



SMARTERAPP INTERFACE SPECIFICATION
for
CORE STANDARDS COMPONENT
American Institutes for Research

Revision History

Revision Description	Author/Modifier	Date
Initial Release	David Lopez de Quintana	April 9, 2014

Contents

API Documentation Conventions	3
Core Standards API Overview	4
Publisher (GET).....	7
Publisher (GET with Publisher)	9
Subject (GET).....	11
Subject (GET with Subject)	13
Subject (GET with Publisher)	15
Grade (GET)	17
Publication (GET with Publisher and Subject)	21
Standard (GET with Publication and Grade).....	23
Category (GET with Publication)	25
Grade (GET with Publication).....	27
SOCK (GET with Publication)	29
Publisher (POST)	31
Subject (POST).....	33
Grade (POST).....	35
Staging Publication (POST).....	37
Staging Publication (PUT)	39
Staging Publication NextVersion (GET)	41
Staging Publication (GET)	42
Staging Category (GET)	44
Staging Grade (GET).....	46
Staging Standard (GET)	48
Staging SOCK (GET)	50
Staging Category (PUT)	52
Staging Grade (PUT).....	54
Staging Standard (PUT)	56
Staging SOCK (PUT)	58
Staging (DELETE)	60
Publication (POST)	61

API Documentation Conventions

The API endpoints are described as follows:

METHOD /endpointname?urlparameter1={value1}&urlparameter2={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
- urlparameter provides parameterized information to this API endpoint
 - Italicized parameter names and values are optional
 - {curly brackets} denotes variables
 - Additional URL parameters are delimited by an ampersand (&)
- Request parameters are documented separately from URL parameters and always expected in JSON format (content-type:application/json)

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."

Core Standards API Overview

Component Overview

The Core Standards component is a repository of standards publications for use by components that require standards-based information:

- Item Authoring: for aligning items to one or more standards from a particular publication
- Test Authoring: for creating test blueprints against a selected set of standards
- Digital Library: for selecting resources to address teaching and assessing particular standards

Other components such as Test Delivery use standards information in the test blueprint for CAT item selection against a blueprint, but may obtain the standards-based information directly from the blueprint in the test package rather than interfacing directly with Core Standards.

Standards-based information is published by a *publisher*, usually the committee or body that creates the standards. Standards publications are usually written for a particular *subject*, but can be used to create tests of related subjects. For example, a Mathematics publication may also be used to create tests for Algebra or Geometry. Publications are *versioned*, and additional standards or grade alignments can be added by incrementing the version number of a publication. New versions of a publication do not replace older versions of a publication and are added to the Core Standards repository in addition to the previous versions. Individual standards are aligned to particular *grades*, standards can be filtered to examine or use standards information for a particular grade or set of grades.

Core Standards API Summary

The Core Standards API is used to browse and extract standards publication information by publisher, subject, grade or other criteria. It has a public API that is used to browse and extract information in a read-only manner as client applications are not allowed to mutate the standards data. The private API of Core Standards exists to support the user interface in the creation, uploading or editing of standards-based information by administrative users that are authorized to access the Core Standards user interface.

Many of the APIs are “staging” APIs due to the method that Core Standards uses to edit and upload publications. Freshly uploaded publications or publications selected for editing are placed in temporary staging tables. They are edited and validated in these temporary staging tables, and when the publication is deemed correct and consistent, APIs are available to move the temporary publications from the staging tables to permanent database tables. For this reason, a parallel set of GET APIs for obtaining information for the staging tables are deemed part of the private API since they are only for the use of the Core Standards user interface.

The following tables describe the Core Standards public and private APIs.

API Endpoint	HTTP Request	Description
publisher	GET publisher	Obtains information about all publishers
publisher	GET publisher/{publisherKey}	Obtains information about a specific publisher

API Endpoint	HTTP Request	Description
subject	GET subject	Obtains information about all subjects
subject	GET subject/{subjectKey}	Obtains information about a specific subject
subject	GET subject?publisher={publisherKey}	Obtains information about subjects used by a specific publisher
grade	GET grade	Obtains information about all grades
publication	GET publication?publisher={publisherKey}&subject={subjectKey}	Obtains information about grades used by a specific publisher for a specific subject
standard	GET publication/{publicationKey}/standard?grade={gradekey}	Obtains standards information for standards in a specific publication aligned to specific grade
category	GET publication/{publicationKey}/category	Obtains information about the standards categories for a given publication
grade	GET publication/{publicationKey}/grade	Obtains information about the grades used by a specific publication
sock	GET publication/{publicationKey}/sock	Obtains information about the SOCK information for a given publication

Table 2. Core Standards Public API Summary

API Endpoint	HTTP Request	Description
publisher	POST publisher	Creates a new publisher
subject	POST subject	Creates a new subject
grade	POST grade	Creates a new grade
publication	POST staging/publication	Copies a publication into staging tables for editing
publication	PUT staging/publication	Makes publication in staging tables editable
staging	GET staging/publication/nextVersion	Retrieves next version number for a given publication
staging	GET /staging/publication	Obtains publication metadata from staging tables
staging	GET staging/category	Obtains information about the standards categories for a publication in staging tables
staging	GET staging/grade	Obtains information about the grades used by a specific publication in staging tables
staging	GET staging/standard	Obtains standards information for a publication in staging tables
staging	GET staging/sock	Obtains SOCK information for a publication in staging tables
staging	PUT staging/category	Edits category information in staging tables
staging	PUT staging/grade	Edits grade information in staging tables
staging	PUT staging/standard	Edits standard information in staging tables
staging	PUT staging/sock	Edits SOCK information in staging tables

API Endpoint	HTTP Request	Description
staging	DELETE staging	Deletes a publication from the staging tables
staging	POST publication	Moves a publication from the staging tables into the main tables

Table 3: Core Standards Private API Summary

Publisher (GET)

Overview

This API obtains information about all publishers.

