

GSRS 3.1 Release Notes

December 2023

Table Contents

[Overview 3](#_Toc152312876)

[Highlighted New Features and Improvements 3](#_Toc152312877)

[New Feature: Data Exchange - Data Import 3](#_Toc152312878)

[New Feature: User Saved Lists 9](#_Toc152312879)

[Major Improvement: Structure Rending Improvements 14](#_Toc152312880)

[New Feature: Create Developer SOP for Database Schema Changes 15](#_Toc152312881)

[Major Improvement: Validation Message Customization 17](#_Toc152312882)

[New Feature: Create Load Scripts to Populate Empty Product, Application and CT 18](#_Toc152312883)

[Major Improvement: Impurities Module Improvement 19](#_Toc152312884)

[Other Issues Addressed 20](#_Toc152312885)

[New Feature: Quick Registration Tools 20](#_Toc152312886)

[Improvement: Structure Search Quality of Life Improvements 21](#_Toc152312887)

[Improvement: Product Module Support for Multiple Provenances 22](#_Toc152312888)

[Major Improvement: Impurities Module Improvement 25](#_Toc152312889)

[New Feature: Specified Substance Group 4 Manufacturing (SSG4M) (**Preview Release**) 28](#_Toc152312890)

[Other selected Improvements and bug fixes: 31](#_Toc152312891)

# Overview

GSRS 3.1 software release is a major release incorporating new features in data exchange, which introduces staging area for importing, user saved lists, and structure rending Improvements in substances module. It also includes new features and approvements in Product, Application, Clinical Trials, Impurities models.

Highlighted new features include:

* Data Exchange - Data Import
* User saved lists
* Structure Rending Improvements (Custom Graphics/Thumbnails)
* Create Developer SOP for Database Schema Changes
* Create Load Scripts to Populate Empty Product, Application and CT

Highlighted improvements and bug fixes include:

* Validation Message Customization
* Impurities Module Improvement

# Highlighted New Features and Improvements

## New Feature: Data Exchange - Data Import

In GSRS 3.0.3, we expanded the user’s ability to export data by providing limited data cleaning and expansion of a set of records to include other records that help define the original records, for example, adding component substances to the export set that already included a mixture substance.

In GSRS 3.1, we provide a robust data import feature that allows you to select an SD file or .gsrs, compressed, multi-record JSON file for import.

GSRS 3.1 allows you to specify how the fields in an SD file map to the data fields in GSRS.

Also, data can be loaded into a temporary section of GSRS, the Staging Area, where it can be reviewed, edited, deleted or moved to the main database.

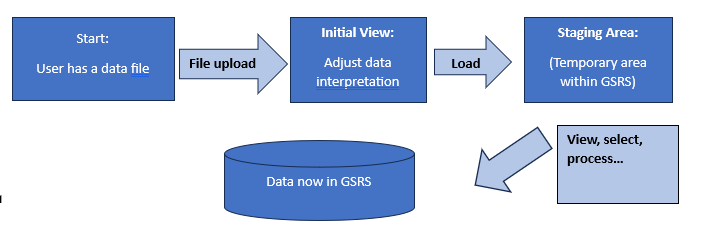


Figure 1-1 – Full Export Options Dialog

**Purpose and Motivation:**

SD files are a common means of data exchange for chemists. An SD file may include a structure, associated IDs, names, and properties for each record of a multiple-record set. A GSRS user may wish to load the data into GSRS to add to the existing database.

More information about SD files is available here:

<https://en.wikipedia.org/wiki/Chemical_table_file>

<https://discover.3ds.com/ctfile-documentation-request-form>

**How it works:**

Make sure you are logged into GSRS as an administrative user.

Select the Admin Panel from the drop-down menu at the top of the screen, on the right side.

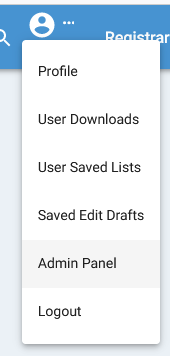


Figure 1-2 – Drop-down menu list

Select the ‘Data Import’ tab on the Admin Panel.

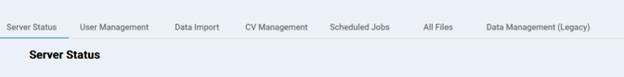
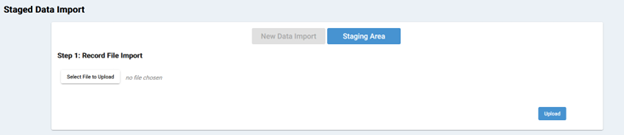


Figure 1-3 – Admin Panel tabs

Use the ‘Select File to Import’ button to bring up an open file dialog box:



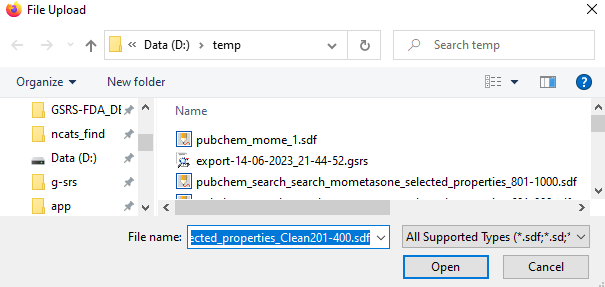


Figure 1-4 – File Import selection and dialog box

Select an SD file.

The system now recognizes that you have selected an SD file and selects the appropriate adapter. (Note: if, for some reason, the system selects an incorrect adapter, you can change the selection of adapter by selecting the appropriate radio button.)

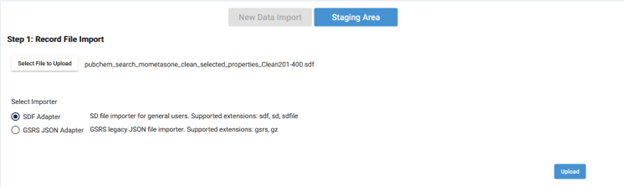


Figure 1-5 – File import radio button selection

Click the Upload button.

The system now attempts to guess the appropriate processing action for each field in your file:

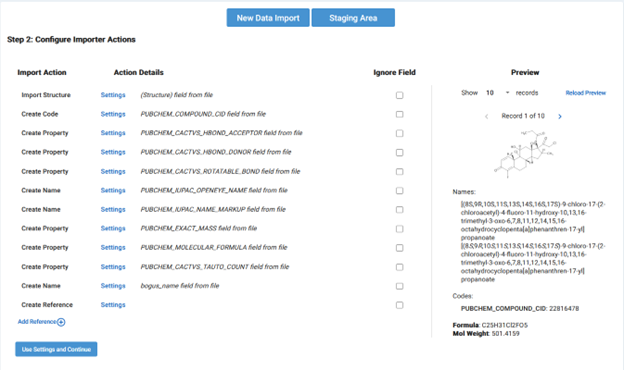


Figure 1-6 – Import Action list

Each field is listed along with the selected action and a hotlink to change the action.

For example, the file field PUBCHEM\_CACTVS\_ROTATABLE\_BOND field is recognized as a GSRS property.



Figure 1-7 – GSRS Property

The preview panel on the right side of the screen shows how the data would look if the current set of processing instructions were followed.

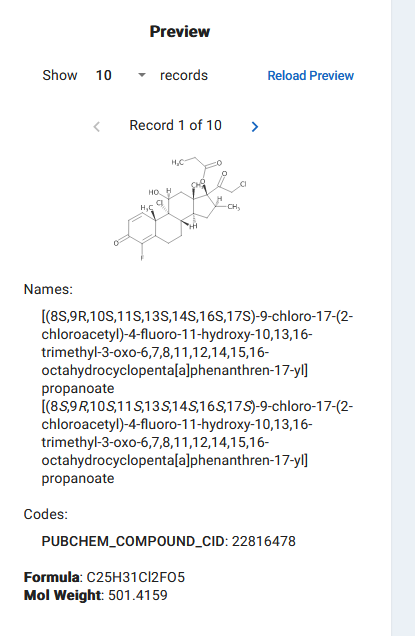


Figure 1-8 – File Import Preview

Clicking on the Settings link brings up a dialog box where you can change the way the field is mapped to GSRS fields:

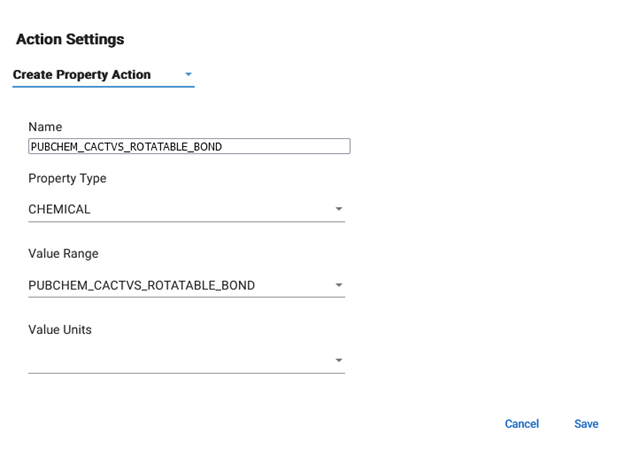


Figure 1-9 – Action Settings field mapping

Once the data look the way you intend them to look, click ‘Use Setting and Continue.’

