

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
1.0 Initial Release	Shajib Khan	05-May-2016

Table of Contents

API Documentation Conventions	5
Authentication	5
TDS Administration API Overview	7
TDS Administration Overview	7
TDS Administration API Summary	7
Authorization	7
REST APIs	7
/getOpportunities?{keys}={values} (GET)	9
Overview	9
HTTP Request [Public API]	9
Request Parameters	9
Request Header	9
Request Body	9
Response	9
Examples	11
/resetOpportunity (POST)	12
Overview	12
HTTP Request [Private API]	12
Path Parameters	12
Request Parameters	12
Request Header	12
Request Body	12
Response	12
Examples	13
/invalidateOpportunity (POST)	14
Overview	14
HTTP Request [Private API]	14
Path Parameters	14
Request Parameters	14
Request Header	14
Request Body	14
Response	14

Examples	15
/restoreOpportunity (POST)	16
Overview	16
HTTP Request [Private API]	16
Path Parameters	16
Request Parameters	16
Request Header	16
Request Body	16
Response	16
Examples	17
/reopenOpportunity (POST)	18
Overview	18
HTTP Request [Private API]	18
Path Parameters	18
Request Parameters	18
Request Header	18
Request Body	18
Response	18
Examples	19
/extendOppGracePeriod (POST)	20
Overview	20
HTTP Request [Private API]	20
Path Parameters	20
Request Parameters	20
Request Header	20
Request Body	20
Response	21
Examples	21
/alterOpportunityExpiration (POST)	22
Overview	22
HTTP Request [Private API]	22
Path Parameters	22
Request Parameters	22

Request Header	22
Request Body	22
Response	23
Request Header	23
Request Body	23
Response	23
Examples	23
/setOpportunitySegmentPerm (POST).....	25
Overview	25
HTTP Request [Private API]	25
Path Parameters	25
Request Parameters	25
Request Header	25
Request Body	26
Response	26
Examples	26

Tables

Table 1. HTTP Response Codes	5
Table 2: TDS Administration API Summary	8

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 1. 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 TDS Administration API calls require authorization. This is accomplished via oauth2 bearer authorization in request headers in this format:

```
Accept: application/json
Authorization: Bearer insert-oauth-guid-here
Content-Type: application/json
```

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.

Table 2 provides an overview of the available APIs, which are described in detail later in the document.

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.
/extendOppGracePeriod	POST /extendOppGracePeriod	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.

API Endpoint	HTTP Request	Description
/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 2: 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 Header

GET /tdsadmin/rest/getOpportunities?procedure=invalidate&extSsId=103 HTTP/1.1
Content-Type: application/json
Accept: application/json
Authorization: Bearer f88fac8e-5640-457f-85a0-30a0b14860ab

Request Body

NA

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": "00a42d6a-4fa1-4d04-b138-9392aefaacce",
      "altSsid": "103",
      "name": "Smith, John",
      "testName": "SBAC-ICA-FIXED-G3M-MATH-3",
      "subject": "MATH",
      "sessionId": "four-3",
      "status": "invalidated",
      "dateStarted": 1433200425514,
      "dateExpired": null,
      "dateCompleted": null,
      "datePaused": 1456448449426,
      "segmentName": null,
      "restart": 0,
      "result": null,
      "restoreOn": "segment",
      "segmentPosition": 0,
      "ispermeable": -1,
      "permeable": false,
      "reason": null,
      "selected": false,
      "selectedSitting": 0,
      "doUpdate": true,
      "dayIncrement": 0
    },
    {
      "oppKey": "ff227f9d-7759-4b9f-8992-ea2abd4a405a",
      "altSsid": "103",
      "name": "Smith, John",
      "testName": "SBAC-OP-ADAPTIVE-G3M-MATH-3",
      "subject": "MATH",
      "sessionId": "six-2",
      "status": "submitted",
      "dateStarted": 1432687632557,
      "dateExpired": null,
      "dateCompleted": 1432687768185,
      "datePaused": null,
      "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=103	200
GET /getOpportunities?procedure=alter&ssid=999	200
GET /getOpportunities?ssid=999&sessionId=alp-99&procedure=alter	200
GET /getOpportunities?ssid=999&sessionId=alp-99&procedure=alter&extSsid=789	200
GET /getOpportunities?procedure=alter	400
GET /getOpportunities?ssid=999	400
GET /getOpportunities?procedure=foobar&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 [Private API]