HTTP Request

```
GET publisher
```

URL Parameters

No URL Parameters are required for this API.

Request Parameters

All parameters are provided on the command line. No request parameters are provided.

Authorization

This interface is a public interface that is part of the shared service interface of the Core Standards component. It is intended for machine-to-machine API calls. All calls to this API must originate from a server with the appropriate shared keys for authentication and authorization of the calling server.

Request Body

Do not supply a request body for this API.

Response

This API returns a JSON object that describes each publisher's key and name.

Examples

Example	Response Code
<pre>REQ: GET /publisher RESP: { "status": "success", "errors": [], "warnings": [], "validationsPassed": [], "payload": [{ "key": "CC", "name": "Common Core" }, { "key": "SBAC", "name": "Smarter Balanced Assessment Consortium" }] }</pre>	200

Example	Response Code
}	

Table 4. Publisher (GET) Examples

Publisher (GET with Publisher)

Overview

This API obtains information about a specific publisher.

HTTP Request

```
GET publisher/{publisherKey}
```

URL Parameters

Parameter Name	Value	Description	Acceptable Values	Length
publisherKey	string	The identifier of a particular publisher to obtain information	Any printable ASCII character	50

Table 5. Publisher (GET with Publisher) URL Parameters

Request Parameters

All parameters are provided on the command line. No request parameters are provided.

Authorization

This interface is a public interface that is part of the shared service interface of the Core Standards component. It is intended for machine-to-machine API calls. All calls to this API must originate from a server with the appropriate shared keys for authentication and authorization of the calling server.

Request Body

Do not supply a request body for this API.

Response

This API returns a JSON object that describes each publisher's key and name. It fails if the specified publisher does not exist.

Examples

Example	Response Code
<pre>REQ: GET /publisher/SBAC RESP: { "status": "success", "errors": [], "warnings": [], "validationsPassed": [], "payload": { "key": "SBAC", "name": " Smarter Balanced Assessment Consortium " } }</pre>	200

Example	Response Code
<pre> } REQ: /publisher/fakePublisher RESP: { "status": "failed", "errors": [""], "warnings": [], "validationsPassed": [], "payload": null } </pre>	<p style="text-align: center;">404</p>

Table 6. Publisher (GET with Publisher) Examples

Subject (GET)

Overview

This API obtains information about subject information stored in Core Standards.

HTTP Request

```
GET subject
```

URL Parameters

No URL parameters are required for this API.

Request Parameters

All parameters are provided on the command line. No request parameters are provided.

Authorization

This interface is a public interface that is part of the shared service interface of the Core Standards component. It is intended for machine-to-machine API calls. All calls to this API must originate from a server with the appropriate shared keys for authentication and authorization of the calling server.

Request Body

Do not supply a request body for this API.

Response

This API returns a JSON object with information about all subjects.

Examples

Example	Response Code
<pre>REQ: GET /subject RESP: { "status": "success", "errors": [], "warnings": [], "validationsPassed": [], "payload": [{ "key": "ELA", "name": "English Language Arts", "code": "ELA" }, { "key": "MA", "name": "Mathematics", "code": "MA" }] }</pre>	200

Example	Response Code
<pre data-bbox="185 344 250 407">}] }</pre>	

Table 7. Subject (GET) Examples

Subject (GET with Subject)

Overview

This API obtains information about a specific subject stored in Core Standards.

HTTP Request

```
GET subject/{subjectKey}
```

URL Parameters

Parameter Name	Value	Description	Acceptable Values	Length
subject	string	A particular subject about which to obtain information	Any printable ASCII character	50

Table 8. Subject (GET with Subject) URL Parameters

Request Parameters

All parameters are provided on the command line. No request parameters are provided.

Authorization

This interface is a public interface that is part of the shared service interface of the Core Standards component. It is intended for machine-to-machine API calls. All calls to this API must originate from a server with the appropriate shared keys for authentication and authorization of the calling server.

Request Body

Do not supply a request body for this API.

Response

This API returns a JSON object with information about a particular subject. It fails if the specified subject doesn't exist.

Examples

Example	Response Code
<pre>REQ: GET /subject/MA RESP: { "status": "success", "errors": [], "warnings": [], "validationsPassed": [], "payload": [{ "key": "MA", "name": "Mathematics",</pre>	200

Example	Response Code
<pre> "code": "MA" }] } </pre>	
<pre> REQ: GET /subject/fakeSubject RESP: { "status": "failed", "errors": [""], "warnings": [], "validationsPassed": [], "payload": null } </pre>	404

Table 9. Subject (GET with Subject) Examples

Subject (GET with Publisher)

Overview

This API obtains information about subjects used by a specific publisher.

