**Start Page Overview**

The start page of the CRC 1625 portal is the gateway to a centralized platform for managing, organizing, and sharing research data related to the **Collaborative Research Center 1625**. The page is designed to be intuitive, offering quick access to the platform’s core functionalities.

**Main Content**

The center of the start page is where most of the action happens:

* Welcome Message: A brief introduction explains the purpose of the portal and how it can assist researchers.
* Search Bar: A simple yet powerful search tool lets users look for specific data, rubrics, or objects stored in the system. You can type keywords or apply filters for more precise results.
* Highlighted Features: Key sections highlight what the platform can do, like:
  + Data management tools.
  + FAIR compliance for research data.
  + Integration with advanced computational tools.
* Quick Links: Shortcut buttons guide users to frequent actions such as:
  + Creating new objects or samples.
  + Uploading files.
  + Managing containers and compositions.

**How to Use the Start Page**

* New Users: Start by clicking on Register to create an account. Once registered, log in to explore the portal’s features.
* Returning Users: Log in and use the search bar to find specific datasets or navigate to sections like rubrics or file uploads.
* General Visitors: Without an account, you can still browse public information and learn about the CRC 1625 project.

**Behind the Scenes**

* The website is built with modern web technologies like React.js (frontend) and Django (backend), ensuring a fast and secure experience.
* Data Storage: All data is securely stored in an SQL Server database, making it easy to manage rubrics, objects, and related files.
* Security: The platform uses token-based authentication and role-based access controls, ensuring that only authorized users can access sensitive data.

The first section of the platform is **Area A01** (or any other area, accessible via the URL /a01). In this area, users have the following capabilities:

1. **Add Containers**: Users can create containers that can hold multiple libraries, organizing related resources in one place.
2. **Add Ideas and Plans**: Users can document their ideas and plans directly in the system, enabling collaborative brainstorming and planning.
3. **Add Objects**: Users can create and save objects with detailed metadata, ensuring they are properly cataloged and searchable.
4. **Upload Files**: Files can be uploaded and automatically linked to objects, with all data securely saved in the database for future reference.

This structure allows for efficient data organization and seamless management of research-related resources.

* The container will be stored in the **Rubricinfo** table in the database, ensuring its hierarchical structure and metadata are properly managed and easily accessible.

|  |  |  |
| --- | --- | --- |
| Field Name | Data to Save | Notes |
| rubricid | New unique ID generated (max\_rubric\_id + 1). | Auto-incremented based on the maximum rubricid in the database. |
| tenantid | Tenant object (tenant\_id). | 4 for crc 1625 |
| field\_created | Current timestamp. | Time of creation |
| field\_createdby | The user who created it. | The user who created it. |
| field\_updated | Time of update | Time of update |
| field\_updatedby | The user who updated it. | The user who updated it. |
| typeid | Type object (type\_info). | Foreign key to the Typeinfo table, which is equal to 2. |
| parentid | Parent rubric ID. | Foreign key to the Rubricinfo table. Nullable for root rubrics. |
| level | Level in the hierarchy. | Count of } in rubricpath. |
| leafflag | Boolean flag indicating if the rubric is a leaf. | Default value: True. |
| flags |  |  |
| sortcode | Place of the object in the list tree | Default value: 0. |
| accesscontrol | The private or public | Mapped from ACCESS\_CONTROL\_MAP (e.g., public = 0, protected = 1, etc.). |
| ispublished | Publication status. | Default value: False. |
| rubricname | Name of the rubric. | The name from the user |
| rubricnameurl | URL-safe name of the rubric. | Slugified combination of rubric\_name and name (e.g., "sub-category-example"). |
| rubricpath | Hierarchical path of the rubric | Will be like this }  Area A}A01}Pd-Pt |

The **Add Object** and **Upload** buttons will serve as tools for creating objects, such as the **APT object type** and others. These features enable users to create new objects, upload files, and save associated data directly to the database, complete with all relevant metadata for proper organization and accessibility.

