password is not valid	{
		status: "FAILURE",
		message: "You do not have access to this service"
		}

Course update

This service is used in order to change the course/route of a race. It can be called using a HTTP POST request adding the suffix update_course to the base URL:

Server	URL
traclive.dk	http://tracms.traclive.dk/update_course
secondary.traclive.dk	http://tracms.secondary.traclive.dk/update_course
skitrac.traclive.dk	http://tracms.skitrac.traclive.dk/update_course
tracdev.dk	http://tracms.tracdev.dk/update_course

It has to be called using the parameters:

Parameter	Description	Required
username	A valid event manager user name	YES
password	The password of the user	YES
eventid	A valid event identifier	YES
raceid	A valid race identifier	YES

This request also requires a JSON with the course configuration in the body of the request. An example of

request is:


```
{

    "course": {
        "name": "Course",
        "waypoints": [
            {
                "name": "ControlPointWithTwoMarks White P / White S",
                "passingInstruction": "Line",
                "controlPoint": {
                    "@class": "ControlPointWithTwoMarks",
                    "name": "Start Line",
                    "left": {
                        "@class": "Mark",
                        "name": "White P",
                        "id": "56a44d00-3b61-0132-d536-14dae9614d7b",
                        "color": "White",
                        "shape": "Conical",
                        "type": "BUOY"
                    },
                    "right": {
                        "@class": "Mark",
                        "name": "White S",
                        "id": "60610110-3b61-0132-d538-14dae9614d7b",
                        "color": "White",
                        "shape": "Conical",
                        "type": "BUOY"
                    }
                }
            },
            {
                "name": "Orange",
                "passingInstruction": "Port",
                "controlPoint": {
                    "@class": "Mark",
                    "name": "Orange",
                    "id": "a0c59f10-3be4-0132-b7b5-14dae9614d7b",
                    "color": "Orange",
                    "shape": "Conical",
                    "type": "BUOY"
                }
            },
            {
                "name": "ControlPointWithTwoMarks White P / Black Conical",
                "passingInstruction": "Gate",
                "controlPoint": {
                    "@class": "ControlPointWithTwoMarks",
                    "name": "Finish Gate",
                    "left": {
                        "@class": "Mark",
```

```
                        "name": "White P",
                        "id": "1a210de0-3be4-0132-b7a1-14dae9614d7b",
                        "color": "White",
                        "shape": "Conical",
                        "type": "BUOY"
                    },
                    "right": {
                        "@class": "Mark",
                        "name": "Black Conical",
                        "id": "94f92760-3be4-0132-b7b2-14dae9614d7b",
                        "color": "Black",
                        "pattern": "Checkered",
                        "shape": "Conical",
                        "type": "BUOY"
                    }
                }
            }
        ]
    }
}
```


And the possible output values for the attribute status are:

Status	Description	Example
DONE	The request has been successfully executed and it is registered in the system. It will be applied when the event administrator decides to accept the course.	{ status: "DONE", message: "File parsed successfully" }
FAILURE	The course can not be updated	{ status: "FAILURE", message: ""Physical device not found for the control point 56a44d00-3b61-0132-d536-14dae9614d7b" }

Race start time

This service is used to change the race start time of a race. It can be called using a HTTP GET request adding the suffix update_race_start_time to the base URL:

Server	URL
traclive.dk	http://tracms.traclive.dk/update_race_start_time
secondary.traclive.dk	http://tracms.secondary.traclive.dk/update_race_start_time
skitrac.traclive.dk	http://tracms.skitrac.traclive.dk/update_race_start_time
tracdev.dk	http://tracms.tracdev.dk/update_race_start_time

It has to be called using the parameters:

Parameter	Description	Required
username	A valid event manager user name	YES
password	The password of the user	YES
eventid	A valid event identifier	YES
raceid	A valid race identifier	YES
race_start_time	The race start time, formatted as the number of milliseconds since January 1, 1970, 00:00:00 UTC.	YES

And the possible output values for the attribute status are:

Status	Description	Example

DONE	The race start time has been successfully updated	{
		status: "DONE",
		message: "Race start time changed successfully"
		}

FAILURE	The race start time can not be updated	{
		status: "FAILURE",
		message: "The value for the race start time 2014-11-20 09:00:00 UTC is not included in the event interval [2014-10-29 00:00:00 UTC,2014-10-29 23:00:00 UTC]"
		}

Abandon race

This service is used to abandon a race that has the tracking enabled. It can be called using a HTTP GET request adding the suffix abandon_race to the base URL:

Server	URL
traclive.dk	http://tracms.traclive.dk/abandon_race
secondary.traclive.dk	http://tracms.secondary.traclive.dk/abandon_race
skitrac.traclive.dk	http://tracms.skitrac.traclive.dk/abandon_race
tracdev.dk	http://tracms.tracdev.dk/abandon_race

It has to be called using the parameters:

Parameter	Description	Required
username	A valid event manager user name	YES
password	The password of the user	YES
eventid	A valid event identifier	YES
raceid	A valid race identifier	YES

And the possible output values for the attribute status are:

Status	Description	Example

DONE	The race has been abandoned	{
		status: "DONE",
		message: "Race abandoned"
		}

FAILURE	The race can not be abandoned	{
		status: "FAILURE",
		message: "The race 8b48e30-4185-0132-4048-14dae9614d7b doesn't exist"
		}

Update race status

This service is used to change the race status. It can be called using a HTTP GET request adding the suffix update_race_status to the base URL:

Server	URL
traclive.dk	http://tracms.traclive.dk/update_race_status
secondary.traclive.dk	http://tracms.secondary.traclive.dk/update_race_status
skitrac.traclive.dk	http://tracms.skitrac.traclive.dk/update_race_status
tracdev.dk	http://tracms.tracdev.dk/update_race_status

It has to be called using the parameters:

Parameter	Description	Required
username	A valid event manager user name	YES
password	The password of the user	YES
eventid	A valid event identifier	YES
raceid	A valid race identifier	YES
race_status	The race status, that can be ABANDONED (the race start time is removed) or POSTPONED (the race start time is removed and the tracking is stopped)	YES

And the possible output values for the attribute status are:

Status	Description	Example

DONE	The race has been abandoned	{
		status: "DONE",
		message: "The status of the race has been changed successfully"
		}

FAILURE	The race can not be abandoned	{
		status: "FAILURE",
		message: "The values for the race_status parameters only can be ABANDONED or POSTPONED"
		}