**API SPECIFICATION**

**for**

**TDS Administration**

Authored by

American Institutes for Research

Revision History

|  |  |  |
| --- | --- | --- |
| Revision Description | Author/Modifier | Date |
| 0.1 Initial Draft | Shajib Khan | 25-Apr-2016 |

Table of Contents

[API Documentation Conventions 4](#_Toc449448397)

[Authentication 4](#_Toc449448398)

[TDS Administration API Overview 6](#_Toc449448399)

[TDS Administration Overview 6](#_Toc449448400)

[TDS Administration API Summary 6](#_Toc449448401)

[Authorization 6](#_Toc449448402)

[REST APIs 6](#_Toc449448403)

[/getOpportunities?{keys}={values} (GET) 8](#_Toc449448404)

[Overview 8](#_Toc449448405)

[HTTP Request [Public API] 8](#_Toc449448406)

[Request Parameters 8](#_Toc449448407)

[Request Body 8](#_Toc449448408)

[Response 8](#_Toc449448409)

[Examples 9](#_Toc449448410)

[/resetOpportunity (POST) 10](#_Toc449448411)

[Overview 10](#_Toc449448412)

[HTTP Request [Public API] 10](#_Toc449448413)

[Path Parameters 10](#_Toc449448414)

[Request Parameters 10](#_Toc449448415)

[Request Header 10](#_Toc449448416)

[Request Body 10](#_Toc449448417)

[Response 10](#_Toc449448418)

[Examples 11](#_Toc449448419)

[/invalidateOpportunity (POST) 12](#_Toc449448420)

[Overview 12](#_Toc449448421)

[HTTP Request [Public API] 12](#_Toc449448422)

[Path Parameters 12](#_Toc449448423)

[Request Parameters 12](#_Toc449448424)

[Request Header 12](#_Toc449448425)

[Request Body 12](#_Toc449448426)

[Response 12](#_Toc449448427)

[Examples 13](#_Toc449448428)

**Tables**

[Table 1. HTTP Response Codes 6](#_Toc442363863)

# API Documentation Conventions

The API endpoints are described as follows:

METHOD /endpointname/{pathparameter}?requestparameter1={value1}&requestparameter2={value2}&...

* METHOD indicates the specific HTTP request method: GET, POST, PUT or DELETE
* endpointname is the name used to invoke this API endpoint, assumed to follow a base URL dependent on the particular deployment of the component (for example: https://smarterbalanced.org/permissions/rest/endpointname)
* requestparameter provides parameterized information to this API endpoint
  + *Italicized* parameter names and values are optional
  + {curly brackets} denotes variables
  + Additional Request Parameters are delimited by an ampersand (&)
* pathparameter - Path Parameters are the variable parts of the path, and are documented separately from Request Parameters.

The following table documents the possible response codes for these APIs.

| **HTTP Response Code** | **Meaning** |
| --- | --- |
| 200 | (“OK”) Used to indicate nonspecific success. |
| 201 | (“Created”) Used to indicate successful resource creation |
| 202 | (“Accepted”) Used to indicate successful start of an asynchronous action |
| 204 | (“No Content”) Used when the response body is intentionally empty |
| 301 | (“Moved Permanently”) Used to indicate the resource has been relocated |
| 303 | (“See Other”) Used to refer the client to a different URI |
| 304 | (“Not Modified”) Used to preserve bandwidth |
| 307 | (“Temporary Redirect”) Used to tell clients to resubmit the request to another URI |
| 400 | (“Bad Request”) Used to indicate nonspecific failure |
| 401 | (“Unauthorized”) Used when there is a problem with the client’s credentials |
| 403 | (“Forbidden”) Used to forbid access regardless of authorization state |
| 404 | (“Not Found”) Used when a client’s URI cannot be mapped to a resource |
| 405 | (“Method Not Allowed”) Used when the HTTP method is not supported |
| 406 | (“Not Acceptable”) Used when the requested media type cannot be served |
| 409 | (“Conflict”) should be used to indicate a violation of resource state |
| 412 | (“Precondition Failed”) should be used to support conditional operations |
| 415 | (“Unsupported Media Type”) Used when the media type of a request’s payload cannot be processed |
| 500 | (“Internal Server Error”) should be used to indicate API malfunction |

Table . HTTP Response Codes

Please note that all information provided via the APIs documented here are case sensitive. For example, “Administrator” is not the same as “administrator.”

## Authentication

Each API endpoint is designated as either *Public* or *Private*.

Private APIs mutate data in the TDS Session database. These require a user that is authenticated and authorized to modify the TDS Session database.

