OHDSI Gold Standard Phenotype Library Technical Specifications Document

Contents

Objective

The objective of this document is to specify the technical requirements of the OHDSI Gold Standard Phenotype Library. We assert that the objective of this library is to enable members of the OHDSI community to find, evaluate, utilize, and contribute community-validated cohort definitions for research and other activities.

Summary

In a previous Requirements Document released to the community on August 20, 2019, we detailed the requirements of the OHDSI Gold Standard Phenotype Library. We put forward an architecture that would meet these requirements, but we refrained from going into the technical details, as implementation details are conceptually separate from the requirements details. Now, we will review the architecture we outlined and delve into the technology being proposed to make the library come to fruition.

Architecture Overview

An updated version of the architecture diagram previously proposed is as below:

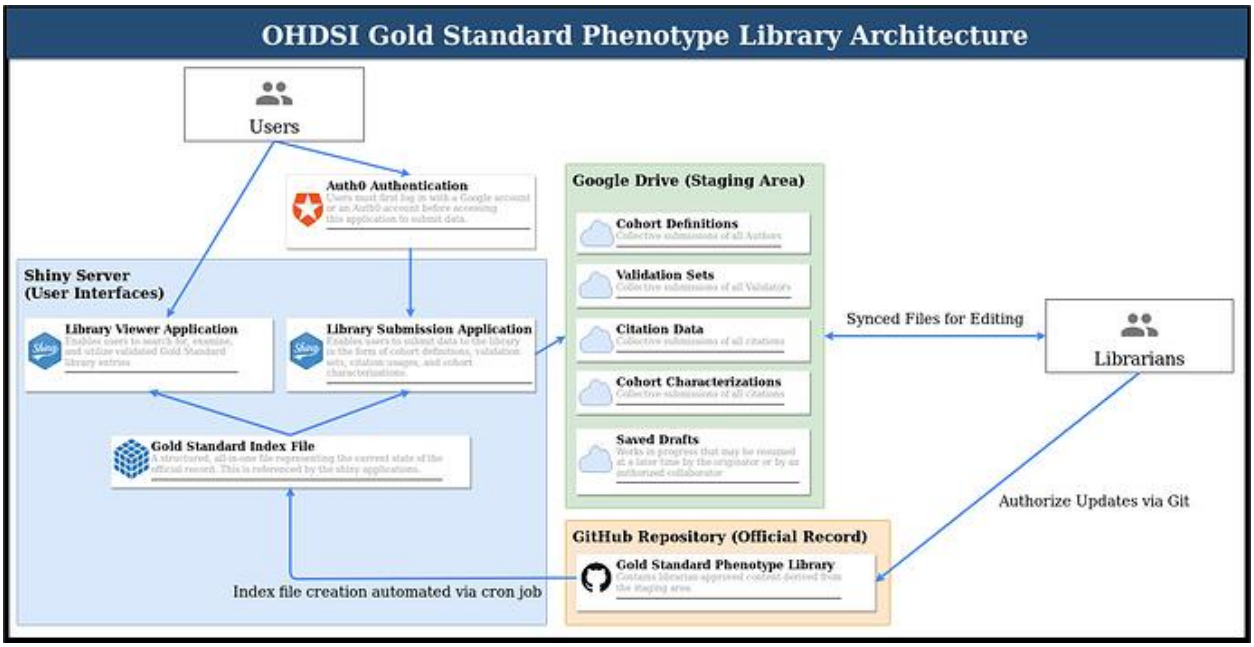

Figure 1. Library Architecture Diagram

Generally, the process remains the same as before, where users add data to the library, the data are curated by librarians, and librarians upload that data to the official record after the peer review process. Then, the data in the official record are referenced by the applications. This is a circular, knowledge-accumulating process.

This time, in each step of this process, we have input the technology we see as the most promising to accomplish the task at hand. We will now explain in detail what these technologies are and why we are proposing to use them.

Technology Overview

Our plan encompasses a mix of R Shiny applications, the Auth0 authentication service, Google Drive, and GitHub. We attempt to draw from the benefits of all of these services to avoid "reinventing the wheel" as much as possible, while allowing sufficient flexibility to develop the library in order to fulfill its requirements and address feedback from the community. The four proposed platforms are all rigorously developed and are either entirely free to use, or free to use up to limits that far exceed the expected library traffic.

