Google Summer Of Code 2022

Organization: **Cuneiform Digital Library Initiative.**

Project Idea: **Improve the credits ownership and licencing of digital assets**

Name: CHIDIEBERE CHUKWUDI
Gitlab Username: @jovialcore
Email: chididevelopper@gmail.com
Location: Bauchi, Nigeria
Time Zone: GMT+1

This project is targeted at updating how digital assets' license, ownership and credits are stored, while considering how they can be displayed in a more detailed way.

Considerations for the system:

1. Manage single(Individuals) or multiple(institution) assets owners
2. Manage Credits for individuals or institutions
    a. Under consideration no 2, we will also have to factor in the **type of credit** . For example, Some credits will not be part of the official citation, like crediting the person operating the capture device who might not be the owner of the assets but **courtesy credits** can be extended to the person.
3. Manage Licenses: We will consider Generic licenses, custom licenses, say in the form of a text string or a web link.

4. Other: Display links on the web page on procedures for asking for rights to collections.

APPROACH

1. **Table schema for the new credit system data**

Since the framework (project) already has images table that represents the digital assets,there will be a relationship between the credits table and the images table. The following is a draft table schema and a cakephp migration schema for the proposed credit system.

Credits Table Schema:

- The table columns for the are as follows:
  image_id int i.id
- license_type enum('Generic','custom')
- license_info string
- owner_type enum('Individual','Organization')
- owner_info string

- created varchar
- modified varchar

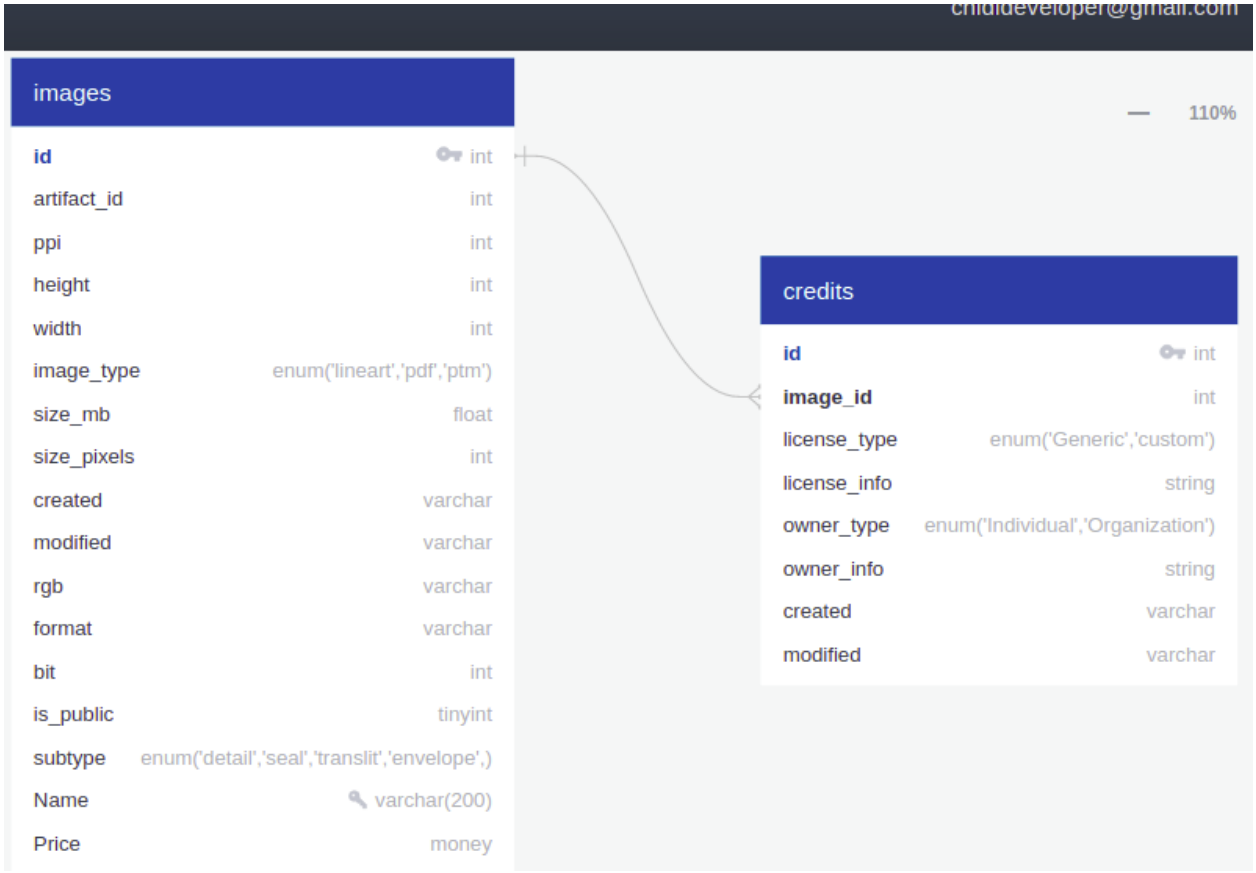


Table schema I created representing the credit system table and the relationship with images table.
Link to the table diagram schema for full information:
https://app.quickdatabasediagrams.com/#/d/7wNCT7