Public APIs are intended for machine-to-machine API calls. All calls to these APIs must originate from a server with the appropriate shared keys for authentication and authorization of the calling server.

# TDS Administration API Overview

## TDS Administration Overview

The TDS Administration component is responsible for

* Showing the opportunities to the user
* Extend grace period, extend expiration date, change segment permeability, reset, reopen, invalidate and restore opportunities.

## TDS Administration API Summary

## Authorization

All Tdsadmin API calls require authorization. This is accomplished via cookie in the request headers in this format:

Accept: application/json

Authorization: Bearer *insert-oauth-guid-here*

Content-Type: application/json

Accept:text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,\*/\*;q=0.8

Cookie:JSESSIONID={sessionInfo}; \_sm\_au\_c={authInfo}; amlbcookie=01; iPlanetDirectoryPro-drcdev={oauth-key}

## REST APIs

The TDS Administration component has 8 APIs. The private APIs allow mutation of the data contained in TDS and are intended to be used solely by administrators with access to the TDS Administration user interface to define roles, components, permissions and mappings. The public API provides read-only access to this information for client components to use in determining the appropriate level of authorization to grant a given user based on the user’s roles.

The following tables describe the TDS Administration public and private APIs.

| API Endpoint | HTTP Request | Description |
| --- | --- | --- |
| /getOpportunities | GET /getOpportunities? {keys}={values} | Provides the capability to query opportunities from the TDS based on searchable metadata. (public) |
| /resetOpportunity | POST /resetOpportunity | Reset wipes out an existing student’s test opportunity, allowing the student to take an opportunity again. Reset is used for situations where the student inadvertently commenced an opportunity or inadvertently submitted a test too early. |
| /invalidateOpportunity | POST /invalidateOpportunity | Invalidate makes an existing test opportunity invalid. This is used for situations such as when cheating is suspected. |
| /restoreOpportunity | POST /restoreOpportunity | Restore undoes a Reset action. This is used for situations where a test was inadvertently reset. |
| /reopenOpportunity | POST /reopenOpportunity | Reopen allows a student to complete a previously expired test opportunity. |
| /extendingOppGracePeriod | POST /extendingOppGracePeriod | If a test is paused beyond the grace period through no fault of the student, Grace Period Extension allows the test to be resumed as if the test was paused and resumed within the grace period. |
| /alterOpportunityExpiration | POST /alterOpportunityExpiration | For test opportunities that have already expired, Extend Expiration allows the student to resume the test and be able to go back and change previous responses. |
| /setOpportunitySegmentPerm | POST /setOpportunitySegmentPerm | This is additive to Grace Period Extension. In segmented tests where the segment is impermeable, Grace Period Extension will reopen the test but will not allow the student to go back into a prior impermeable segment. Adding Change Segment Permeability to Grace Period Extension allows the student to go back and change responses in otherwise inaccessible segments. |

Table : TDS Administration API Summary

# /getOpportunities?{keys}={values} (GET)

## Overview

Returns opportunities based on search criteria.

## HTTP Request [Public API]

GET /getOpportunities?sessionId={session\_id}&ssId={student\_id}&extSsId={external\_student\_id}&procedure={selected\_procedure}

## Request Parameters

| Parameter Name | Value | Description | Acceptable Values | Length |
| --- | --- | --- | --- | --- |
| ssId={student\_id} | String | Student identifier (ssid) from TDS session db. Either ssid or extSSID may be used; neither are required. If both exist, this API will choose the ssid. If neither exist, the sessionID must exist. If neither sessionID nor any SSID exist, this API call will fail. | Any valid TDS student identifier | Any |
| extSsId={external\_student\_id} | String | Student identifier (external ssid) from ART student db. Either ssid or extSSID may be used; neither are required. If both exist, this API will choose the ssid. If neither exist, the sessionID must exist. If neither sessionID nor any SSID exist, this API call will fail. | Any valid ART student identifier | Any |
| sessionId={sessionId} | String | Proctored session identifier from TDS session db. If neither ssId nor extssId exist, the sessionID must exist. If neither sessionID nor any SSID exist, this API call will fail. | Any valid session id | Any |
| procedure={procedure} | String | One of the seven stored procedures | Any valid stored procedure name from the following list:  changeperm, alter, extend, reset, reopen, restore, invalidate. | Any |

## Request Body

Accept:text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,\*/\*;q=0.8

Cookie:JSESSIONID=B7FD742A6AECC26A5DDD7042FA9726D4; \_sm\_au\_c=iVVtqWgRQwnSWr5513; amlbcookie=01; iPlanetDirectoryPro-drcdev=AQIC5wM2LY4SfcxMnufCvCliGnsG5JhfWf6MDXnDR-ElXZE.\*AAJTSQACMDIAAlNLABM4NDY2MDEzMTQ0NDU0NDQxNjY0AAJTMQACMDE.\*