POST /resetOpportunity

Path Parameters

Parameter Name	Value	Description	Acceptable Values	Length
n/a				

Request Parameters

Parameter Name	Value	Description	Acceptable Values	Length
oppkey	UUID	This is the opportunity key that uniquely identifies an opportunity in the TDS session DB.	Java UUID data type. Defined by IETF RFC-4122. 128-bit integer (32 hexadecimal digits, with segments separated by a hyphen).	Any
requester	String	This is the email address of the logged-in user.	Any string	Any
reason	String	The reason is entered by the user while executing the procedure. (optional)	Any string	Any

Request Header

```
POST /tdsadmin/rest/resetOpportunity HTTP/1.1
Content-Type: application/x-www-form-urlencoded
Accept: application/json
Authorization: Bearer f88fac8e-5640-457f-85a0-30a0b14860ab
```

Request Body

```
{
  oppkey=01e01bdd-cd92-49c4-bc8a-de203b9fa5ce
  requester=user%40example.com
  reason=example+reason
}
```

Response

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

```
{
  "status": "success",
  "reason": null,
  "context": null,
  "appKey": null
}
```

Examples

Example	Response Code
POST /resetOpportunity (with correct body contents) If oppkey is not found in database, status will equal 'failed'	200
POST /resetOpportunity (with incorrect body contents) If any required parameter is empty or missing, a 400 error will result. If any values fall outside of their boundaries, a 400 error will result indicating the invalid value(s). If any required parameter mismatches the expected parameter type, a 400 error will result with an error indicating a syntax problem.	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 [Private API]

POST /invalidateOpportunity

Path Parameters

Parameter Name	Value	Description	Acceptable Values	Length
n/a				

Request Parameters

Parameter Name	Value	Description	Acceptable Values	Length
oppkey	UUID	This is the opportunity key that uniquely identifies an opportunity in the TDS session DB.	Java UUID data type. Defined by IETF RFC-4122. 128-bit integer (32 hexadecimal digits, with segments separated by a hyphen).	Any
requester	String	This is the email address of the logged-in user.	Any string	Any
reason	String	The reason is entered by the user while executing the procedure.(optional)	Any string	Any

Request Header

POST /tdsadmin/rest/invalidateOpportunity HTTP/1.1
Content-Type: application/x-www-form-urlencoded
Accept: application/json
Authorization: Bearer f88fac8e-5640-457f-85a0-30a0b14860ab

Request Body

```
{
  oppkey=05ab5a2e-53f7-48aa-9181-0e3b49475829
  requester=user%40example.com
  reason=example+reason
}
```

Response

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

```
{
  "status": "success",
```

```
"reason": null,  
"context": null,  
"appKey": null  
}
```

Examples

Example	Response Code
POST /invalidateOpportunity (with correct body contents) If oppkey is not found in database, status will equal 'failed'	200
POST /invalidateOpportunity (with incorrect body contents) If any required parameter is empty or missing, a 400 error will result. If any values fall outside of their boundaries, a 400 error will result indicating the invalid value(s). If any required parameter mismatches the expected parameter type, a 400 error will result with an error indicating a syntax problem.	400

/restoreOpportunity (POST)

Overview

Post the opportunity information in the request body. Restore undoes a Reset action. This is used for situations where a test was inadvertently reset. The call returns a ProcedureResult object in JSON format.

HTTP Request [Private API]

POST /restoreOpportunity

Path Parameters

Parameter Name	Value	Description	Acceptable Values	Length
n/a				

Request Parameters

Parameter Name	Value	Description	Acceptable Values	Length
oppkey	UUID	This is the opportunity key that uniquely identifies an opportunity in the TDS session DB.	Java UUID data type. Defined by IETF RFC-4122. 128-bit integer (32 hexadecimal digits, with segments separated by a hyphen).	Any
requester	String	This is the email address of the logged-in user.	Any string	Any
reason	String	The reason is entered by the user while executing the procedure.(optional)	Any string	Any

Request Header

POST /tdsadmin/rest/restoreOpportunity HTTP/1.1
Content-Type: application/x-www-form-urlencoded
Accept: application/json
Authorization: Bearer b785aac2-fde1-4a2b-a4a8-24c07caea06f

Request Body

```
{
  oppkey=05ab5a2e-53f7-48aa-9181-0e3b49475829
  requester=user%40example.com
  reason=example+reason
}
```

Response

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

```
{
  "status": "success",
```



```
"reason": null,  
"context": null,  
"appKey": null  
}
```

Examples

Example	Response Code
POST /restoreOpportunity (with correct body contents) If oppkey is not found in database, status will equal 'failed'	200
POST /restoreOpportunity (with incorrect body contents) If any required parameter is empty or missing, a 400 error will result. If any values fall outside of their boundaries, a 400 error will result indicating the invalid value(s). If any required parameter mismatches the expected parameter type, a 400 error will result with an error indicating a syntax problem.	400

/reopenOpportunity (POST)

Overview

Post the opportunity information in the request body. Reopen allows a student to complete a previously expired test opportunity. The call returns a ProcedureResult object in JSON format.