At this point, GSRS reads each record in your file, applies the mapping actions you selected and creates GSRS substances. These substances are validated, compared to the records already in your GSRS database, stored within a temporary database (the ‘staging area’ as explained below) and indexed for searching.

Depending on the size of your file and the provisioning of your server, this process can take a while.

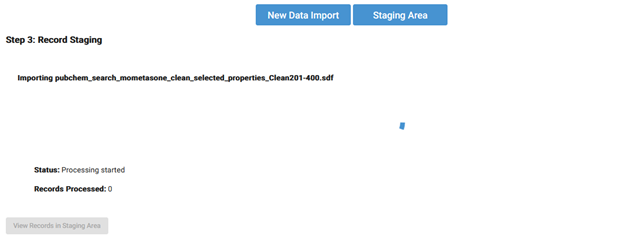


Figure 1-10 – File Import Processing

There are other options for data import that are discussed in a separate Data Import Users Guide.

**Notes for the future:**

In the future, GSRS will consider the following

* Providing an easy way to merge data from records in the staging area into records in the main database.
* Import of additional file types
* Adding more options to the import process
* Extending the options to other entities

## New Feature: User Saved Lists

**Purpose and Motivation:**

Users need a way to group substances together for later use. They also need to have the capability to modify this group, that is, to add more substances to the group, or remove some substances out of the group. Once they are done with the group of substances, they will not be needed anymore. Users can then remove it from the system. In this release, we add this new feature – user saved lists for these needs.

**How it works:**

**1. Access user saved lists:**   
You can access your user saved lists either from the “User List” facet, or from the user account drop-down menu. Note that if you do not have any user saved list created, the “User List” facet would not show up.

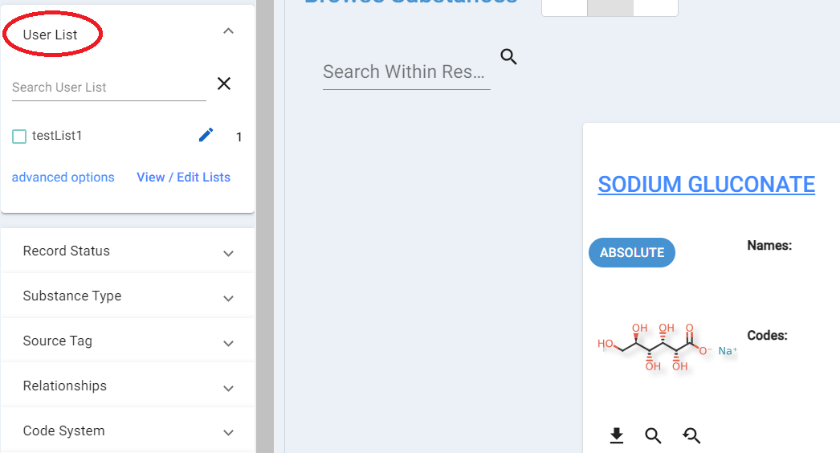


Figure 2-1 Accessing User Saved Lists from Facets

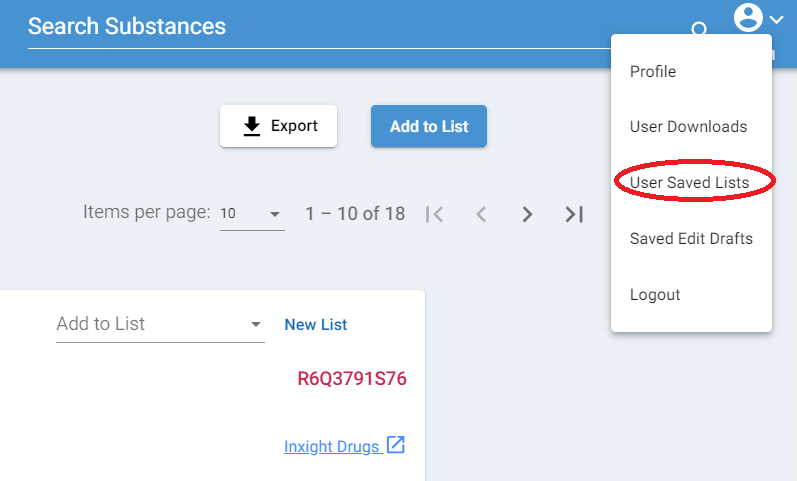


Figure 2-2 Accessing User Saved Lists from Menu

**2. Create a new user saved list**

You can create a user saved list from each substance by clicking on “New List” like shown in the figure below. A popup window will show up to let you input the name of the list.

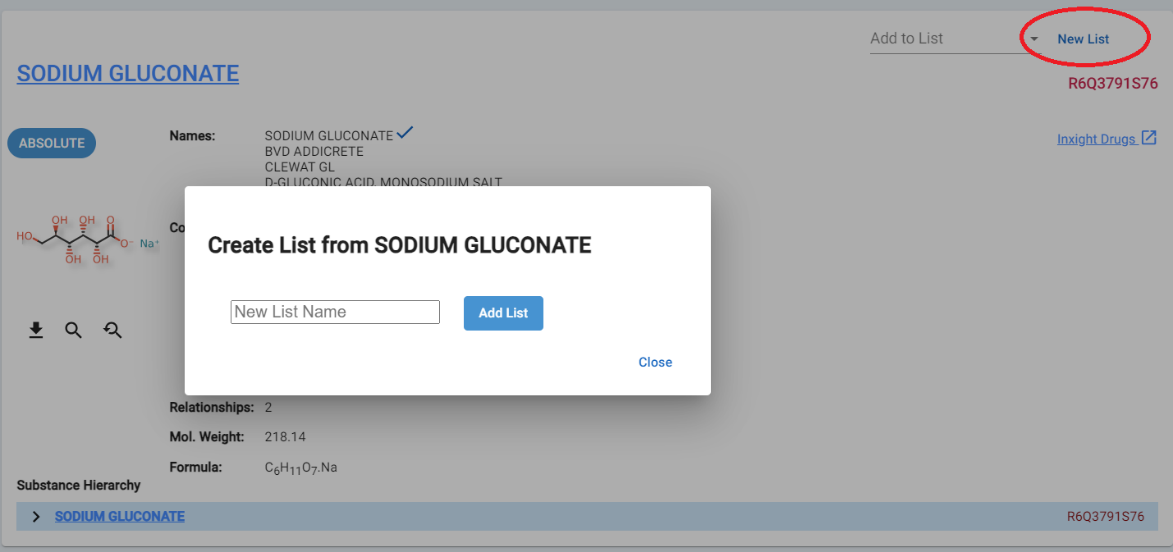


Figure 2-3 Creating a User Saved List

Another way to create a user saved list is to do a search first, then save all the result substances to a user saved list by clicking on “Add to List”.