R Shiny Applications

There are two applications associated with the library. One of these is a viewer application, which is associated with read-only activities (e.g. viewing, searching, and downloading phenotypes). The other is a submission application, which is associated with submitting data to the library. Currently, the modes of data submissions are cohort definitions, validation sets, citation data, and cohort characterizations. The proposed framework allows other modes which have not yet been anticipated to be easily added in later.

A guiding principle behind using R Shiny is that OHDSI is already an R-based community. At the time of writing, there exist 24 R Shiny applications on OHDSI's Shiny Server at data.ohdsi.org. Most of the tools OHDSI has developed have been in the R framework, and we are simply attempting to tap into this; we envision that both applications could live on the existing Shiny Server, which would make integration easy and familiar. It also enables contributions from members of the community who already know how write Shiny application code. Fortunately, this doesn't come with any obvious tradeoff; we believe that Shiny provides an excellent interface to fulfill all of the requirements moving forward.

The Viewer Application

As mentioned, the viewer application is a place for read-only activities. Although the Official Record holds all of the public-facing data for the library, we anticipate that the viewer application can make it easier for individuals to search for, understand, and download the cohort definitions they require.

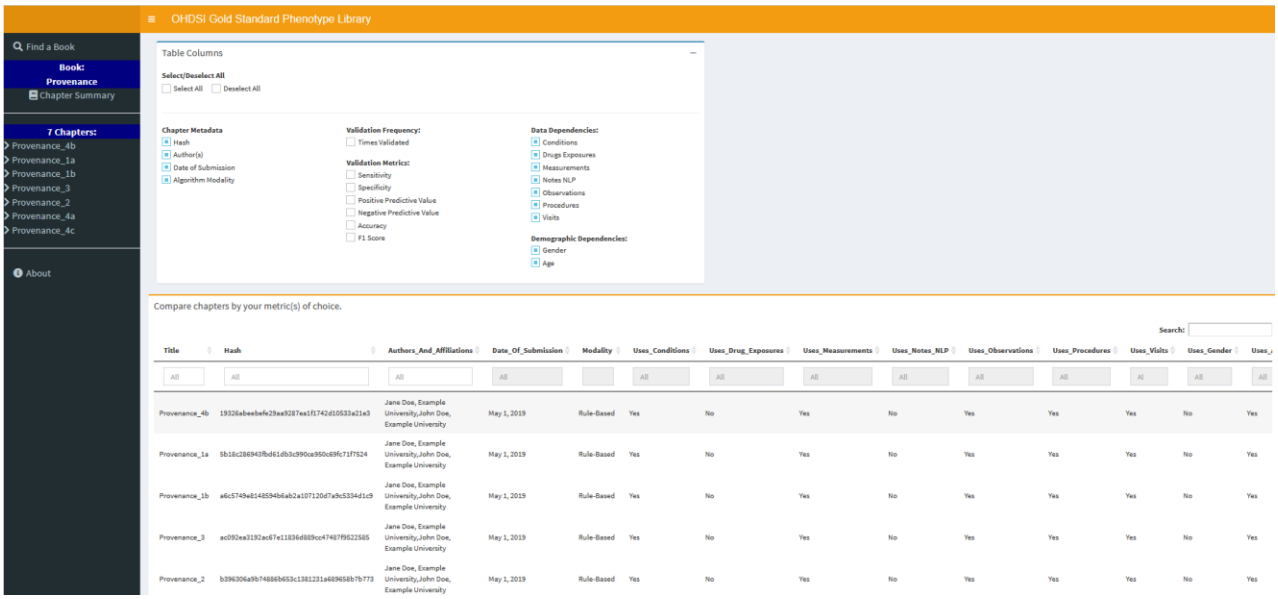

Figure 2. A screenshot of a page of the viewer application

There are several menus associated with the application:

- Main Menu: A place to begin by searching and selecting a book (phenotype) from the library

- Chapter Summary: A tool to search, filter, sort, and otherwise explore the metadata of all chapters (cohort definitions) within the currently selected book

- Chapter Menu: Allows for a deep dive within a selected chapter via the following tabs:

    - Summary: Rendered markdown file of all Common Data Elements comprising the chapter.