HTTP Request

```
GET subject?publisher={publisherKey}
```

URL Parameters

Parameter Name	Value	Description	Acceptable Values	Length
publisherKey	string	The identifier of a particular publisher to get subject information	Any printable ASCII character	50

Table 10. Subject (GET with Publisher) URL Parameters

Request Parameters

All parameters are provided on the command line. No request parameters are provided.

Authorization

This interface is a public interface that is part of the shared service interface of the Core Standards component. It is intended for machine-to-machine API calls. All calls to this API must originate from a server with the appropriate shared keys for authentication and authorization of the calling server.

Request Body

Do not supply a request body for this API.

Response

This API returns a JSON object about subjects used by publications belonging to a particular publisher. It fails if the publisher doesn't exist.

Examples

Example	Response Code
<pre>REQ: GET /subject?publisher=SBAC RESP: { "status": "success", "errors": [], "warnings": [], "validationsPassed": [], "payload": [{ "key": "ELA",</pre>	200

Example	Response Code
<pre> "name": "English Language Arts", "code": "ELA" }, { "key": "MA", "name": "Mathematics", "code": "MA" }] } </pre>	

Table 11. Subject (GET with Publisher) Examples

Grade (GET)

Overview

This API obtains information about all grades.

HTTP Request

GET /grade

URL Parameters

No URL Parameters are required for this API.

Request Parameters

All parameters are provided on the command line. No request parameters are provided.

Authorization

This interface is a public interface that is part of the shared service interface of the Core Standards component. It is intended for machine-to-machine API calls. All calls to this API must originate from a server with the appropriate shared keys for authentication and authorization of the calling server.

Request Body

Do not supply a request body for this API.

Response

This API returns a JSON object with information about all grades.

Examples

Example	Response Code
<pre> REQ: GET /grade RESP: { "status": "success", "errors": [], "warnings": [], "validationsPassed": [], "payload": [{ "key": "1", "name": "1", "description": null }, { "key": "10", "name": "10", "description": null }] } </pre>	200

Example	Response Code
<pre> }, { "key": "11", "name": "11", "description": null }, { "key": "11-12", "name": "11-12", "description": "Grade 11 and 12 for Common Core Standards" }, { "key": "12", "name": "12", "description": null }, { "key": "2", "name": "2", "description": null }, { "key": "3", "name": "3", "description": null }, { "key": "4", "name": "4", "description": null }, { "key": "5", "name": "5", "description": null }, { "key": "6", "name": "6", "description": null }, { "key": "6-8", "name": "6-8", "description": null }, { "key": "7", "name": "7", "description": null }, { "key": "8", "name": "8", "description": null }, { "key": "9", "name": "9", "description": null }, { "key": "9-10", </pre>	

Example	Response Code
<pre> "name": "9-10", "description": "Grade 9 and 10 for Common Core Standards" }, { "key": "9-12", "name": "9-12", "description": null }, { "key": "A", "name": "A", "description": null }, { "key": "Biological Science", "name": "Biological Science", "description": null }, { "key": "Chemistry", "name": "Chemistry", "description": null }, { "key": "F", "name": "F", "description": null }, { "key": "G", "name": "G", "description": null }, { "key": "HS", "name": "HS", "description": "High School for Common Core Standards" }, { "key": "KG", "name": "KG", "description": null }, { "key": "N", "name": "N", "description": null }, { "key": "NA", "name": "NA", "description": null }, { "key": "Physics", "name": "Physics", "description": null }, { "key": "S", "name": "S", "description": null } </pre>	

Example	Response Code
<pre>] } </pre>	

Table 12. Grade (GET) Examples

Publication (GET with Publisher and Subject)

Overview

This API obtains information about publications for a given publisher and subject.

HTTP Request

```
GET /publication?publisher={publisherKey}& subject={subjectKey}
```

URL Parameters

Parameter Name	Value	Description	Acceptable Values	Length
publisherKey	string	Publisher on which to obtain publication information	Any printable ASCII character	50
subjectKey	string	Subject on which to obtain publication information	Any printable ASCII character	50

Table 13. Publication (GET with Publisher and Subject) URL Parameters

Request Parameters

All parameters are provided on the command line. No request parameters are provided.

Authorization

This interface is a public interface that is part of the shared service interface of the Core Standards component. It is intended for machine-to-machine API calls. All calls to this API must originate from a server with the appropriate shared keys for authentication and authorization of the calling server.

Request Body

Do not supply a request body for this API.

Response

This API returns a JSON object with information about all publications for a given publisher and subject. This API fails if the given publisher or subject do not exist.

Examples

Example	Response Code
<pre>REQ: GET /publication?publisher=SBAC&subject=ELA RESP: { "status": "success", "errors": [], "warnings": [], "validationsPassed": [], "payload": [{</pre>	200

Example	Response Code
<pre> "key": "SBAC-ELA-v1", "fkPublisher": "SBAC", "version": 1, "description": "Common Core ELA Standards", "fkSubject": "ELA", "subjectLabel": "English Language Arts", "status": null }] </pre>	

Table 14. Publication (GET with Publisher and Subject) Examples

Standard (GET with Publication and Grade)

Overview

This API obtains standards information from a particular publication that are filtered by a particular grade.