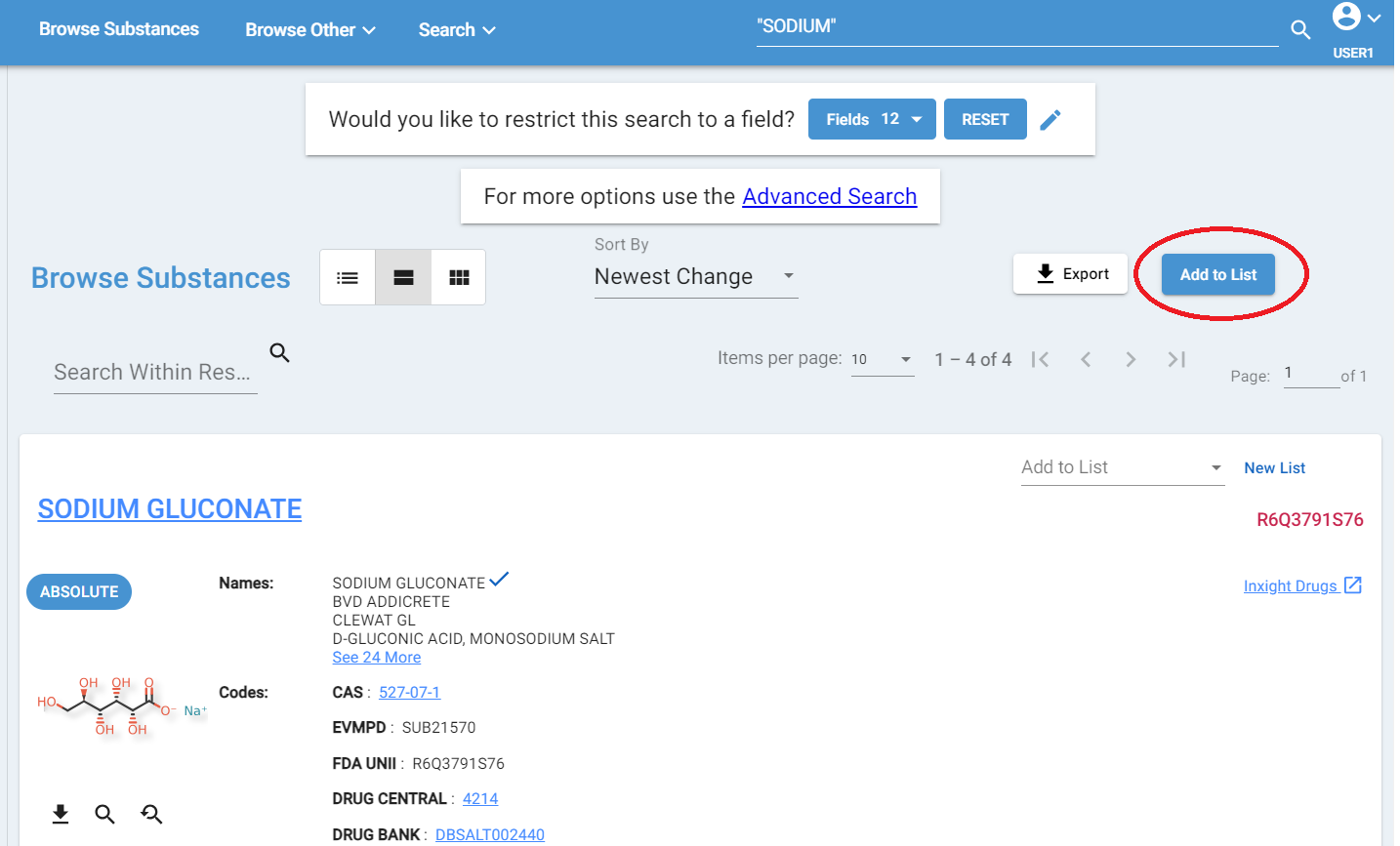


Figure 2-4 Adding to a User Saved List

A popup window will show up and provide users the options to create a new user saved list or append the current search result to an existing user saved list.

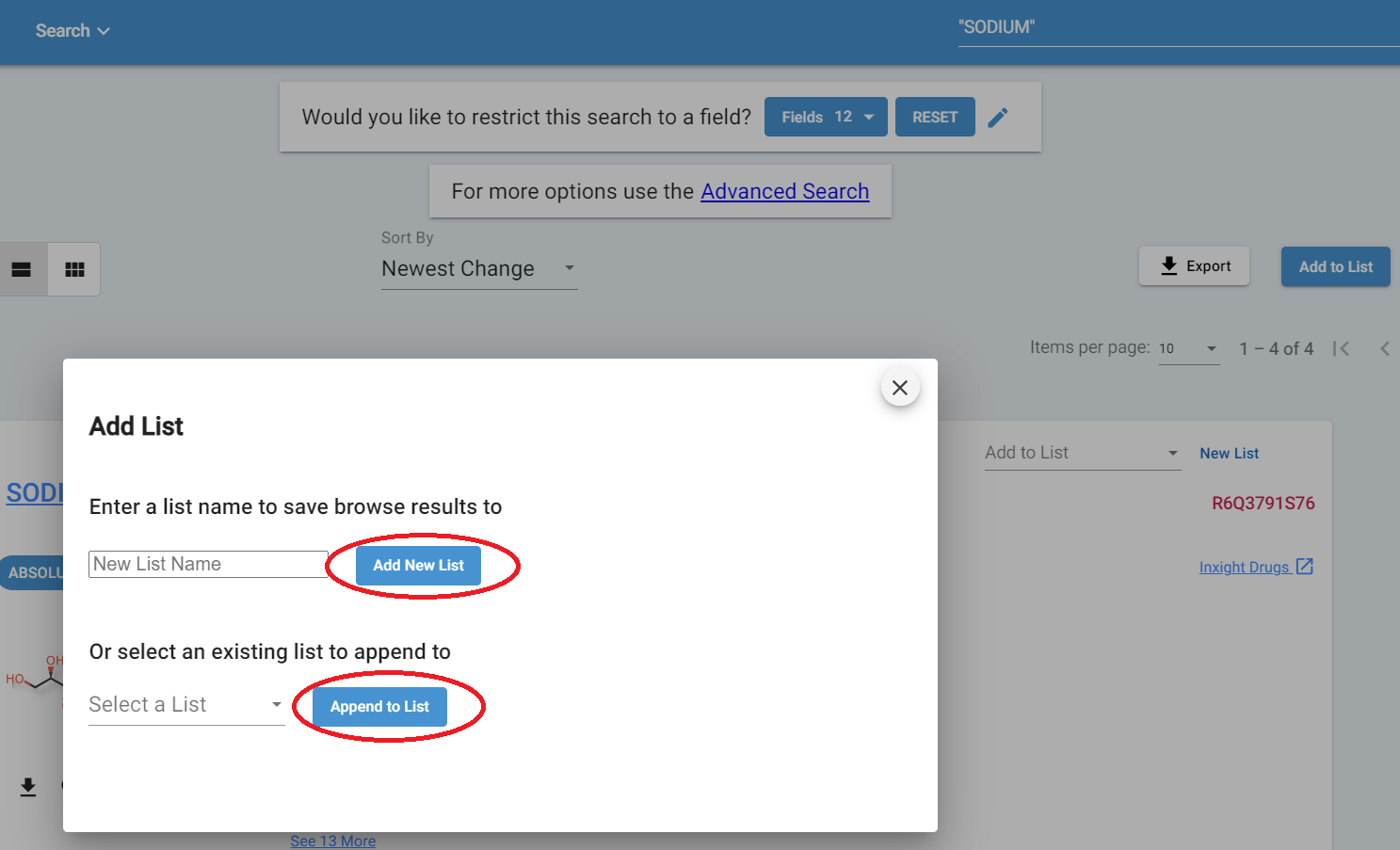


Figure 2-5 Adding to a User Saved List – Options

**3. Add a substance or substances to an existing user saved list**

After you get a search result, you can also add the result substances to an existing list by clicking on the button “Append to List” shown in figure #. While browsing substances, you can add a substance to an existing list by clicking on the “Add to List” as showed in the figure below.

Graphical user interface, text, application

Description automatically generated  
Figure 2-6 Adding to a User Saved List from Card

**4. Remove a substance or substances from a user saved list**

From the “User List”facet, or the user account drop down menu, you can find the list you need to update. Use the trashcan icon as shown in the figure below to delete one or more substances.

Graphical user interface, application, Teams

Description automatically generated

Figure 2-7 Removing content from a User Saved List and Export List

**5. Export a user saved list**

To export a user saved list, go to the edit page of the list. Click on “Export List” as show in the figure above. A window will pop up and ask for the location to save the export file. The user saved list will be saved as a Json file. Note the Json file does not contain all the fields of the substance. It only contains the four fields as shown in the popup window.   
  
**6. Import from an exported user saved list**

User saved lists can be imported from a Json file, in the same format as the exported Json file we can get from the system. It provides a way to share lists with other users. One user can export a user list and another user can import the user list using the exported Json file. Clicking on “Saved List Import” as shown in figure, a popup window will show up as shown in figure. You can specify the import file and the name of the user saved list and go from there.

Graphical user interface, application, Teams

Description automatically generated

Figure 2-8 Import and Removing a User Saved List

**Graphical user interface, text, application, email

Description automatically generated**

Figure 2-9 Importing a User Saved List

**7. Delete a user saved list**

You can delete the entire user saved list once you do not need it by clicking on the trashcan icon as shown in figure 2-8 Import and Removing a User Saved List.

**Notes for the Future:**

The team will explore other user stories of user saved lists and add more useful features.

Major Improvement: Structure Rending Improvements

**Purpose and Motivation:**

Substances of types such as protein, nucleic acid, mixture, structurally diverse, etc. are generally displayed with generic icons when browsing substances.

It is now possible to designate a graphic file to use when displaying an individual substance of any type.