    - Validation: Presents and overview of the validation done on the chapter, as well as the ability to deep dive into any single validation.

    - Provenance: Shows a network graph of the chapter and how it relates to other chapters in the library

- About: A menu that details information about the Gold Standard Phenotype Library

A screenshot of Chapter Summary menu is given in Figure 2.

The Submission Application

For submitting data to the library, we are operating in a different environment than the read-only one for the Shiny viewer application. We therefore opt for a separate Shiny application, primarily because we require authentication to utilize this application, as discussed further in the Auth0 section below.

The submission application has a set of menus of its own:

- Main Menu: A place to begin by loading an existing draft or by selecting one of the following modes of submission, where each mode has its own form of data needing to be filled out:

    - Cohort Definition: Propose a new book and/or chapter to be added to the library.

    - Validation Set: Validate a Gold Standard Phenotype and supply the metrics obtained to the library.

    - Citation Usage: Submit a citation to highlight the published use of a Gold Standard Phenotype

    - Cohort Characterization: Submit descriptive statistics (a "Table 1" of data) on a cohort produced by applying a Gold Standard Phenotype to a dataset

    Each form represents the data elements outlined in the Requirements Document and required to be entered in order for the phenotype to be considered a "Gold Standard" one. At the time of writing, the Citation Usage and Cohort Characterization elements are relatively new and hence are works in progress.

- About: A menu that details information about the Gold Standard Phenotype Library

Figure 3 shows the main menu of the application, and Figure 4 shows the top of the form for submitting a new cohort definition.