HTTP Request

```
GET publication/{publicationKey}/standard?grade={gradekey}
```

URL Parameters

Parameter Name	Value	Description	Acceptable Values	Length
publicationKey	string	Publication on which to obtain standards	Any printable ASCII character	50
gradeKey	string	Grade on which to filter the standards publication	Any printable ASCII character	50

Table 15. Standard (GET with Publication and Grade) URL Parameters

Request Parameters

All parameters are provided on the command line. No request parameters are provided.

Authorization

This interface is a public interface that is part of the shared service interface of the Core Standards component. It is intended for machine-to-machine API calls. All calls to this API must originate from a server with the appropriate shared keys for authentication and authorization of the calling server.

Request Body

Do not supply a request body for this API.

Response

This API returns a JSON object with information about publication standards for a specific publication that are aligned to a specific grade. This API fails if

Examples

Example	Response Code
<pre>REQ: GET /publication/SBAC-ELA-v1/standard?grade=5 RESP: { "status": "success", "errors": [], "warnings": [], "validationsPassed": [],],</pre>	200

Example	Response Code
<pre> "payload": [{ "key": "SBAC-ELA-v1:1-IT", "name": "1-IT", "fkParent": null, "fkPublication": "SBAC-ELA-v1", "description": "Read Analytically: Informational Text - Students can read closely and analytically to comprehend a range of increasingly complex literary and informational texts.", "treeLevel": 1, "fkGradeLevel": "5", "shortName": "Read Analytically: Informational Text" }, <repeat for every standard>] </pre>	

Table 16. Standard (GET with Publication and Grade) Examples

Category (GET with Publication)

Overview

This API obtains category information about a particular publication.

HTTP Request

```
GET publication/{publicationKey}/category
```

URL Parameters

Parameter Name	Value	Description	Acceptable Values	Length
publicationKey	string	Publication to obtain category information	Any printable ASCII character	50

Table 17. Category (GET with Publication) URL Parameters

Request Parameters

All parameters are provided on the command line. No request parameters are provided.

Authorization

This interface is a public interface that is part of the shared service interface of the Core Standards component. It is intended for machine-to-machine API calls. All calls to this API must originate from a server with the appropriate shared keys for authentication and authorization of the calling server.

Request Body

Do not supply a request body for this API.

Response

This API returns a JSON object about the hierarchical category levels in the specified publication. This API fails if the publication doesn't exist.

Examples

Example	Response Code
<pre>REQ: GET /publication/SBAC-MA-V1/category RESP: { "status": "success", "errors": [], "warnings": [], "validationsPassed": [], "payload": [{ "name": "Claim", "treeLevel": 1, }] }</pre>	200

Example	Response Code
<pre> "fkPublication": "SBAC-MA-v1" }, { "name": "Content Domain", "treeLevel": 2, "fkPublication": "SBAC-MA-v1" }, { "name": "Target", "treeLevel": 3, "fkPublication": "SBAC-MA-v1" }, { "name": "Emphasis", "treeLevel": 4, "fkPublication": "SBAC-MA-v1" }, { "name": "Common Core Standard", "treeLevel": 5, "fkPublication": "SBAC-MA-v1" }] </pre>	

Table 18. Category (GET with Publication) Examples

Grade (GET with Publication)

Overview

This API obtains grade information about a particular publication.

HTTP Request

```
GET publication/{publicationKey}/grade
```

URL Parameters

Parameter Name	Value	Description	Acceptable Values	Length
publicationKey	string	Publication to obtain grade information	Any printable ASCII character	50

Table 19. Grade (GET with Publication) URL Parameters

Request Parameters

All parameters are provided on the command line. No request parameters are provided.

Authorization

This interface is a public interface that is part of the shared service interface of the Core Standards component. It is intended for machine-to-machine API calls. All calls to this API must originate from a server with the appropriate shared keys for authentication and authorization of the calling server.

Request Body

Do not supply a request body for this API.

Response

This API returns a JSON object with information about all grades that are aligned to standards on a given publication. This API fails if the publication doesn't exist.

Examples

Example	Response Code
<pre>REQ: GET RESP: { "status": "success", "errors": [], "warnings": [], "validationsPassed": [], "payload": [{ "key": "1", "name": "1",</pre>	200

Example	Response Code
<pre> "description": null }, { "key": "10", "name": "10", "description": null }, { "key": "11", "name": "11", "description": null }, { "key": "2", "name": "2", "description": null }, { "key": "3", "name": "3", "description": null }, { "key": "4", "name": "4", "description": null }, { "key": "5", "name": "5", "description": null }, { "key": "6", "name": "6", "description": null }, { "key": "7", "name": "7", "description": null }, { "key": "8", "name": "8", "description": null }, { "key": "9", "name": "9", "description": null }, { "key": "KG", "name": "KG", "description": null }] </pre>	

Table 20. Grade (GET with Publication) Examples

SOCK (GET with Publication)

Overview

This API obtains SOCK information about a particular publication.

HTTP Request

```
GET publication/{publicationKey}/sock
```

URL Parameters

Parameter Name	Value	Description	Acceptable Values	Length
publicationKey	string	Publication to obtain SOCK information	Any printable ASCII character	50

Table 21. SOCK (GET with Publication) URL Parameters

Request Parameters

All parameters are provided on the command line. No request parameters are provided.