HTTP Request [Private API]

POST /reopenOpportunity

Path Parameters

Parameter Name	Value	Description	Acceptable Values	Length
n/a				

Request Parameters

Parameter Name	Value	Description	Acceptable Values	Length
oppkey	UUID	This is the opportunity key that uniquely identifies an opportunity in the TDS session DB.	Java UUID data type. Defined by IETF RFC-4122. 128-bit integer (32 hexadecimal digits, with segments separated by a hyphen).	Any
requester	String	This is the email address of the logged-in user.	Any string	Any
reason	String	The reason is entered by the user while executing the procedure. (optional)	Any string	Any

Request Header

POST /tdsadmin/rest/reopenOpportunity HTTP/1.1
Content-Type: application/x-www-form-urlencoded
Accept: application/json
Authorization: Bearer bb5e00ff-ea75-4a82-940d-1d86d27fd789

Request Body

```
{
  oppkey=05ab5a2e-53f7-48aa-9181-0e3b49475829
  requester=user%40example.com
  reason=example+reason
}
```

Response

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

```
{
```

```
"status": "success",
"reason": null,
"context": null,
"appKey": null
}
```

Examples

Example	Response Code
POST /reopenOpportunity (with correct body contents) If oppkey is not found in database, status will equal 'failed'	200
POST /reopenOpportunity (with incorrect body contents) If any required parameter is empty or missing, a 400 error will result. If any values fall outside of their boundaries, a 400 error will result indicating the invalid value(s). If any required parameter mismatches the expected parameter type, a 400 error will result with an error indicating a syntax problem.	400

/extendOppGracePeriod (POST)

Overview

Post the opportunity information in the request body. 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. The call returns a ProcedureResult object in JSON format.

HTTP Request [Private API]

POST /extendOppGracePeriod

Path Parameters

Parameter Name	Value	Description	Acceptable Values	Length
n/a				

Request Parameters

Parameter Name	Value	Description	Acceptable Values	Length
oppkey	UUID	This is the opportunity key that uniquely identifies an opportunity in the TDS session DB.	Java UUID data type. Defined by IETF RFC-4122. 128-bit integer (32 hexadecimal digits, with segments separated by a hyphen).	Any
requester	String	This is the email address of the logged-in user.	Any string	Any
reason	String	The reason is entered by the user while executing the procedure. (optional)	Any string	Any
selectedsitting	Integer	Grace period is extended on a particular sitting.	Any integer from 0-99, inclusive	2
doupdate	Boolean	Procedure is executed only if true.	True or 1	4

Request Header

POST /tdsadmin/rest/extendOppGracePeriod HTTP/1.1
Content-Type: application/x-www-form-urlencoded
Accept: application/json
Authorization: Bearer c9eaf1fc-f8d7-45fe-b87f-aff2336fd6ae

Request Body

```
{
  oppkey=05ab5a2e-53f7-48aa-9181-0e3b49475829
  requester=user%40example.com
  reason=example+reason
}
```

```
selectedsitting=3
doupdate=true
}
```

Response

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

```
{
  "status": "success",
  "reason": null,
  "context": null,
  "appKey": null
}
```

Examples

Example	Response Code
POST /extendOppGracePeriod (with correct body contents) If oppkey is not found in database, status will equal 'failed'	200
POST /extendOppGracePeriod (with incorrect body contents) If any required parameter is empty or missing, a 400 error will result. If any values fall outside of their boundaries, a 400 error will result indicating the invalid vaue(s). If any required parameter mismatches the expected parameter type, a 400 error will result with an error indicating a syntax problem.	400

/alterOpportunityExpiration (POST)

Overview

Post the opportunity information in the request body. 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. The call returns a ProcedureResult object in JSON format.