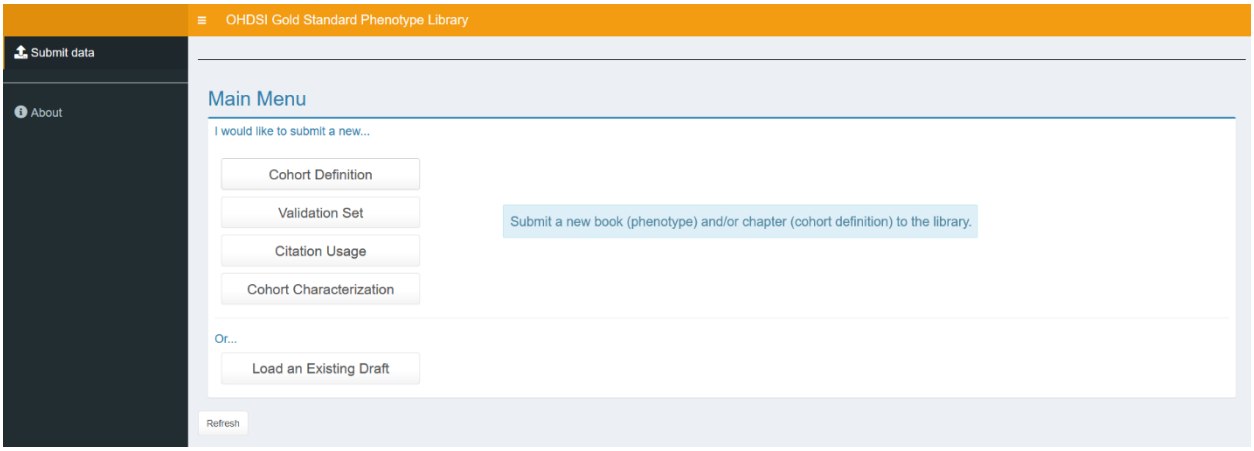

Figure 3. A screenshot of the submission application main menu

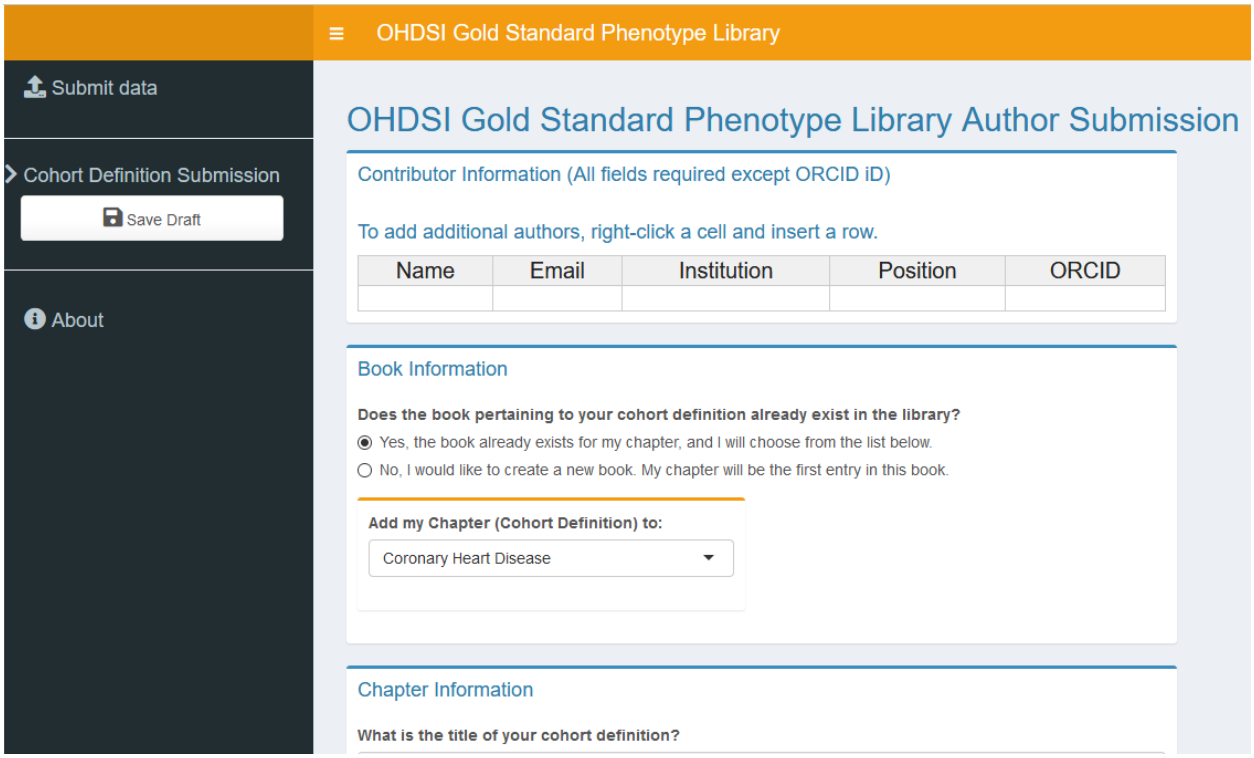

Figure 4. A screenshot of the cohort definition submission form

Index File

Technically, on loading, the Shiny applications could run queries against the Official Record to obtain up-to-date information regarding the library's contents. However, it is prohibitively slow and unnecessary to expect a user to wait for this process to complete every time the application loads.

Instead, we have the Shiny applications load an index file of pre-compiled data on a session instantiation, which is much faster and more efficient. The index is a single RData file containing all of the information contained in the Official Record, as well as derived data elements. Some examples of derived data are the metrics (e.g. sensitivity, specificity, etc.) computed from the raw True/False Positive/Negative cells given by the validation sets, as well as connected components to display a network of all related cohort definitions. The index file step also presents an opportunity to insert additional advanced calculations that have not yet been fully anticipated. For example, we have considered the idea of comparing semantically similar definitions to reduce cohort definition redundancy.

The index file only needs to be updated whenever there is a change to the Official Record. Currently, we have a job scheduler (cron) in place that periodically checks the timestamp of the index file against the timestamp of the latest commit to the official record GitHub repository. If, during this check, the index file appears outdated compared to the official record, the R script that builds the index is triggered. The cron job works off of regularly scheduled time intervals, but we welcome suggestions to get the script to trigger only when the repository has been updated, perhaps under a continuous integration framework.

Auth0 Authentication

Per the requirements of the library, authentication is required in order to submit data to the library (but not to view data). Hence, prior to accessing the submission application, we require users to log in. In order to accomplish this, we are proposing using the Auth0 authentication service. Under this service, when a user tries to access the submission application, they first encounter the login screen (Figure 5) where they can authenticate with a Google or Auth0 account. After a successful login, they are redirected to the submission application to continue.