Authorization

This interface is a public interface that is part of the shared service interface of the Core Standards component. It is intended for machine-to-machine API calls. All calls to this API must originate from a server with the appropriate shared keys for authentication and authorization of the calling server.

Request Body

Do not supply a request body for this API.

Response

This API returns SOCK (Some Other Category of Knowledge) information contained in a given publication. SOCKs are generalized categories used to group items in an assessment. This API fails if the publication doesn't exist.

Examples

Example	Response Code
<pre>REQ: GET /publication/SBAC-TST-V1/sock RESP: { "status": "success", "errors": [], "warnings": [], "validationsPassed": [], "payload": [{ "knowledgeCategory": "Depth of Knowledge Level 1",</pre>	200

Example	Response Code
<pre> "description": "Depth of Knowledge Level 1 (Description)", "socksKey": "SBAC-TST-v1:Depth of Knowledge Level 1", "fkPublication": "SBAC-TST-v1" }, { "knowledgeCategory": "Depth of Knowledge Level 2", "description": "Depth of Knowledge Level 2 (Description)", "socksKey": "SBAC-TST-v1:Depth of Knowledge Level 2", "fkPublication": "SBAC-TST-v1" }, { "knowledgeCategory": "Depth of Knowledge Level 3", "description": "Depth of Knowledge Level 3 (Description)", "socksKey": "SBAC-TST-v1:Depth of Knowledge Level 3", "fkPublication": "SBAC-TST-v1" }, { "knowledgeCategory": "Informational", "description": "Informational (Description)", "socksKey": "SBAC-TST-v1:Informational", "fkPublication": "SBAC-TST-v1" }, { "knowledgeCategory": "Literary", "description": "Literary (Description)", "socksKey": "SBAC-TST-v1:Literary", "fkPublication": "SBAC-TST-v1" }] </pre>	

Table 22. SOCK (GET with Publication) Examples

Publisher (POST)

Overview

This API creates a new Publisher.

HTTP Request

POST publisher

URL Parameters

No URL parameters are required for this API.

Request Parameters

Provide a JSON object with the key and name of the new publisher to be created:

```
{
  "name": "testPublisher2",
  "key": "TP2"
}
```

Authorization

This interface is a private interface that mutates the data in Core Standards and is intended for the Core Standards user interface. It requires a user that is authenticated and authorized to access the Core Standards user interface.

Request Body

Do not supply a request body for this API.

Response

This API returns a JSON object that describes the status of the request and the name and key of the publisher created. This API fails if the name or key of the publisher already exists.

Examples

Example	Response Code
<pre>REQ: POST /publisher REQ PARAM: { "name": "testPublisher2", "key": "TP2" } RESP: { "status": "success", "errors": [], "warnings": [], "validationsPassed": [],],</pre>	<p>200</p>

Example	Response Code
<pre>"payload": { "key": "TP2", "name": "testPublisher2" }</pre>	

Table 23. Publisher (POST) API Examples

Subject (POST)

Overview

This API creates a new Subject.

HTTP Request

POST publisher

URL Parameters

No URL parameters are required for this API.

Request Parameters

A JSON object with the code and name of the new subject to be created:

```
{
  "name": "testSubject1",
  "code": "TS1"
}
```

Authorization

This interface is a private interface that mutates the data in Core Standards and is intended for the Core Standards user interface. It requires a user that is authenticated and authorized to access the Core Standards user interface.

Request Body

Do not supply a request body for this API.

Response

This API returns a JSON object that describes the status of the request and the name and code of the subject created. This API fails if the name or code of the subject already exists.

Examples

Example	Response Code
<pre>REQ: POST /subject REQ PARAM: { "name": "testSubject1", "code": "TS1" } RESP: { "status": "success", "errors": [], "warnings": [], "validationsPassed": [],],</pre>	200

Example	Response Code
<pre>"payload": { "key": null, "name": "testSubject1", "code": "TS1" }</pre>	

Table 24. Subject (POST) API Examples

Grade (POST)

Overview

This API creates a new Grade.

HTTP Request

POST grade

URL Parameters

No URL parameters are required for this API.

Request Parameters

A JSON object with the name and description of the new subject to be created:

```
{
  "name": "T03",
  "description": "Test Grade 3"
}
```

Authorization

This interface is a private interface that mutates the data in Core Standards and is intended for the Core Standards user interface. It requires a user that is authenticated and authorized to access the Core Standards user interface.

Request Body

Do not supply a request body for this API.

Response

This API returns a JSON object that describes the status of the request and the name and description of the grade created. This API fails if the name of the grade already exists.

Examples

Example	Response Code
<pre>REQ: POST /Grade REQ PARAM: { "name": "T03", "description": "Test Grade 3" } RESP: { "status": "success", "errors": [], "warnings": [], "validationsPassed": [],],</pre>	200

Example	Response Code
<pre>"payload": { "key": null, "name": "T03", "description": "Test Grade 3" }</pre>	

Table 25. Grade (POST) API Examples

Staging Publication (POST)

Overview

This API copies a publication into staging tables for editing.

HTTP Request

```
POST staging/publication
```

URL Parameters

No URL parameters are required for this API.