**How it works:**

Create a new reference on the substance. Designate the type as ‘Image Reference’

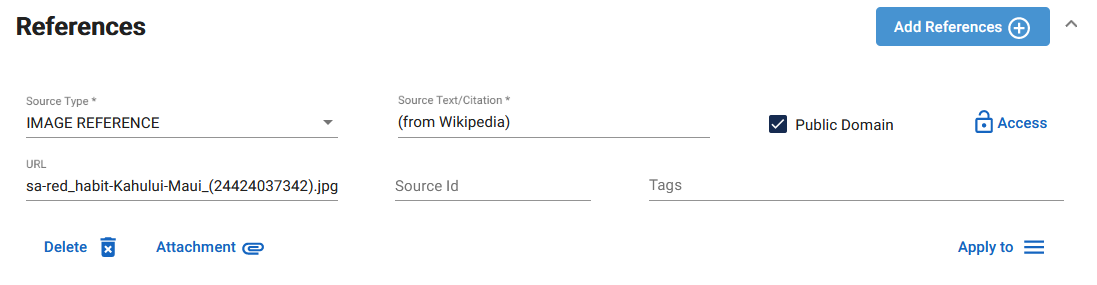


Figure 3-1 – Image Reference

Add an attachment, in the form of an image file (SVG, JPG or PNG).

Save the substance.

From now on, when you browse results that include substance, you will see the image you uploaded with the reference.

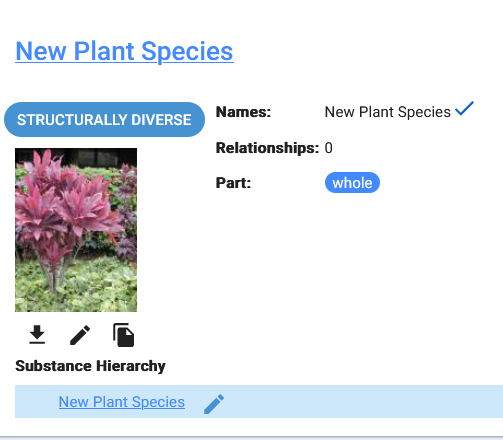


Figure 3-1 – Image Reference Display

Image from https://en.wikipedia.org/wiki/Cordyline\_fruticosa#/media/File:Starr-010420-0119-Cordyline\_fruticosa-red\_habit-Kahului-Maui\_(24424037342).jpg

**Notes for the Future:**

The team will explore other user stories of substance display and add more useful features.

## New Feature: Create Developer SOP for Database Schema Changes

**Purpose and Motivation:**

With new features and improvements introduced to GSRS system with each release, database schema changes are inevitable to accommodate these changes. Developers might need to make changes to the database schema objects, such as creating a new table, modifying an existing table, or adding a new constraint or an index, etc. Database schema changes need to be handled carefully for the production system to prevent losing data in the database or leaving the database in an inconsistent state. GSRS is an open-source project and allows customized extensions, so an organization could have its own customized database changes. We need to set up and a process of changing database schema, testing, and deploying the changes to existing database and verify it.

**How it works:**

We provide the following items for database schema changes.   
1. In each release, we will put the database schema change scripts and README in a folder named using the release version. For each of the four supported database flavors, it includes:

* A full database schema SQL script, which generates the entire GSRS schema for a blank database. This is useful for brand new GSRS installations.
* A delta database schema SQL script, which modifies the database from a previous version of GSRS to work with the new version. This is useful when upgrading an existing GSRS installation.
* Other SQL scripts needed. We manage the database GSRS version ourselves for now. The script that inserts rows to the version table is part of this.
* README describes how to use these scripts to update the database to test and deploy.

They are located at   
[**https://github.com/ncats/gsrs3-main-deployment/tree/main/substances/database/sql**](https://github.com/ncats/gsrs3-main-deployment/tree/main/substances/database/sql)

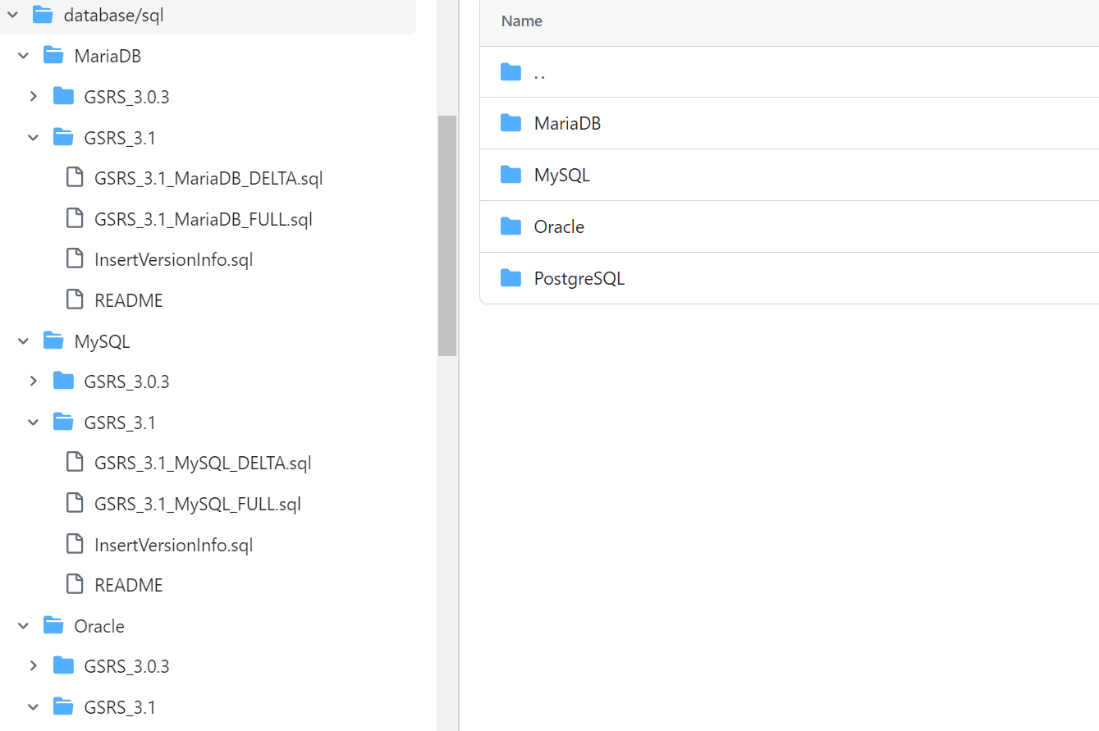


Figure 4-1 – Schema Change Scripts and READMEs

2. We provide a developer SOP document to describe how to generate the items mentioned above.   
It is located:   
<https://github.com/ncats/gsrs-ci/tree/gsrs-example-deployment/docs>

**Notes for the Future:**

The team will research and investigate database migration tools, such as Flyway and Liquibase, to decide whether we should use one of those migration tools.

## Major Improvement: Validation Message Customization

**Purpose and Motivation:**

GSRS allows validation rules to be specified which will either prevent a user from entering data that is considered invalid or will warn the user about a potential data issue. The validation rules allowed specifying a severity level, with “Errors” preventing the submission and “Warnings” being shown to the user but not preventing the submission. The default behavior is also to store the warnings as notes inside the records. While this works, there are 2 areas where additional functionality would be useful.

First, there are some kinds of warnings which should be shown to the user but should not be stored as notes. A new severity level called “notice” is used for this purpose which allows an additional severity level to avoid having too many notes.

Secondly, there are cases where a specific configuration or deployment may want to *override* the default severity levels of a validation message based on local business rules or user privileges. This new feature allows existing validation rules to be modified within the configuration file instead of requiring recompiling. This is especially useful for cases where, for example, a “duplicate check” validation rule should be considered an “error” for most users, but only a “warning” for significantly privileged expert users who should be allowed to override such a message.

**How it works:**

The configuration files (for example application.conf) can have a defined “processing strategy” which will allow configuration changes for validation rules. An example is given below. This specific example overrides the validation rule which checks for substance uniqueness and makes it so that users who have a “super” role see a WARNING instead of a “ERROR”. The “regex” field can be a regular expression for the message text *or* can be the unique ID of the validation message (as it is here). These unique IDs are given by the REST API validation responses and are calculatable via the message type and message. The first character of the ID is the severity (E=”error”, W=”warning”, N=”notice”, and I=”info”), the next 3 characters are for the java class which reports the validation rule (this is the first 3 digits of the java hashcode of that class name), and the last 4 digits are for the validation message itself (this is a java hashcode of the validation message String).

|  |
| --- |
| # This feature allows one to filter on validation message ids or message texts to  # for example change a warning to a notice for users with a given role.  # OverrideRule(s) could look like this:  # {"regex": "W.\*", "userRoles": ["Approver","Admin"], "newMessageType": "NOTICE"}  # In this sample rule, the regex determines if the messageId begins with W, implying  # a warning.  gsrs.processing-strategy = {  "defaultStrategy": "ACCEPT\_APPLY\_ALL",  "overrideRules": [  # SubstanceUniquenessValidator  {"regex": "E4562650", "userRoles": ["SuperUpdate", "SuperDataEntry"], "newMessageType": "WARNING"}  ]  } |

Additional examples of override rules for validation messages are shown below:

|  |
| --- |
| #This would make all warning messages notices instead  {"regex": "W[0-9]{7}", "newMessageType": "NOTICE"}   #This would make any message that contains the word “duplicate” become an error  {"regex": ".\*duplicate.\*", "newMessageType": "ERROR"}  #This would make any message that is generated by the Validator Class  #SubstanceUniquenessValidator (warning, error or notice) become an “INFO” severity level  {"regex": "[WEN]456.\*", "newMessageType": "INFO"} |

**Notes for the future:**

The addition of having unique identifiers for the validation rules is quite useful generally, and it can be expanded in the future for localization, customization, and even more detailed workflow management. In the future, we intend to make the validation message identifiers more discoverable and use them in more sophisticated ways.