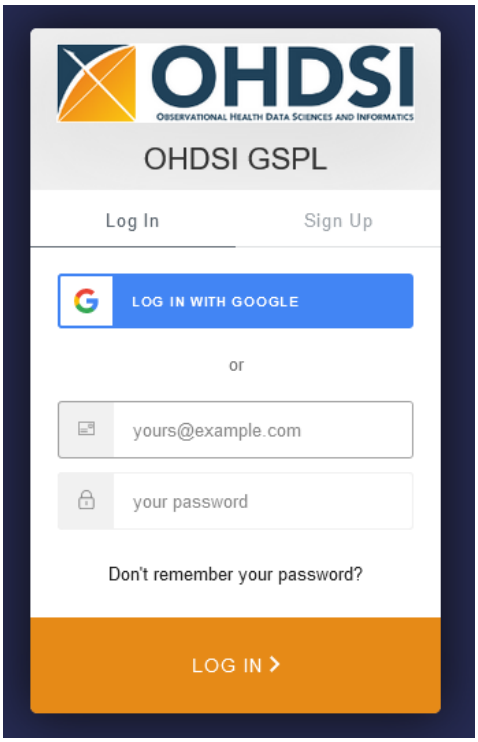

Figure 5. The OHDSI-Customized Auth0 Login Screen.

Auth0 is a service that provides an authentication platform for web services. It is a "freemium" service that offers 7,000 logins per month. However, Auth0 has an Open Source Program that enables unlimited logins *if* the application is open source (which the library is). The login limit is certainly high enough for the foreseeable future, but if the library grew such that it became insufficient, we could petition to have the limit removed.

The advantages that Auth0 offers over other authentication options are twofold:

1) A dashboard of easily configurable options, account management, and login history is readily available with the service (Figure 6).

2) The `auth0` R package makes connecting an R Shiny Application to the Auth0 service trivial, by simply replacing the call `shinyApp(ui, server)` with `shinyAppAuth0(ui, server)`. Auth0 ascribes a unique user id to the individual logged in, which can be referenced in the application code. An Auth0 config file is also required but is easy to set up; ours has just 7 lines in our testing environment.

In general, authentication offers potential to enhance the user experience by making the submission process more efficient. For example, if we've seen a submission by an individual before, we might pre-populate the author table for a subsequent submission. We can also enable features not otherwise possible without authentication, such as saving and loading drafts, since we can tie the draft data to the user doing the saving/loading.

Authentication also offers a degree of security. Since data are to be submitted to the library, requiring individuals to identify themselves before submitting data mitigates the ability to submit malicious anonymous content.

