Nile Engineering: Scoped Search

Objective

This document captures the requirement and design for scoped search api.

Requirement

Search APIs enhancement -

- Search should enable the user to select "what they want to search for" (filters) and "where they want to search for it" (targets)
    - Filters: text is OPTIONAL, then standards, audiences, grade levels
        - Filtering by standards should bring up the nice drill-down tree of standards and let them choose & unchoose standards
        - Also requires Content Type Filter: Courses, Lessons, Units, Collections, Assessments...
    - Targets should be a dial up-down from:
        - My Content
        - Subtenant library or multiple libraries
        - Tenant library or multiple libraries
        - Specific accessible library or multiple libraries
        - Open Gooru Featured Courses
        - All of open Gooru

Target / Scope To Be Supported

Search from which kind of contents/ targets/ scope should be supported at search.

Scope can be from below,

1. My Content
2. Subtenant library (s) if multiple
3. Tenant library (s) if multiple
4. Open Featured Courses
5. All of open contents

Enhancements in Search

Indexing Support

Indexing support for original resource reference :

Currently search supports original resources. For mycontent search support, references of original resources should also be indexed.

Add support at event based and direct indexing to index resource references.

Indexing support for library contents :

Currently, library contents are not indexed and used at search, need to index library contents to support search from library contents.

While indexing contents, check if the content is part of any library and set library flag to support filter for library contents. when requested to search from api caller's tenant library, we can search for contents which are part of some library under api caller's tenant - assuming a content will be part of only one library, tenant of both library and content should be same.

Also index associated library names for library name based content search (may be required in future)

Support at Search API

V3- API : **Request body to support scoped search:**

Key	**Data Type**	**Default Value**	**Allowed values**	**Mandatory**	**Comment**
scope.key	Object	null	my-content, tenant-library, subtenant-library, open-library, open-featured, open-all *Defer:* tenant-content,	No	

			subtenant-content		
scope.idList	Array	null	UUID	No	
scope.targetNames	Array	null	Featured course title / content title / library short name	No	To be specified when one wants to search from a particular **featured course** or from **contents having requested titles** or from a particular **library** which has requested **short name**

V2-API : **Request Param to Pass for scoped search**:

Key	**Data Type**	**Default Value**	**Allowed values**	**Mandatory**	**Comment**
scopeKey	String		my-content, tenant-library, subtenant-library, open-library, open-featured, open-all	Yes	
scopeTargetNames	String		Featured course title / content title / library short name. To pass multiple target name use tilde ('~') to separate them.	Yes	To be specified when one wants to search from a particular featured course or from contents having requested titles or from a particular library which has requested short name

Scope Functionality :

When scope parameter is not passed :

Behavior should be same as current search.

When scope parameter is passed :

Use appropriate filter to search from requested target(s).

scope:my-content :

Request body:

```
{
        "scope":{
                "key":"my-content"
        }
}
```

Search from contents which are created by api caller, irrespective of publish status of content.

scope:tenant-library :

Request body:

```
{
        "scope":{
                "key":"tenant-library",
                "targetNames":[
                        "sausd”
                ]
        }
}
```

Search from API caller’s tenant library contents, if scope.targetNames is provided, search from those libraries.

Apply filter to search for contents which are part of api caller’s tenant and has isLibraryContent flag as true, irrespective of publish status of content..

scope:subtenant-library :

Request body:

```
{
        "scope":{
                "key":"subtenant-library",
                "targetNames":[
                        "sausd”
                ]
        }
}
```

Search from API caller tenant’s root tenant library contents, if scope.targetNames is provided, search from those libraries. We may need to indicate whether to rank sub-tenant libraries ahead of tenant libraries or otherwise.

Apply filter to search for contents which are part of root tenant of api caller’s tenant and has isLibraryContent flag as true, irrespective of publish status of content.

scope:open-library :

Request body:

```
{
        "scope":{
                "key":"open-library",
                "targetNames":[
                        "sausd”
                ]
        }
}
```

Search from library contents, if scope.targetNames is provided, search from those libraries. Apply filter to search for contents which has isLibraryContent flag as true, irrespective of publish status of content.

Sample request :

```
curl -i  -H 'Authorization: Token
MjoxNTI5NTcyMTIwNDYzOmFub255bW91czo6YmE5NTZhOTctYWUxNS0xMWU1LWEzMDItZjhhOTYz
MDY1OTc2' -H 'Content-Type: application/json' -XPOST
'http://nile-dev.gooru.org/gooru-search/rest/v3/search/collection?q=food&flt.grade=5' -d
'{"scope":{"key":"open-library", "targetNames":["sausd"]}}'
```

scope:open-featured :

Request body:

```
{
        "scope":{
                "key":"open-featured",
                "targetNames":[
                        "Integrated Science Phenomena and Solutions”
                ]
        }
}
```

Search from open tenant featured courses, where content may be published or unpublished as not all content of featured course will be published.

scope:open-all :

Request body:

```
{
        "scope":{
                "key":"open-all"
        }
}
```

Search open/ published contents from user & open tenant.