While this is currently primarily used for substance entities, there is theoretically no significant barrier to using the same approach for all entity validation schemes.

## New Feature: Create Load Scripts to Populate Empty Product, Application and CT

**Purpose and Motivation:**

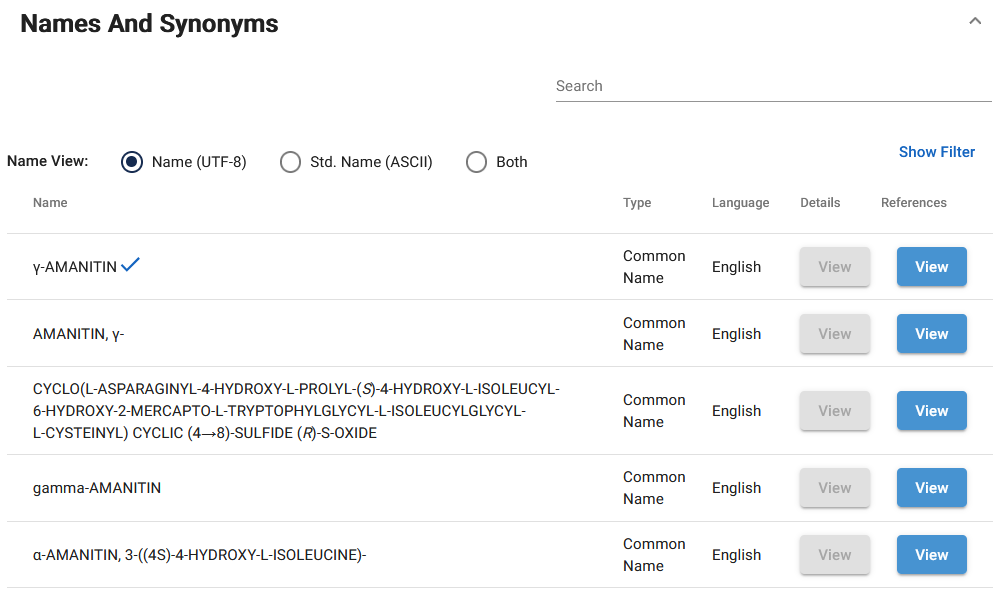


Figure 5-1 – Names and Synonyms

Initially, GSRS provided support for ASCII text-based names but, as chemical information experts know, substance names commonly include Greek letters, superscripts, italic letters and other non-ASCII characters. With this release, GSRS allows users to enter names that include superscripts, subscripts, small caps and italics as HTML markup, as well as non-ASCII characters.

**How it works:**

When entering a substance name, you can now provide some HTML markup to control display of the name. (For security reasons, we limit the HTML tags we support to: superscript, subscript, italics and small caps.) You can also use Greek letters and other non-ASCII characters.

Provided your database system supports the characters entered (check your database documentation or your database administrator), the characters you entered will be preserved and used in the display of your substance within GSRS.

**Notes for the future:**

We intend to remove HTML tags from standardized names (separate from the main ‘name’ field).

Please let the GSRS team know if you have additional HTML tags in mind.

## Major Improvement: Impurities Module Improvement

**Purpose and Motivation:**

Each organization that uses GSRS may wish to customize the appearance and basic assets such as the logo and some key display terms.

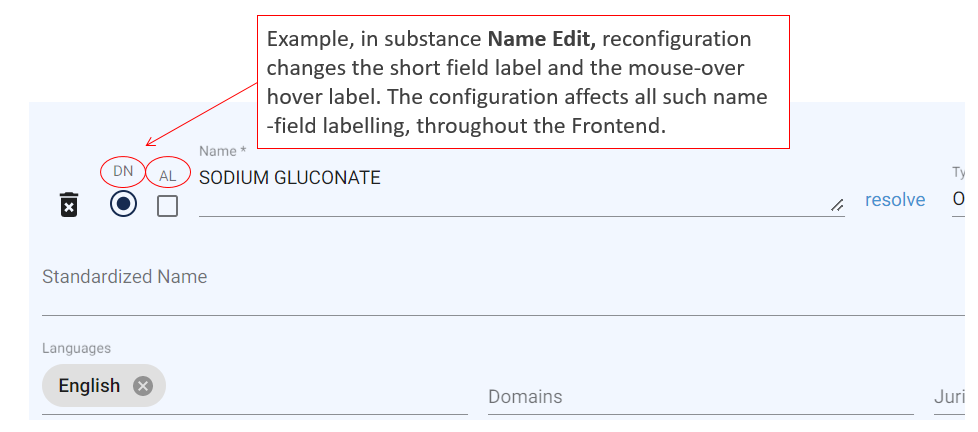
**How it works:**

A) We put some basic styling such as header color, link color and button color in a central location. In the angular code base, see the file ‘src/styles/\_styles.css’. If you wish for the link color to be different throughout your GSRS instance, change it in the `\_styles.css` file. Then, in your angular component’s SCSS file, reference the color variable instead of hard coding a color. For example, this is preferred:

|  |
| --- |
| .mylink {  color: var(--link-color);  } |

B) We made certain frontend substance display term labels configurable. Go to the frontend configuration and find or add this section and for example change “Display Name” to “My Display Name”. Now, “My Display Name” shows as the label on the substance overview and edit pages, amongst other places.

|  |
| --- |
| "elementLabelDisplay": {  "labels": {  "substance\_names\_name": {  "displayNameTitle": "Display Name",  "displayNameShortTitle":"DN",  "preferredTitle": "Additional Listing Name",  "preferredShortTitle": "AL"  }  } |

Figure 6-1 - Frontend Configurable Substance Name Field Labels

**Notes for the Future:**

This is part of an ongoing effort to improve front-end customizability, both with and without requiring a custom build of the front-end.

# Other Issues Addressed

## New Feature: Quick Registration Tools

**Purpose and Motivation:**

In order to speed up the process of discovering missing and registering new substances, we have added the ability to seed an edit form URL with the SMILES of a structure, preventing registrars from having to re-draw a structure from either a hard-coded URL or embedded link.

**How it works:**

Structure search results now include a link with the label ‘Or start a new chemical registration using this SMILES’ Clicking or sharing this link will take you to a new chemical registration form with that structure already added to the editor.

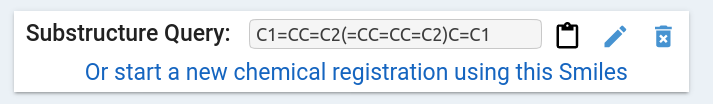


Figure 7-1 - Frontend Configurable Substance Name Field Labels

**Notes for the Future:**

Along with planned changes to speed up registration further, this feature will be extended to molfiles as well using an encoded url JSON string, which will allow seeding a registration form with a more detailed stub, which may include names, codes and references.

## Improvement: Structure Search Quality of Life Improvements

**Purpose and Motivation:**

In the GSRS 3.0 rewrite the chemical structure search and sequence search features lost the default sort order of similarity. Due to this, the search results would typically be returned based on last-edited status, with other common sort orders available in the dropdown. While users could still find the “most” similar records by adjusting the similarity cutoff, not having the option to sort on similarity was noticeably confusing for users.

**How it works:**

Now the default sort order for chemical similarity and sequence similarity is the similarity score (descending). The order can be changed in the “sort by” drop down, with “relevance” always meaning “similarity”.

**Notes for the Future:**

Additional improvements to quality of life for structure and sequence search are planned. These include making it easier to qualify structure searches to switch to specific stereochemistry, flex/substructure and to make the loading of large searches faster and have fewer page refreshes.

## Improvement: Product Module Support for Multiple Provenances

**Purpose and Motivation:**

In GSRS 3.1 release, the Product module is redesigned to store multiple provenances for each Product along with many more fields are added to capture relevant data. The provenance is a dropdown fields with values such as GSRS, HEALTH CANADA, or any other source from where the data is getting imported. The updated Product module is more improved and robust in maintaining Product data, which can be shared and exchanged between different agencies.