Below is my proposed cakephp migration schema for the credit system table:

```php
/**
 * Change Method.
 *
 * More information on this method is available here:
 * https://book.cakephp.org/phinx/0/en/migrations.html#the-change-metho
 * @return void
 */
public function change()
{
    $table = $this->table('digital_asessts');

    $table->addColumn('id', 'integer', [
        'autoIncrement' => true,
    ]);

    $table->addColumn('image_id', 'integer', [
        'null' => false,
    ]);

    $table->addColumn('license_type', 'enum', [ // generic, custom, etc
        'values' => ['generic', 'custom'],
        'null' => false,
    ]);
```

```
]);

$table->addColumn('license_type', 'enum', [ // generic, custom, etc
    'values' => ['generic', 'custom'],
    'null' => false,
]);

$table->addColumn('license_info', 'string', [
    'limit' => 255,
    'null' => false,
]);

$table->addColumn('owner_type', 'enum', [ // individual or  organisa
    'values' => ['individual', 'organisation'],
    'null' => false,
]);

$table->addColumn('owner_info', 'string', [ // owner information
    'null' => false,
]);


$table->create();
```

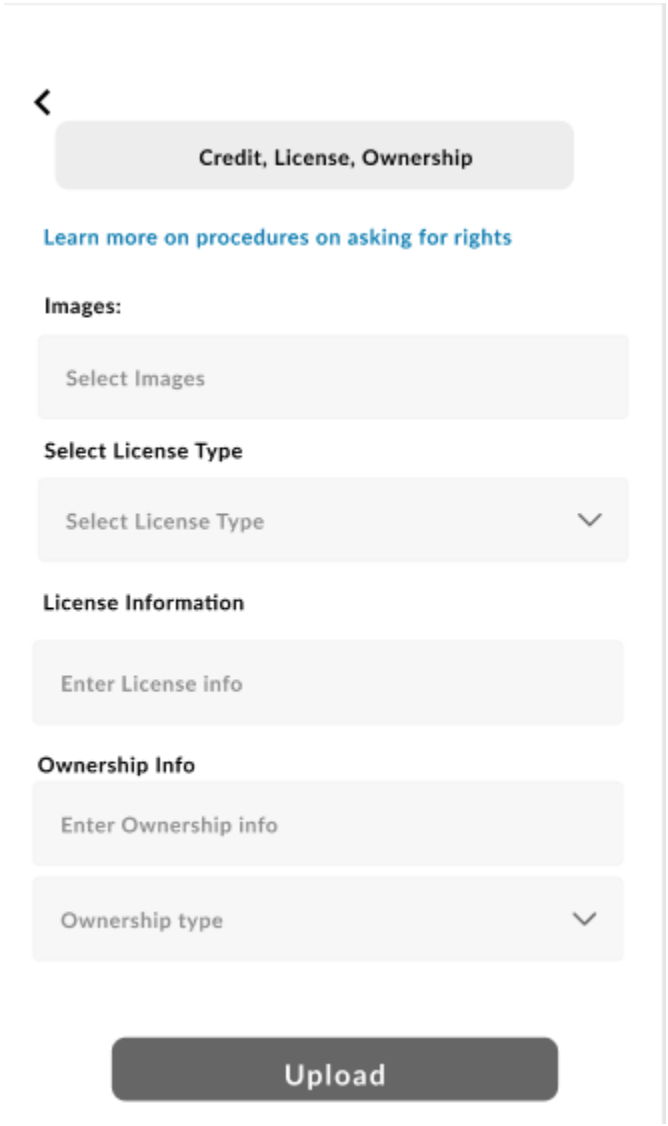
2. **Extending the API**
   Here, I will write the REST api to extend the credits api data to be available to other machines.

3. **Extend the interface to display the information regarding the credits system where appropriate (all the places where a digital asset can be displayed) :**
   Here, I will work on the frontend of the interface responsible for displaying digital assets. Since I have good knowledge of html, css, and javascript, I will be meticulous enough to make sure that the data I will be supplying from my controller to the views template are correct

   Also, I will collaborate with the design 1 challenge person to feed the cite button for digital assets.