Request Parameters

A JSON object with the key of the existing publication to copy into staging tables:

```
{
  "key": "SBAC-TST-v1"
}
```

Authorization

This interface is a private interface that mutates the data in Core Standards and is intended for the Core Standards user interface. It requires a user that is authenticated and authorized to access the Core Standards user interface.

Request Body

Do not supply a request body for this API.

Response

This API returns a JSON object that describes the key of the publication copied into the staging tables. This API fails if the publication provided doesn't exist.

Examples

Example	Response Code
<pre>REQ: POST /staging/publication REQ PARAM: { "key": "SBAC-TST-v1" } RESP: { "status": "success", "errors": [], "warnings": [], "validationsPassed": [], "payload": { "key": "SBAC-TST-v1",</pre>	<p>200</p>

Example	Response Code
<pre> "fkPublisher": null, "version": 0, "description": null, "fkSubject": null, "subjectLabel": null, "status": null } </pre>	

Table 26. Staging Publication (POST) API Examples

Staging Publication (PUT)

Overview

This API makes publication in staging tables editable.

HTTP Request

```
PUT staging/publication
```

URL Parameters

No URL parameters are required for this API.

Request Parameters

A JSON object indicating the publication to make editable in the staging tables:

```
{
  "key": "SBAC-TST-v1"
}
```

Authorization

This interface is a private interface that mutates the data in Core Standards and is intended for the Core Standards user interface. It requires a user that is authenticated and authorized to access the Core Standards user interface.

Request Body

Do not supply a request body for this API.

Response

This API returns a JSON object indicating the success or failure of the loading and making the specified publication editable in the loader tables.

Examples

Example	Response Code
<pre>REQ: PUT /staging publication REQ PARAM: { "key": "SBAC-TST-v1" } RESP: { "status": "success", "errors": [], "warnings": [], "validationsPassed": [], "payload": { "key": "SBAC-TST-v1",</pre>	<p>200</p>

Example	Response Code
<pre> "fkPublisher": null, "version": 0, "description": null, "fkSubject": null, "subjectLabel": null, "status": null } </pre>	

Table 27. Staging Publication (PUT) API Examples

Staging Publication NextVersion (GET)

Overview

This API retrieves the next version for a given publication in the staging tables. This API is used to support the user interface by determining the next version of a publication that is being edited in the staging tables.

HTTP Request

```
GET staging/publication/nextVersion
```

URL Parameters

No URL parameters are required for this API.

Request Parameters

All parameters are provided on the command line for this API. No request parameters are required.

Authorization

This interface is a private interface that is intended for the Core Standards user interface. It requires a user that is authenticated and authorized to access the Core Standards user interface.

Request Body

Do not supply a request body for this API.

Response

This API returns a JSON object that provides information about the next version that can be saved for the publication in the staging tables.

Examples

Example	Response Code
<pre>REQ: GET /staging/publication/nextVersion RESP: { "status": "success", "errors": [], "warnings": [], "validationsPassed": [], "payload": "2.0" }</pre>	<p>200</p>

Table 28. Staging Publication NextVersion (GET) API Examples

Staging Publication (GET)

Overview

This API obtains publication metadata about a publication in the staging tables. This API is used to support the user interface to render publication information.

HTTP Request

```
GET staging/publication
```

URL Parameters

No URL parameters are required for this API.

Request Parameters

All parameters are provided on the command line for this API. No request parameters are required.

Authorization

This interface is a private interface that is intended for the Core Standards user interface. It requires a user that is authenticated and authorized to access the Core Standards user interface.

Request Body

Do not supply a request body for this API.

Response

This API returns a JSON object that provides publication information about the publication in the staging tables.

Examples

Example	Response Code
<pre>REQ: GET /staging/publication RESP: { "status": "success", "errors": [], "warnings": [], "validationsPassed": [], "payload": { "key": "SBAC-MA-v1", "fkPublisher": null, "version": 0, "description": null, "fkSubject": null, "subjectLabel": null, "status": null } }</pre>	200

Table 29. Staging Publication (GET) API Examples

Staging Category (GET)

Overview

This API obtains information about the hierarchical categories of a publication in the staging tables. This API is used to support the user interface to render publication information.

HTTP Request

```
GET staging/category
```

URL Parameters

No URL parameters are required for this API.

Request Parameters

All parameters are provided on the command line for this API. No request parameters are required.

Authorization

This interface is a private interface that is intended for the Core Standards user interface. It requires a user that is authenticated and authorized to access the Core Standards user interface.

Request Body

Do not supply a request body for this API.

Response

This API returns a JSON object that provides information about the hierarchical categories of a publication in the staging tables. It reflects any new category information entered by the user when using the user interface to edit the category names and labels.

Examples

Example	Response Code
<pre>REQ: GET /staging/category RESP: { "status": "success", "errors": [], "warnings": [], "validationsPassed": [], "payload": [{ "category": "Claim", "treeLevel": 1, "publicationKey": null, "sessionKey": null, "newCategory": null }] }</pre>	<p>200</p>

Example	Response Code
<pre> }, { "category": "Content Domain", "treeLevel": 2, "publicationKey": null, "sessionKey": null, "newCategory": null }, { "category": "Target", "treeLevel": 3, "publicationKey": null, "sessionKey": null, "newCategory": null }, { "category": "Emphasis", "treeLevel": 4, "publicationKey": null, "sessionKey": null, "newCategory": null }, { "category": "Common Core Standard", "treeLevel": 5, "publicationKey": null, "sessionKey": null, "newCategory": null }] </pre>	

Table 30. Staging Category (GET) API Examples

Staging Grade (GET)

Overview

This interface is a private interface that mutates the data in Core Standards and is intended for the Core Standards user interface. It requires a user that is authenticated and authorized to access the Core Standards user interface.