**How it works:**

When registering a new Product or updating an existing Product record, it is required to select at least one provenance from the Provenance section. Each Provenance has Product Name, Code, Company, Company Code, Document IDs, and Indication sections. Also, each Manufacture Items has Manufacturer Item Code, Lots, and Ingredient Names. In the new Manufacturer Item Code section, it can store details about Manufacturer such as Manufacturer Name, Manufacturer Role, and Manufacturer Code and Code Type. In the Product update and detail pages, added a new Export JSON button which allows to export the data into a Json format.

Below is the snapshot of the register Product page from release 3.0.3 and 3.1 to show how much the Product model has changed from GSRS release 3.0.3 to GSRS release 3.1.

The Product details page is also updated with the new model.

Below is the snapshot of the register Product page from release 3.0.3.

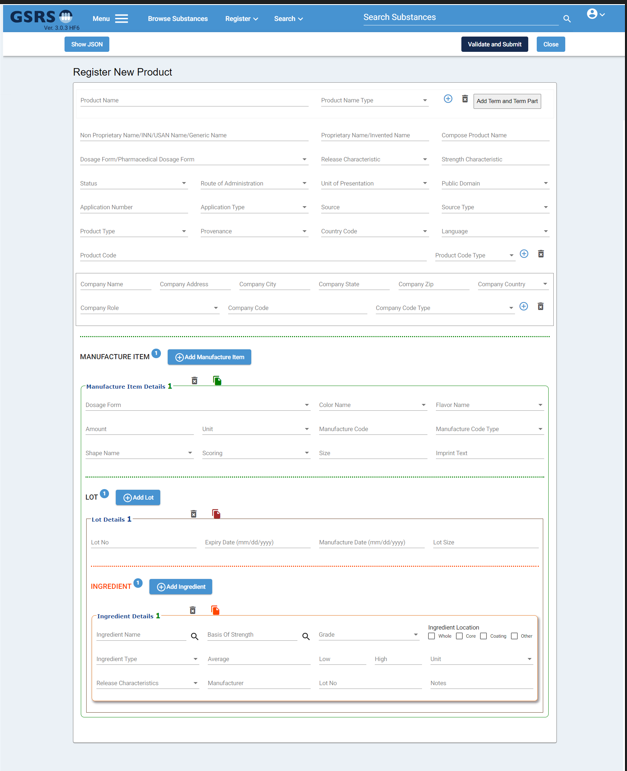


Figure 10-1 – Register Product page from GSRS release 3.0.3

Below is the snapshot of the register Product page from release 3.1.

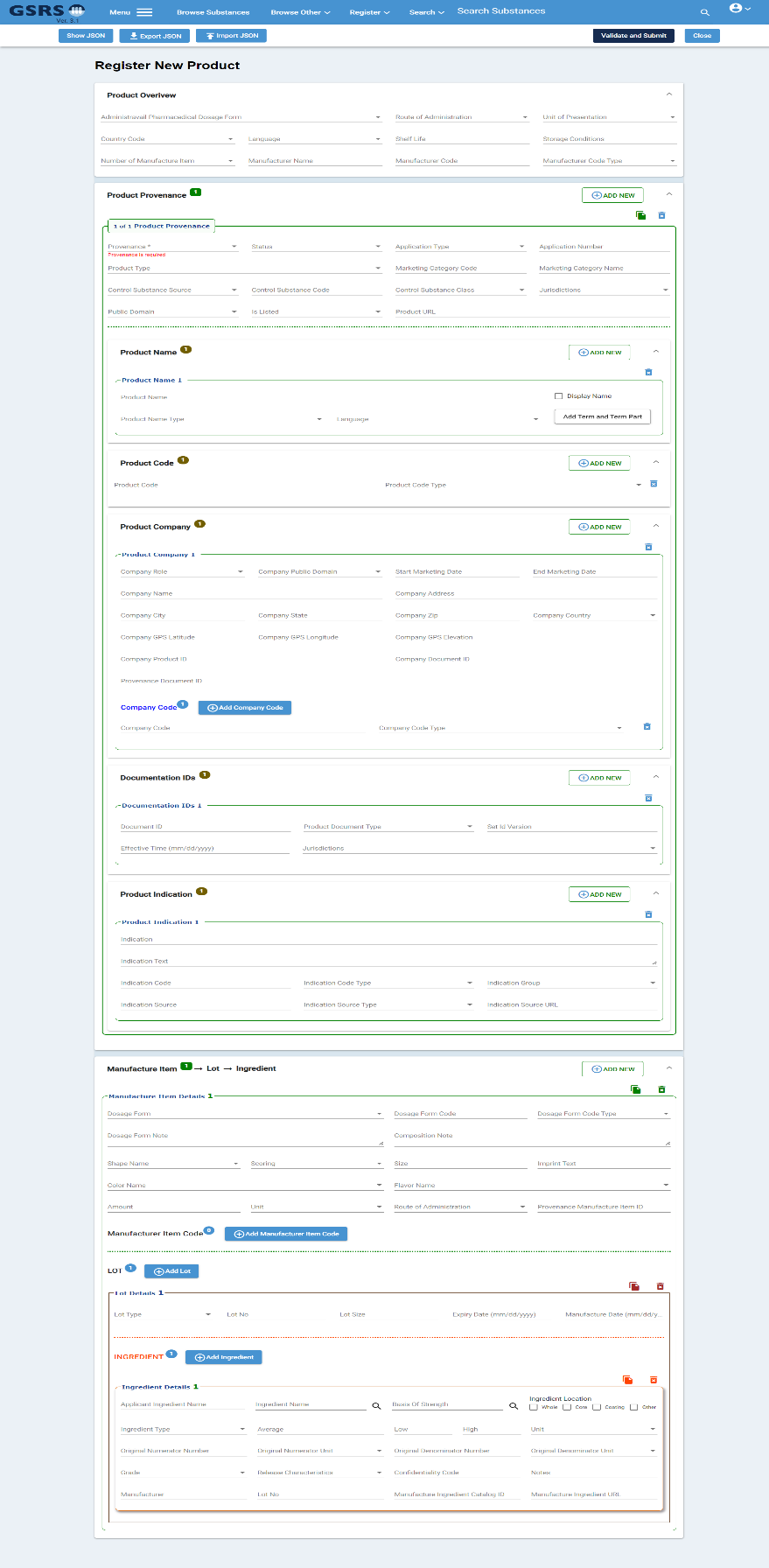


Figure 10-2 – Register Product page from GSRS release 3.1

In release 3.1, each section is collapsible, which will allow to hide and show each section when user clicks on the section header. Now, since there are more fields in the register and update pages, having a collapsible feature will make it easier to navigate the data based on the need.

**Notes for the Future:**

In the future release, product module will have only one REST API endpoint, as in the current release 3.1, there are three endpoints. In the future release, will rewrite the Browse Product page, exports, and display of product data in Substance pages.

## Major Improvement: Impurities Module Improvement

**Purpose and Motivation:**

In GSRS 3.1 release, the Impurities module has some improvements in the Impurities Register, Update, and Detail pages. These improvements will allow user to capture more valuable and important data in the form and save into the database.

**How it works:**

In this 3.1 release, there are many new fields added in the ‘Test’ section, such as Source Type, Source ID, Column Packing Type, Injection Volume Amount, Diluent, Suitability Requirements Resolution, and many more fields. Also, under the ‘Test’ section added a new dropdown ‘Elution Type’ field which can have controlled vocabulary value ‘Gradient’ or any other value. In this release also added new ‘Elution Solvent’ section, which contains fields ‘Elution Solvent’ and ‘Elution Solvent Id’.

Additionally, two new sections ‘Residual Solvents Test’ and ‘Inorganic Impurities Test’ are added to capture more details about the tests. The existing ‘Residual Solvents’ section is now a child of a new ‘Residual Solvents Test’ section, and ‘Inorganic Impurities’ section is now a child of a new ‘Inorganic Impurities Test’. Please see Figure 1 below.

Also, there are two new buttons added such as Import JSON and Export JSON. The Import JSON button allows to import existing impurities data, which is in a Json format, to import into the register Impurities form. The data can be modified on the form, and then can be saved into the database as a new record. The Export JSON button allows to export data into the file in a Json format.