4. **Implement edition forms and bulk upload feature for Updating and deletion of credit data for digital assets.**

   My approach: I designed a Hi-Fi prototype to Illustrate my approach to the implementation of this aspect of the idea. Other iterations will be added subsequently for this section.



Since the idealist for this project is in cakephp, I will be writing the code to make sure that the Images and corresponding information are correctly uploaded in the images table. I will also make sure that the relationship

between the credits table and images table are consistent using the ideal foreign keys.

Timeline:

Phase 0: Pre-community bonding period (Present - May 20th)

- Work on current issues present in the CDLI framework .
- Take a glance at cakephp documentation
- Explore the framework and post new issues encountered

Phase 1: Community bonding period (May 20th - June 13)

- Discuss with the mentor about the implementation in detail.
- Document the workflow for better understanding of the implementation approach.
- Keep working on the issues on the framework already while preparing to work on my proposal.

Phase 2: Coding Phase 1 (13th June - 25th July)

Weeks	Tasks	Deliverables
Week 1 (13th June - 20th june)	● Reevaluate the table schema to fully represent the targeted objectives. ● Reevaluate the other necessary table relationships that we should have	Perfect database schema for credit system implementation

	between the credit tables and other tables. ● Implement the migration schema following cakephp conventions.	
Week 2 - week 3 (20th June - 4th July)	● Using cakephp, I will prepare the models and other business logic for the project idea. ● Start writing the logic in the controller for passing the right data to the views/template/interface for displaying credits. ● Write the logic for saving the necessary credit information into the database.	Basic Controller Logic for the idea coded.
Week 4 - Week 5 (4th July - 18th July)	● **Extend THE API** Understand and study the already existing api codebase for the different aspects of the project.	Extended API For additional information regarding digital assets, based on search results and based on a single artifact

	• Write a rest API using cakephp for machines to access the additional information regarding digital assets for **search results.** • Write a rest API using cakephp for machines to access the additional information regarding digital assets for **single artifacts.**	
Week 6 (18th July to 25th July)	• Buffer week for completing all the remaining tasks in phase 1 • Fix all the issues developed in phase 1. • Blog post for phase 1.	Final Testing of work Flow

Phase 3: Coding Phase 2 (25th July - Sept 4th)

Weeks	Tasks	Deliverables
Week 1 (25th July - 1st August)	• Work on the views/template for the places where a digital asset can be displayed. • Make adjustments so that the credit information displayed will align properly with the figma design and adjustments will not distort the interface aesthetics.	Extended interface for credit system information to be displayed for all digital assets.
Week 2 - Week 3(1st Aug - 15th Aug)	• Work with the designer (for design challenge 1) as stated in the idea objectives to feed the cite button for digital assets. • Code the edit form views/template.	• Accurate Cite button and provision of good user experience for the credit system. • Edit form and bulk upload functionality Implemented

	• Implement bulk Images upload functionality	
Week 4 - 5 (15th Aug - 29 Aug)	• Implement edition and deletion of credit data for digital assets • Extend the credit system to properly manage extra information for the 3d models. eg. capture equipment and other information	• Delete and Edit Feature for data implemented • Granular information for 3d models for credit system.
Week 5 (29th Aug - 5th Sept)	• Buffer week for pending deliverables. • Prepare the final project report. • Submit the final code.	• Final submission of report, code and blog post

WHY ME ?

Previous Knowledge of MVC php framework (Laravel)
One of the major programming tools for Cuneiform Digital Library Initiative is cakephp, as someone who has good knowledge of laravel php framework, cakephp will definitely be “a piece of cake” for me. I have already gone through some part of cakephp documentation and I can confirm it’s easy to understand because of my previous tech background.

Knowledge of Frontend Stack
Some objectives of my chosen project idea require that the contributor not only know cakephp but also html, css and maybe javascript.
Hence, I have production knowledge of html, css and know some level of javascript plus javascript framework like Vuejs.

Here is my portfolio that describes my stacks and some of the projects I have built around it. (jovialcore.netlify.app)

I am enthusiastic and passionate for technical writing:
GSOC '2022 requires that contributors write blog posts at some point during the programme. I'm glad that I will be leveraging this opportunity as I will be sharing my experience on my own blog (www.jovialcore.tech) amongst other blog posts I have written.

I’m Interested in Cuneiform Digital Library Initiative:
Interacting with mentors and working on issues on the Cuneiform Digital Library Initiative pre-community bonding, I’m stunned at how people like me give their time to open source projects. It portrays the faith they have in the usefulness of the project hence, I will love to be part of the community of dedicated open sourcers. To further prove this, here is my merged contribution to one of the issues on the Cuneiform Digital Library repository.
https://gitlab.com/cdli/framework/-/merge_requests/602