HTTP Request

GET staging/grade

URL Parameters

No URL parameters are required for this API.

Request Parameters

All parameters are provided on the command line for this API. No request parameters are required.

Authorization

This interface is a private interface that is intended for the Core Standards user interface. It requires a user that is authenticated and authorized to access the Core Standards user interface.

Request Body

Do not supply a request body for this API.

Response

This API returns grade alignment information for all the lowest level standards in the publication in the staging tables.

Examples

Example	Response Code
<pre>REQ: GET staging/grade RESP: { "status": "success", "errors": [], "warnings": [], "validationsPassed": [], "payload": [{ "benchmark": "1 G K-1 na 1.G.1", "grade": "1", "sessionKey": null, "newGrade": null }], }</pre>	200

Example	Response Code
<pre><repeat for each one of the lowest level standards>] }</pre>	

Table 31. Staging Grade (GET) API Examples

Staging Standard (GET)

Overview

This API obtains standards information for a publication in staging tables. It is used to support the user interface for rendering of standards information for the publication in the staging tables.

HTTP Request

```
GET staging/standard
```

URL Parameters

No URL parameters are required for this API.

Request Parameters

All parameters are provided on the command line for this API. No request parameters are required.

Authorization

This interface is a private interface that is intended for the Core Standards user interface. It requires a user that is authenticated and authorized to access the Core Standards user interface.

Request Body

Do not supply a request body for this API.

Response

This API returns a JSON object that represents the standards for the standards publication in the staging tables.

Examples

Example	Response Code
<pre>REQ: GET /staging/standards RESP: { "status": "success", "errors": [], "warnings": [], "validationsPassed": [], "payload": [{ "key": "1", "name": "1", "fkParent": null, "fkPublication": null, "description": "Concepts and Procedures - Students can explain and apply mathematical concepts and interpret and carry out mathematical procedures with precision and fluency.?"</pre>	200

Example	Response Code
<pre> "treeLevel": 1, "fkGradeLevel": null, "shortName": "Concepts and Procedures" }, <repeat for every standard in the publication>] }</pre>	

Table 32. Staging Standard (GET) API Examples

Staging SOCK (GET)

Overview

This interface is a private interface that mutates the data in Core Standards and is intended for the Core Standards user interface. It requires a user that is authenticated and authorized to access the Core Standards user interface.

HTTP Request

```
GET staging/sock
```

URL Parameters

No URL parameters are required for this API.

Request Parameters

All parameters are provided on the command line for this API. No request parameters are required.

Authorization

This interface is a private interface that is intended for the Core Standards user interface. It requires a user that is authenticated and authorized to access the Core Standards user interface.

Request Body

Do not supply a request body for this API.

Response

This API returns a JSON object that represents the SOCK (Some Other Category of Knowledge) for the standards publication in the staging tables.

Examples

Example	Response Code
<pre>REQ: GET /staging/sock RESP: { "status": "success", "errors": [], "warnings": [], "validationsPassed": [], "payload": [{ "knowledgeCategory": "Depth of Knowledge Level 1", "description": "Depth of Knowledge Level 1 (Description)", "socksKey": null, "sessionKey": null, "newKnowledgeCategory": null, }] }</pre>	<p>200</p>

Example	Response Code
<pre> "newDescription": null }, <repeat for every SOCK in the publication>] } </pre>	

Table 33. Staging Sock (GET) API Examples

Staging Category (PUT)

Overview

This interface is a private interface that mutates the data in Core Standards and is intended for the Core Standards user interface. It requires a user that is authenticated and authorized to access the Core Standards user interface.

HTTP Request

PUT staging/category

URL Parameters

No URL parameters are required for this API.

Request Parameters

A JSON object that represents the updated category information provided by the user via the user interface:

```
{
  "fkPublisher": "SBAC",
  "description": "Test Publication with SOCK data",
  "fkSubject": "Test Subject",
  "version": "2"
}
```

Authorization

This interface is a private interface that mutates the data in Core Standards and is intended for the Core Standards user interface. It requires a user that is authenticated and authorized to access the Core Standards user interface.

Request Body

Do not supply a request body for this API.

Response

This API returns a JSON object with the updated category information provided by the user.

Examples

Example	Response Code
REQ: PUT staging/category RESP: { "status": "success", "errors": [], "warnings": [], "validationsPassed": [],	200

Example	Response Code
<pre> "payload": [{ "category": "Common Core Standards", "treeLevel": 3, "publicationKey": null, "sessionKey": null, "newCategory": null }] </pre>	

Table 34. Staging Category (PUT) API Examples

Staging Grade (PUT)

Overview

This API edits grade information in staging tables. It is invoked when the user enters a change in the user interface, and the user interface invokes this API providing a JSON request parameter with the modified information.

HTTP Request

PUT staging/grade

URL Parameters

No URL parameters are required for this API.