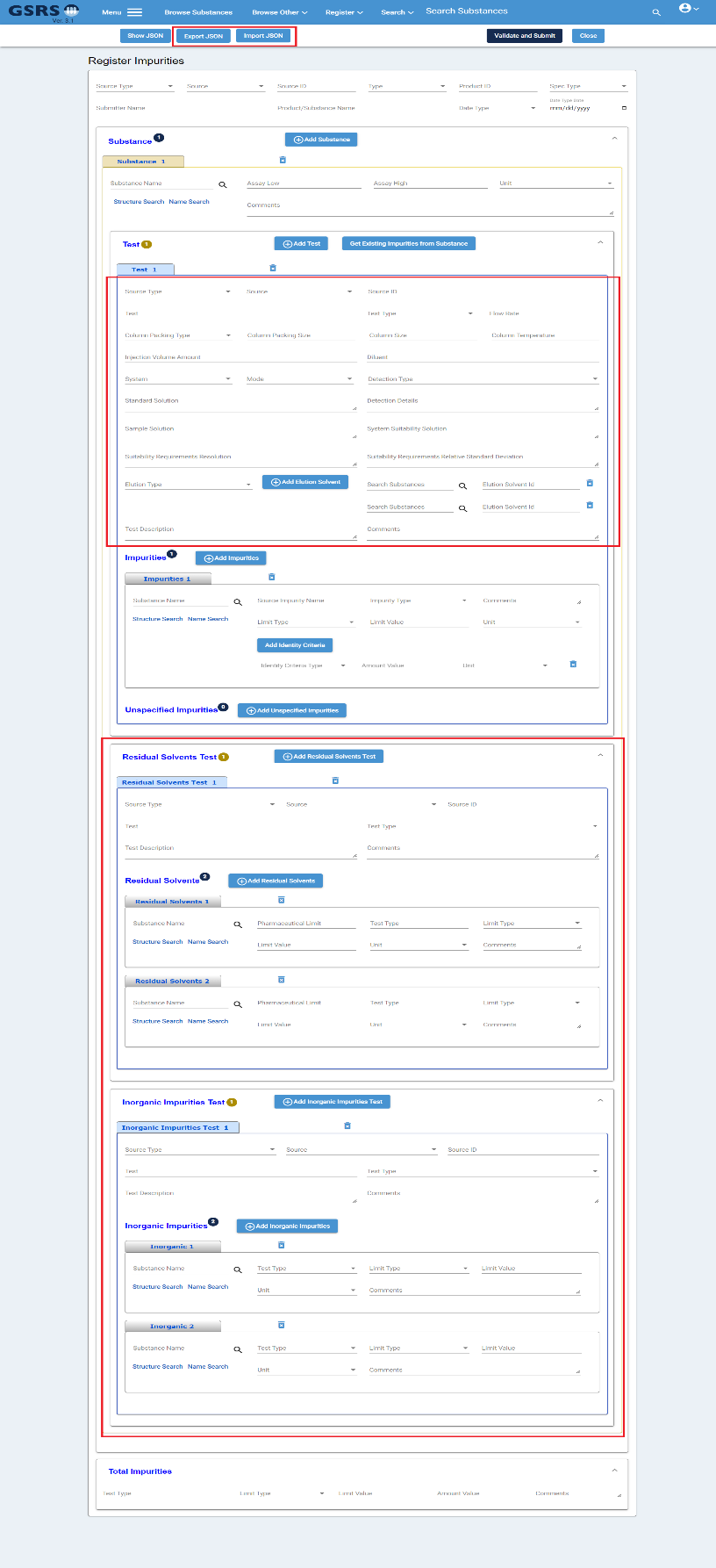


Figure 11-1 – Register Impurities page from GSRS release 3.1

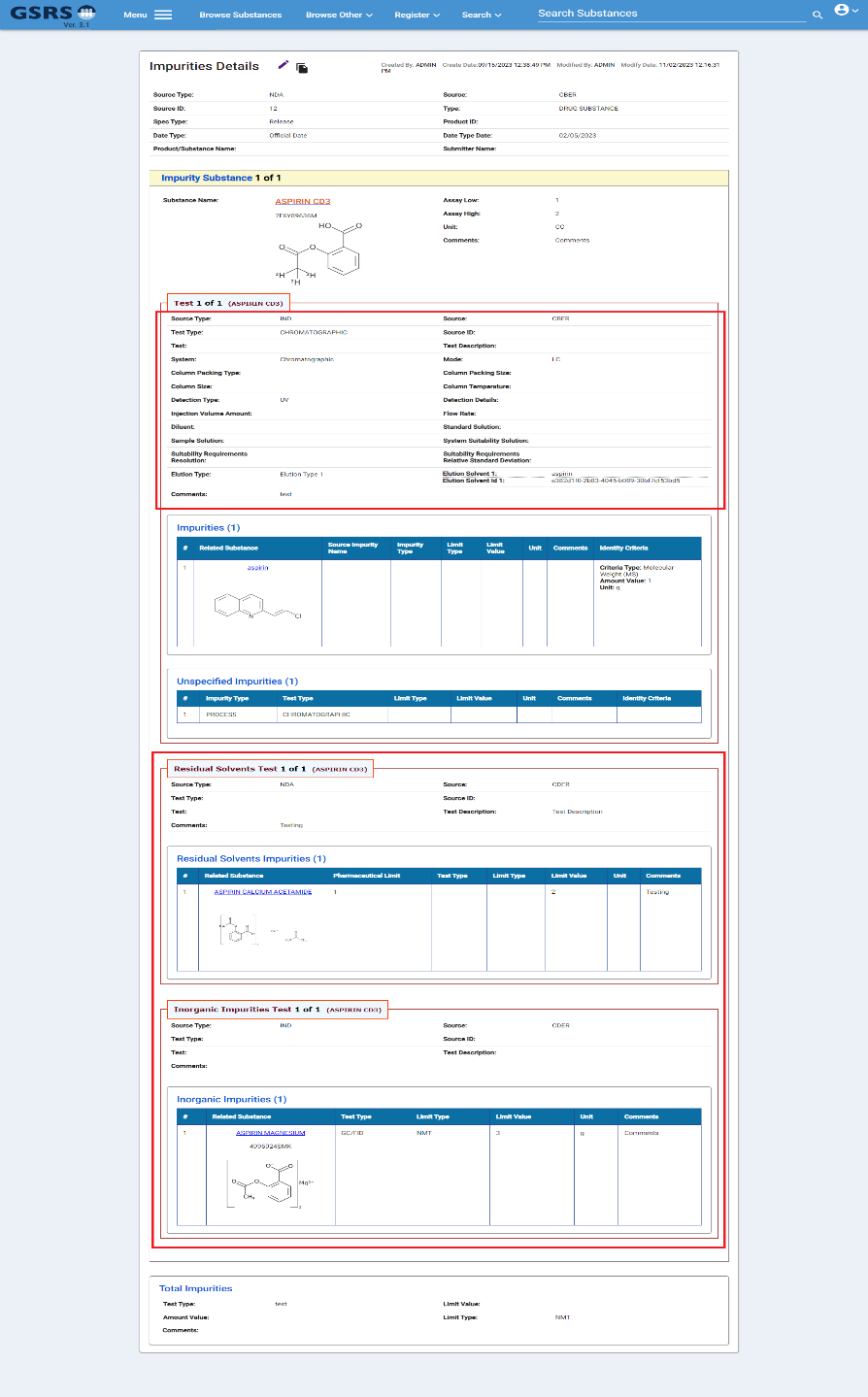


Figure 11-2 – Impurities Details page from GSRS release 3.1

**Notes for the Future:**

In the future release will continue to improve the impurities module and continue to add new features which can be useful and valuable to the users.

## New Feature: Specified Substance Group 4 Manufacturing (SSG4M) (**Preview Release**)

**Purpose and Motivation:**

In GSRS 3.1 release, implemented a new module Specified Substance Group 4 Manufacturing module. This module is released as a **preview** and more features and improvement will be implemented in the future release.

**How it works:**

The Specified Substance Group 4 Manufacturing (SSG4M) form has three tabs: Form View, Step View, and Scheme View. The form view tab contains the data entry form to enter the synthetic pathway details. See Figure 1 below. The Step View tab shows the data in a compact format with more visual and easier to understand pathway structures. See Figure 2 below. The Scheme View tab shows the visualization of the Synthetic Pathway. See Figure 3 below. There are lots of features in the SSG4M form such as all the fields are configurable to either show and hide fields from the form, Import and Export JSON, and auto populate starting and resulting materials. This SSG4M module is release and as a review right now.