|  |  |  |
| --- | --- | --- |
| Field Name | Description | Description |
| objectid | Primary key for the object. | Primary key for the object. |
| tenantid | Links to Tenant model which equal to 4. | Links to Tenant model which equal to 4. |
| field\_created | Timestamp of object creation. | Time of creation |
| field\_createdby | User who created the object, links to Aspnetusers. | User who created the object, links to Aspnetusers. |
| field\_updated | Timestamp of last update. | Timestamp of last update. |
| field\_updatedby | User who last updated the object, links to Aspnetusers. | User who last updated the object, links to Aspnetusers. |
| typeid | Links to Typeinfo model. | Links to Typeinfo model. |
| rubricid | Rubric object (rubric) | Foreign key to Rubricinfo, nullable for objects with no rubric. |
| sortcode | Sorting order for the object. | Sorting order for the object. |
| accesscontrol | Access control level: 0 (public), 1 (protected), 2 (protected NDA), 3 (private). | Access control level: 0 (public), 1 (protected), 2 (protected NDA), 3 (private). |
| ispublished | Indicates if the object is published. | Indicates if the object is published. |
| externalid | External identifier for the object. | Optional field. |
| objectname | Name of the object (e.g., "Sample 1"). | User-provided. |
| objectnameurl | URL-safe name of the object. | The name of the object-objectid like rh\_on\_15nm\_ta\_photo-64903. |
| objectfilepath | Path to the saved file. | Saved under BASE\_FILE\_PATH. |
| objectfilehash | MD5 hash of the file for duplicate checking. | MD5 hash of the file for duplicate checking. |
| objectdescription | Description of the object. | Optional field. |

When creating an object of type **Sample**, the system will add entries to the following tables:

1. **Sample Table**: Stores specific details about the sample, such as the number of elements and their elements.
2. **Objectinfo Table**: Saves general metadata about the object, such as its name, description, and associated files.
3. **Property Table**: Records additional properties or attributes of the sample, such as tolerance values or other relevant parameters.
4. **substrate**: Optional, links the sample to an existing substrate.

This ensures that all aspects of the sample are properly documented and linked within the database.

The entries in the **Objectinfo** table will be saved as usual, with the only differences being in the fields **objectname** and **objectnameurl**, which will be customized based on the specific sample being created.

|  |  |  |
| --- | --- | --- |
| Field | Data to Save | Notes |
| objectname | {objectid} {name} | Name of the object (e.g., "123 Sample 1"). |
| objectnameurl | {name}-{objectid} | URL-safe name (e.g., "Sample-1-123"). |

Data will also be inserted into the **Sample** table, capturing details specific to the sample, such as the number of elements.

|  |  |  |
| --- | --- | --- |
| Field | Data to Save | Notes |
| sampleid | Foreign key to Objectinfo | Same as objectid of the created object. |
| elemnumber | Number of elements | Integer, calculated from the elements. |
| elements | Chemical system (elements) | String of elements separated by hyphens (e.g., Ag-Au). |

**Integer properties linked to the sample object.**

|  |  |  |
| --- | --- | --- |
| Field | Data to Save | Notes |
| objectid | Foreign key to Objectinfo | Same as objectid of the created object. |
| propertyname | Name of the property | Provided in the intProperties list. |
| value | Integer value | Taken from intProperties (value). |
| row | Row number (if provided) | Optional, taken from intProperties. |
| comment | Comment (if provided) | Optional, taken from intProperties. |
| field\_created | Current timestamp | Set to timezone.now(). |
| field\_createdby | User object (created\_by) | Foreign key to Aspnetusers. |

**Float properties linked to the sample object.**

|  |  |  |
| --- | --- | --- |
| Field | Data to Save | Notes |
| objectid | Foreign key to Objectinfo | Same as objectid of the created object. |
| propertyname | Name of the property | Provided in the floatProperties list. |
| value | Float value | Taken from intProperties (value). |
| valueepsilon | Optional epsilon value | Taken from floatProperties (valueEpsilon). |
| row | Row number (if provided) | Optional, taken from intProperties. |
| comment | Comment (if provided) | Optional, taken from floatProperties. |
| field\_created | Current timestamp | Set to timezone.now(). |
| field\_createdby | User object (created\_by) | Foreign key to Aspnetusers. |

**Links the sample object to an existing substrate object.**

|  |  |  |
| --- | --- | --- |
| Field | Data to Save | Notes |
| objectid | Linked to the Objectinfo entry. | Foreign key linking to the new Objectinfo entry. |
| linkedobjectid | Linked to the substrate object (substrate\_id). | Foreign key linking to the existing substrate. |
| linktypeobjectid | Unique ID (max\_objectlinkobjectid + 1). | Auto-incremented identifier for the link. |
| sortcode | Default value: 0 | Integer field for sorting. |
| field\_created | Current timestamp | Set to timezone.now(). |
| field\_createdby | User object (created\_by) | Foreign key to Aspnetusers. |