HTTP Request [Private API]

POST /alterOpportunityExpiration

Path Parameters

Parameter Name	Value	Description	Acceptable Values	Length
n/a				

Request Parameters

Parameter Name	Value	Description	Acceptable Values	Length
oppkey	UUID	This is the opportunity key that uniquely identifies an opportunity in the TDS session DB.	Java UUID data type. Defined by IETF RFC-4122. 128-bit integer (32 hexadecimal digits, with segments separated by a hyphen).	Any
requester	String	This is the email address of the logged-in user.	Any string	Any
reason	String	The reason is entered by the user while executing the procedure. (optional)	Any string	Any
dayincrement	Integer	Number of days to be added to the original expiration date, the new date must be in future.	The value has to be an integer in range <-365,365>	4

Request Header

POST /tdsadmin/rest/alterOpportunityExpiration HTTP/1.1
Content-Type: application/x-www-form-urlencoded
Accept: application/json
Authorization: Bearer c9eaf1fc-f8d7-45fe-b87f-aff2336fd6ae

Request Body

```
{
  oppkey=05ab5a2e-53f7-48aa-9181-0e3b49475829
  requester=user%40example.com
  reason=example+reason
  dayincrement=300
}
```

Response

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

```
{
  "status": "success",
  "reason": null,
  "context": null,
  "appKey": null
}
```

Example 2:

Request Header

```
POST /tdsadmin/rest/alterOpportunityExpiration HTTP/1.1
Content-Type: application/x-www-form-urlencoded
Accept: application/json
Authorization: Bearer c9eaf1fc-f8d7-45fe-b87f-aff2336fd6ae
```

Request Body

```
{
  oppkey=05ab5a2e-53f7-48aa-9181-0e3b49475829
  requester=user%40example.com
  reason=example+reason
  dayincrement=300
}
```

Response

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

```
{
  "status": "failed",
  "reason": "opportunity status (invalidated) prevents expiration date alterations [-----]",
  "context": null,
  "appKey": "opportunity status (invalidated) prevents expiration date alterations"
}
```

Examples

Example	Response Code
POST /alterOpportunityExpiration (with correct body contents) If oppkey is not found in database, status will equal 'failed'	200
POST /alterOpportunityExpiration (with incorrect body contents) If any required parameter is empty or missing, a 400 error will result. If any values fall outside of their boundaries, a 400 error will result indicating the invalid vaue(s). If any required parameter mismatches the expected parameter type, a 400 error will result with an error indicating a syntax problem.	400



/setOpportunitySegmentPerm (POST)

Overview

Post the opportunity information in the request body. 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. The call returns a ProcedureResult object in JSON format.

HTTP Request [Private API]

POST /setOpportunitySegmentPerm

Path Parameters

Parameter Name	Value	Description	Acceptable Values	Length
n/a				

Request Parameters

Parameter Name	Value	Description	Acceptable Values	Length
oppkey	UUID	This is the opportunity key that uniquely identifies an opportunity in the TDS session DB.	Java UUID data type. Defined by IETF RFC-4122. 128-bit integer (32 hexadecimal digits, with segments separated by a hyphen).	32
requester	String	This is the email address of the logged-in user.	Any string	Any
reason	String	The reason is entered by the user while executing the procedure. (optional)	Any string	Any
segmentid	String	Segment name in the test	Any string	Any
segmentposition	Integer	Segment position in the test	Any positive integer.	Any
restoreon	String	The variable represents what state the segment needs to be set to.	Valid values include <i>segment</i> , <i>paused</i> , <i>completed</i> .	Any
ispermeable	Integer	Flag value that indicates whether the segment is permeable or not. Value will be set based on user input.	Valid values are -1 or 1.	2

Request Header

```
POST /tdsadmin/rest/setOpportunitySegmentPerm HTTP/1.1
Content-Type: application/x-www-form-urlencoded
Accept: application/json
Authorization: Bearer c9eaf1fc-f8d7-45fe-b87f-aff2336fd6ae
```

Request Body

```
{
  oppkey=c911273d-35e0-461b-9461-45d01fc3e815
  requester=user%40example.com
  reason=example+reason
  segmentid=Fixed+S
  segmentposition=2
  restoreon=segment
  ispermeable=1
}
```

Response

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

```
{
  "status": "success",
  "reason": null,
  "context": null,
  "appKey": null
}
```

Examples

Example	Response Code
POST /setOpportunitySegmentPerm (with correct body contents) If oppkey is not found in database, status will equal 'failed'	200
POST /setOpportunitySegmentPerm (with incorrect body contents) If any required parameter is empty or missing, a 400 error will result. If any values fall outside of their boundaries, a 400 error will result indicating the invalid vaue(s). If any required parameter mismatches the expected parameter type, a 400 error will result with an error indicating a syntax problem.	400