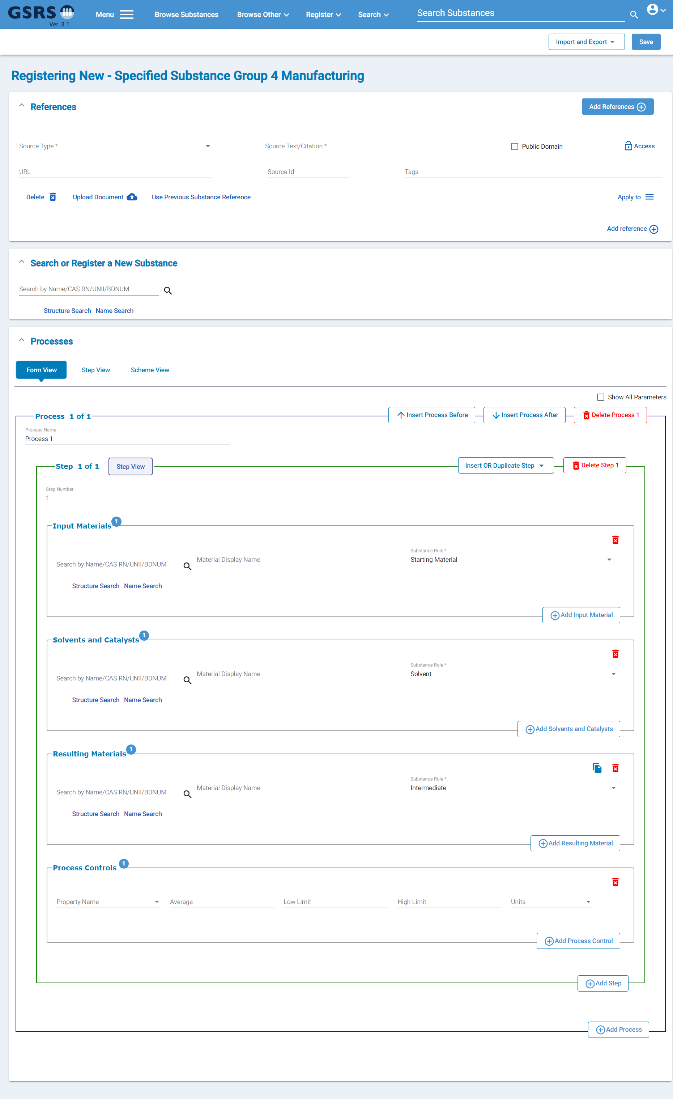


Figure 12-1 – Specified Substance Group 4 Manufacturing, Register Page (Form View)

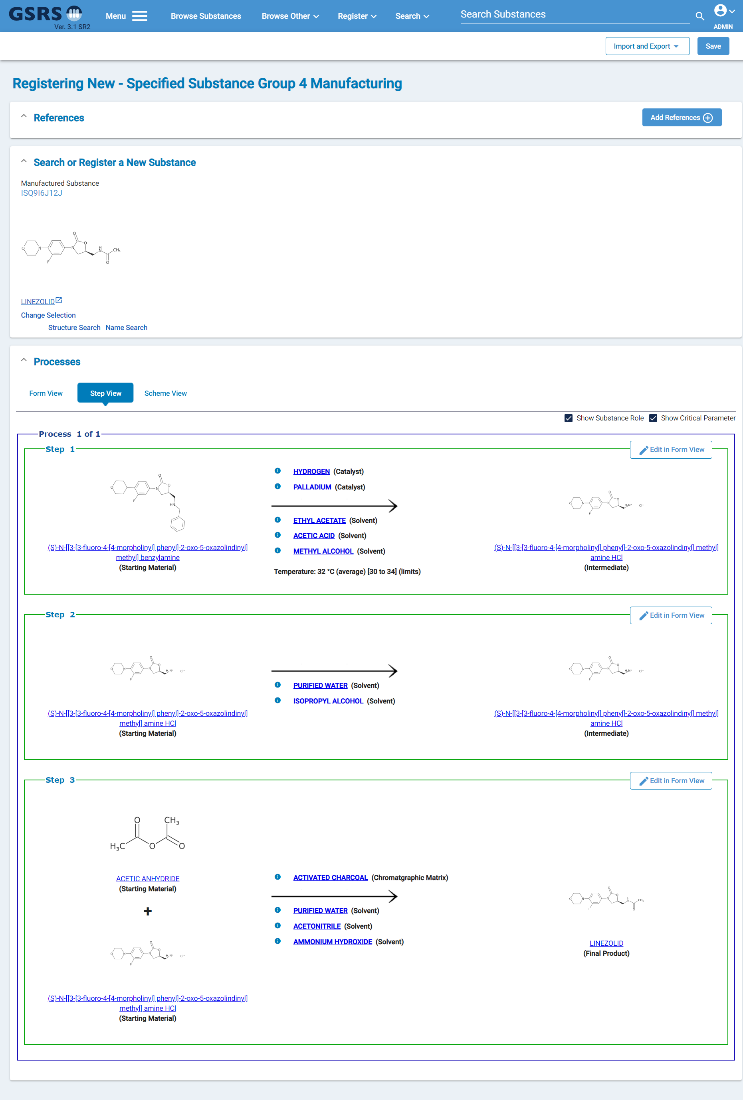


Figure 12-2 – Specified Substance Group 4 Manufacturing, Register Page (Step View)

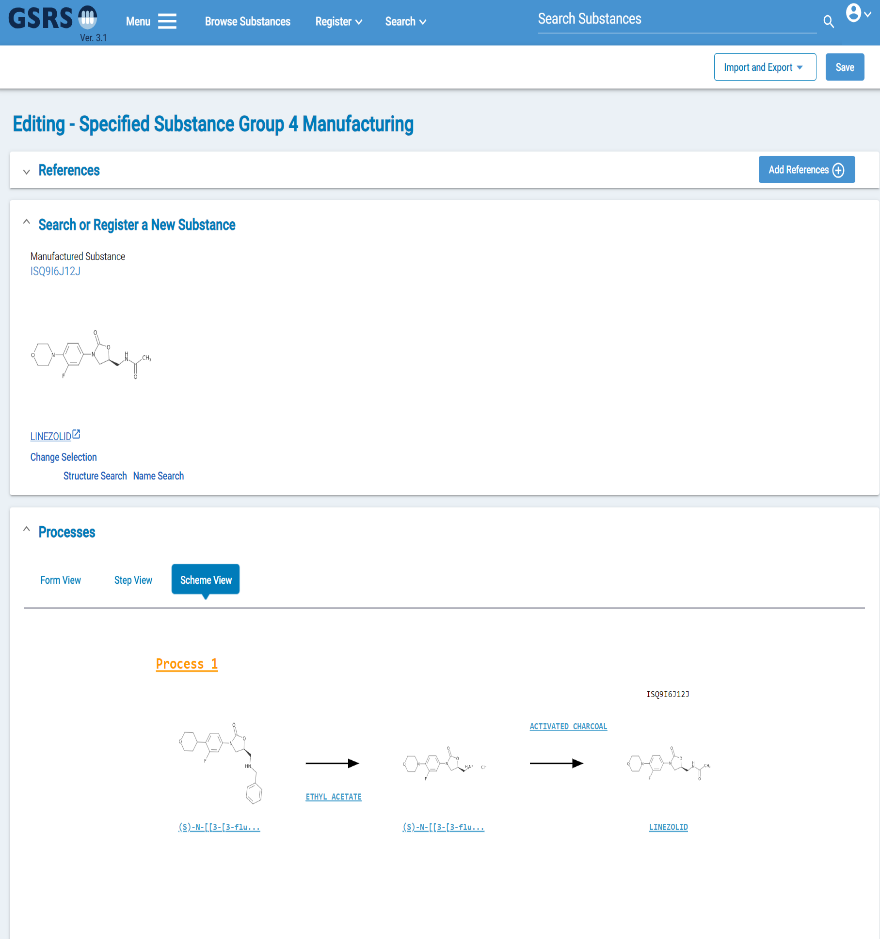


Figure 12-3 – Specified Substance Group 4 Manufacturing, Register Page (Scheme View)

**Notes for the Future:**

In the future release the SSG4M will have more features which will be ready to use.

# 

# Other selected Improvements and bug fixes:

* Structure search results are sorted by relevance by default.
* Add database connection info including active connections in health endpoint.
* A StandardNameDuplciateValidator class was added to prevent duplicates names (See the gsrs3-main-deployment/substances application.conf file for example configuration)
* The Frontend menu is more configurable than before. Fewer menu items are hardcoded. See “navItems” in the Frontend config.json file
* 3.1 New Frontend configurable items relevant to SSO. Organizations that use SSO security might benefit from configuring “logoutRedirectUrl” and/or “showChangeUserPasswordButton”. The first allows you to redirect to a URL after logout. The second regulates whether a button appears for users to change their password. The button may not be a desirable feature when SSO is used to handle logins.
* 3.1 Frontend privacy statement. We added a way to include a “privacy statement" **either** a) by overwriting “src/main/resources/static/assets/html/privacy-statement.html” in the frontend service or b) by adding a configuration entry key “privacyStatement” and a value with json encoded HTML text. If the configuration entry is present, it will take precedence.
* 3.1 New fields added to the Clinical Trials US entity service. When specifying substance to trial relationships, the substance record may now include one or more “Substance Roles”. The roles are populated from controlled vocabulary. In addition, data curators may now make “Outcome Result Notes” in each US trial edit form.