## Response

If successful, this method returns a JSON object which contains a list of zero or more matching opportunities. This object contains the following attributes:

[

{

"oppKey": "2de85758-2657-435a-a26d-bfdd2fcc61f5",

"altSsid": "1009",

"name": "Lawson, james",

"testName": "SBAC-ELA-4-PETER-TO2-4-F-58-1.0",

"subject": "ELA",

"sessionId": "ALP-899",

"status": "expired",

"dateStarted": 1455920976350,

"dateExpired": 1461114114437,

"dateCompleted": null,

"datePaused": 1455931920240,

"segmentName": null,

"restart": 0,

"result": null,

"restoreOn": "segment",

"segmentPosition": 0,

"ispermeable": -1,

"permeable": false,

"reason": null,

"selected": false,

"selectedSitting": 0,

"doUpdate": true,

"dayIncrement": 0

}

]

## Examples

| **Example** | **Response Code** |
| --- | --- |
| **GET /getOpportunities?procedure=alter&extSsId=1030** | 200 |
| **GET /getOpportunities?procedure=alter&ssId=999** | 200 |
| **GET /getOpportunities?procedure=alter** | 400 |
| **GET /getOpportunities? ssId=999** | 400 |

# /resetOpportunity (POST)

## Overview

Post the opportunity information in the request body. Reset wipes out an existing student’s test opportunity, allowing the student to take an opportunity again. Reset is used for situations where the student inadvertently commenced an opportunity or inadvertently submitted a test too early. The call returns a ProcedureResult object in json format.

## HTTP Request [Public API]

POST /resetOpportunity

## Path Parameters

| Parameter Name | Value | Description | Acceptable Values | Length |
| --- | --- | --- | --- | --- |
| n/a |  |  |  |  |

## Request Parameters

| Parameter Name | Value | Description | Acceptable Values | Length |
| --- | --- | --- | --- | --- |
| oppkey | String (GUID) | This is the opportunity key that uniquely identifies an opportunity. | Any valid oppkey existing in the TDS session db. | Any |
| requestor | String | This is the email of the user logged in. | Any valid email. | Any |
| reason | String | The reason is entered by the user while executing the procedure. | Any string | Any |

## Request Header

POST /resetOpportunity

Accept: application/json

Authorization: Bearer *oath-token-here*

Content-Type: application/json

## Request Body

{

oppkey:d7c4c3f3-b6af-4745-a781-c0cda82b67e4

requestor: user@example.com

reason: administration

}

## Response

If successful, this method returns a JSON object with information like status, reason, context, appkey as seen below:

{

"status": "success",

"reason": administration,

"context": tdsadmin,

"appKey": tdsadmin

}

## Examples

| **Example** | **Response Code** |
| --- | --- |
| **POST /**resetOpportunity (with correct body contents) | 200 |
| **POST /**resetOpportunity (with incorrect body contents) | 400 |

# /invalidateOpportunity (POST)

## Overview

Post the opportunity information in the request body. Invalidate makes an existing test opportunity invalid. This is used for situations such as when cheating is suspected. The call returns a ProcedureResult object in json format.

## HTTP Request [Public API]

POST /invalidateOpportunity

## Path Parameters

| Parameter Name | Value | Description | Acceptable Values | Length |
| --- | --- | --- | --- | --- |
| n/a |  |  |  |  |

## Request Parameters

| Parameter Name | Value | Description | Acceptable Values | Length |
| --- | --- | --- | --- | --- |
| oppkey | String (GUID) | This is the opportunity key that uniquely identifies an opportunity. | Any valid oppkey existing in the TDS session db. | Any |
| requestor | String | This is the email of the user logged in. | Any valid email. | Any |
| reason | String | The reason is entered by the user while executing the procedure. | Any string | Any |

## Request Header

POST /invalidateOpportunity

Accept: application/json

Authorization: Bearer *oath-token-here*

Content-Type: application/json

## Request Body

{

oppkey:d7c4c3f3-b6af-4745-a781-c0cda82b67e4

requestor: user@example.com

reason: administration

}

## Response

If successful, this method returns a JSON object with information like status, reason, context, appkey as seen below:

{

"status": "success",

"reason": administration,

"context": tdsadmin,

"appKey": tdsadmin

}

## Examples

| **Example** | **Response Code** |
| --- | --- |
| **POST /**invalidateOpportunity (with correct body contents) | 200 |
| **POST /**invalidateOpportunity (with incorrect body contents) | 400 |