Request Parameters

A JSON object that represents the updated grade information provided by the user via the user interface:

```
{
  "benchmark": "L1-A|L2-A|L3-10",
  "grade": "10",
  "newGrade": "11"
}
```

Authorization

This interface is a private interface that mutates the data in Core Standards and is intended for the Core Standards user interface. It requires a user that is authenticated and authorized to access the Core Standards user interface.

Request Body

Do not supply a request body for this API.

Response

This API returns a JSON object with the updated grade information provided by the user.

Examples

Example	Response Code
REQ: PUT /staging/grade RESP: { "status": "success", "errors": [], "warnings": [], "validationsPassed": [], "payload": []	200

Example	Response Code
<pre> { "benchmark": "L1-A L2-A L3-10", "grade": "11", "sessionKey": null, "newGrade": null }] } </pre>	

Table 35. Staging Grade (PUT) API Examples

Staging Standard (PUT)

Overview

This API edits standards information in staging tables. It is invoked when the user enters a change in the user interface, and the user interface invokes this API providing a JSON request parameter with the modified information.

HTTP Request

```
PUT staging/standard
```

URL Parameters

No URL parameters are required for this API.

Request Parameters

A JSON object that represents the updated standard information provided by the user via the user interface:

```
{
  "treeLevel": "3",
  "key": "L1-A|L2-B|L3-6",
  "name": "L3-6",
  "description": "Level A Sublevel B Grade 6.5",
  "shortName": "Level A Sublevel B Grade 6"
}
```

Authorization

This interface is a private interface that mutates the data in Core Standards and is intended for the Core Standards user interface. It requires a user that is authenticated and authorized to access the Core Standards user interface.

Request Body

Do not supply a request body for this API.

Response

This API returns a JSON object with the updated standards information provided by the user.

Examples

Example	Response Code
REQ: PUT /staging/standard RESP: { "status": "success", "errors": [], "warnings": [], "validationsPassed": []	200

Example	Response Code
<pre>], "payload": [{ "key": "L1-A", "name": "L1-A", "fkParent": null, "fkPublication": null, "description": "Level A", "treeLevel": 1, "fkGradeLevel": null, "shortName": "Level A" }, <repeat for every standard>] } </pre>	

Table 36. Staging Standard (PUT) API Examples

Staging SOCK (PUT)

Overview

This API edits SOCK information in staging tables. It is invoked when the user enters a change in the user interface, and the user interface invokes this API providing a JSON request parameter with the modified information.

HTTP Request

PUT staging/sock

URL Parameters

No URL parameters are required for this API.

Request Parameters

A JSON object that represents the updated standard information provided by the user via the user interface:

```
{
  "knowledgeCategory": "Literary",
  "description": "Literary (Description)",
  "newKnowledgeCategory": "Literary",
  "newDescription": "Literary (Updated Description)"
}
```

Authorization

This interface is a private interface that mutates the data in Core Standards and is intended for the Core Standards user interface. It requires a user that is authenticated and authorized to access the Core Standards user interface.

Request Body

Do not supply a request body for this API.

Response

This API returns a JSON object with SOCK information updated by the user.

Examples

Example	Response Code
REQ: PUT /staging/sock RESP: { "status": "success", "errors": [], "warnings": [], "validationsPassed": [],	200

Example	Response Code
<pre> "payload": [{ "knowledgeCategory": "Depth of Knowledge Level 1", "description": "Depth of Knowledge Level 1 (Description)", "socksKey": null, "sessionKey": null, "newKnowledgeCategory": null, "newDescription": null }, <repeat for every SOCK>] </pre>	

Table 37. Staging SOCK (PUT) API Examples

Staging (DELETE)

Overview

This API deletes the current publication in the staging tables. It does not affect the publications stored in the main publication tables. This API is invoked when the user has made edits to a publication in the staging tables and chooses to cancel the edits.

HTTP Request

```
DELETE /staging
```

URL Parameters

No URL parameters are required for this API.

Request Parameters

No request parameters are required for this API.

Authorization

This interface is a private interface that mutates the data in Core Standards and is intended for the Core Standards user interface. It requires a user that is authenticated and authorized to access the Core Standards user interface.

Request Body

Do not supply a request body for this API.

Response

A JSON object that indicates the success or failure of the DELETE operation from the staging tables.

Examples

Example	Response Code
REQ: DELETE /staging RESP: { "status": "success", "errors": [], "warnings": [], "validationsPassed": [], "payload": null }	200

Table 38. Staging (DELETE) API Examples

Publication (POST)

Overview

This API Moves a publication from the staging tables into the main tables.

HTTP Request

POST publication

URL Parameters

No URL parameters are required for this API.

Request Parameters

All parameters are provided on the command line for this API. No request parameters are required.

Authorization

This interface is a private interface that mutates the data in Core Standards and is intended for the Core Standards user interface. It requires a user that is authenticated and authorized to access the Core Standards user interface.

Request Body

Do not supply a request body for this API.

Response

This API returns a JSON object that describes the success or failure of the move from the staging table to the main database tables.

Examples

Example	Response Code
REQ: POST /publication RESP: { "status": "success", "errors": [], "warnings": [], "validationsPassed": [], "payload": [] }	200

Table 39. Publication (POST) API Examples