**When saving an EDX CSV file, here's what happens:**

1. Main EDX File:
   * The main EDX file is saved in the Objectinfo table as the primary record.
2. Child Objects:
   * Each row in the CSV, representing a measurement area, is saved as a separate child object in the Objectinfo table.
3. Measurement Area Properties:
   * The measurement areas are also saved as properties in the database, linking the main EDX file to the corresponding child objects.
4. Container:
   * A container is created to group and organize all the measurement areas, ensuring everything is stored in a structured way.
5. Samples and Compositions:
   * The elements from each row in the CSV are saved as a sample in the Sample table.
   * Their corresponding percentages are saved in the Composition table.
6. Linked Objects:
   * The main EDX object and its child objects are linked using the Objectlinkobject table.
   * These links create relationships between the primary file, measurement areas, and related samples, enabling easy navigation and structured data access.
   * Also, the Edx is linked from the beginning to a sample object.

This process ensures that all data from the EDX file is properly stored, organized, linked, and accessible for analysis.

The entries in the **Objectinfo** table will be saved as usual, with the only differences being in the fields **objectname** and **objectnameurl**, which will be customized based on the specific sample being created.

|  |  |  |
| --- | --- | --- |
| Field | Data to Save | Notes |
| objectname | "00{sampleid}\_{object\_name}\_EDX" | Formatting the sample ID with leading zeros and appending the object name with \_EDX. |
| objectnameurl | "00{sampleid}\_{object\_name}\_EDX-{objectid}" | URL-safe name (e.g., "Sample-1-123"). |

The entries in the **Rubricinfo** table will be saved as usual, with the only differences being in the fieldsbelow.

|  |  |  |
| --- | --- | --- |
| Field Name | Data to Save | Notes |
| parentid | Parent rubric ID. | Foreign key to the Rubricinfo table. Nullable for root rubrics. |
| level | Level in the hierarchy. | Count of } in rubricpath. |
| ispublished | Publication status. | Default value: False. |
| rubricname | "{sampleid} Measurement Areas" | Name of the rubric. |
| rubricnameurl | {rubric\_nameurl of the parent}\_{sampleid}-measurement-areas | Slugified combination of rubric\_name and name (e.g., "sub-category-example"). |
| rubricpath | {rubric\_path of parent}}{rubricname} | Will be like this, Area A}A01}Ag-Au-Cu-Pd-Pt}10403 Measurement Areas |

**The Sample will be created for each rows and the composition table.**

|  |  |  |
| --- | --- | --- |
| Field Name | Data to Save | Notes |
| compositionid | Unique ID (max\_compositionid + 1). | Auto-incremented identifier for the composition entry. |
| sampleid | Linked to the sample. | Foreign key linking to the Sample table. |
| elementname | Name of the element (e.g., Ag). | Extracted from the file's columns. |
| valueabsolute | Absolute value of the element. | Calculated as value\_percent \* 0.01. |
| valuepercent | Percentage value of the element. | Extracted from the file. |

**A Literature Reference** or **Publication** creates entries in both the Objectinfo table for general metadata and the Reference table for literature-specific details like authors, title, journal, and DOI.

|  |  |  |
| --- | --- | --- |
| Field Name | Data to Save | Notes |
| referenceid | Linked to the Objectinfo entry. | Foreign key to the Objectinfo table. |
| authors | Authors of the reference. | Provided as a comma-separated list. |
| title | Title of the reference. | Required to create a reference. |
| journal | Name of the journal. | Optional. |
| year | Publication year. | Integer value. |
| volume | Journal volume number. | Optional. |
| number | Journal issue number. | Optional. |
| startpage | Starting page of the reference. | Optional. |
| endpage | Ending page of the reference. | Optional. |
| doi | Digital Object Identifier (DOI). | Optional. |
| url | URL for the reference. | Optional. |
| bibtex | BibTeX citation data. | User-provided. |

**The Special object also,**

Request for Synthesis

Calculation/Computational Composition ------- **Percentage and Chemical System**

Composition List ------- **Percentage and Chemical System**

Literature Reference List

Ideas or experiment plans List

Calculation/Computational Sample List ------- **Chemical System**

Computational Composition Atom ------- **Chemical System**

Composition Test ------- **Chemical System**