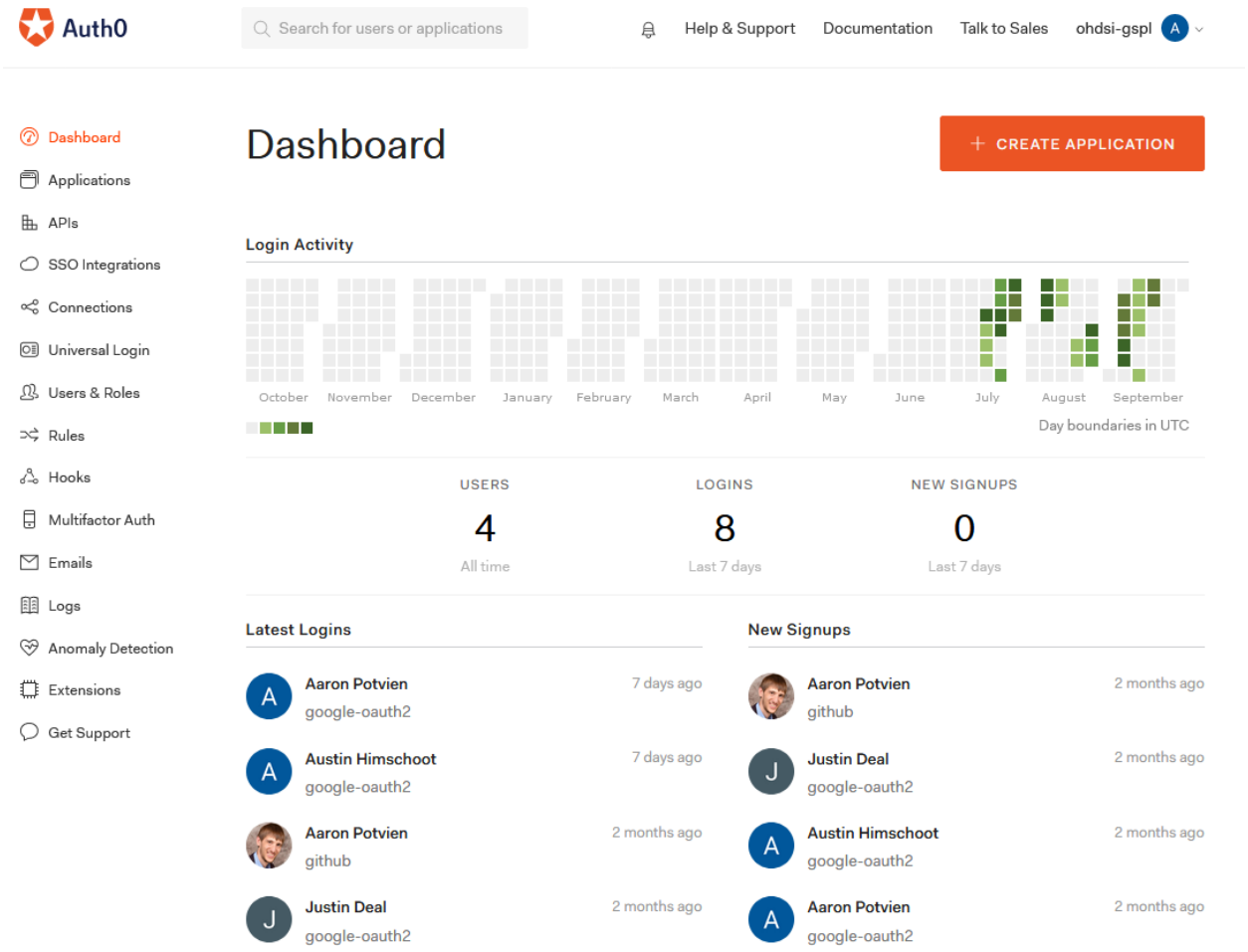

Figure 6. The Auth0 Dashboard for the library submission application

HTTP to HTTPS Redirect

In order for Auth0 to operate, the web application must use the secure HTTPS protocol – Not HTTP. Internally, we have accomplished this in our test environment via a standard nginx HTTP-to-HTTPS redirect (See here for more details) using self-signed Secure Sockets Layer (SSL) certificates. Currently, the OHDSI Shiny Server on data.ohdsi.org operates under HTTP and not HTTPS, so this is an enhancement that would need to be made to the server, ideally using certificates that are not self-

signed. Even outside the scope of the library, this would be a prudent step to take with regard to OHDSI's Shiny Server for security purposes. However, in order to deploy the library as proposed, this would be a necessary step. It is worth noting that doing so would not interfere with any hosted applications.

Google Drive

For the staging area, we are proposing to use Google Drive. In principle, any shared cloud storage mechanism could be used, but Google Drive offers several advantages:

1) Integration with the Shiny applications is straightforward with the `googledrive` and `googlesheets` R packages; these allow for uploading and downloading from a Drive account, among other interactions with Google Drive.

2) Availability of "Google Sync" to allow librarians to connect to the Staging Area and have changes reflected across the board.

3) Per the Google Drive security policy, all uploaded files are scanned for viruses prior to being downloaded or shared.

A disadvantage is that Google offers only limited storage capacity for free. Per their policy, 15GB of storage is free. After that, $1.99 a month is required to store 100GB, and $9.99 a month is required to store 1TB of data. Fortunately, the JSON files are small to characterize cohort definitions. However, bounds have not yet been placed on the supplementary documentation and other attachments allowed to occur with a submission. If all submissions took up 10MB of space, we could store up to 1,500 cohort definitions before an upgrade or transition away from Google Drive becomes necessary.

Google Sync

Although librarians could, in principle, make any needed changes directly on Google Drive online, we are recommending that every librarian download Drive to have the staging area synced locally on their machine. This will make transitioning data from the staging area to the official record easier, as well as making edits prior to this step, as necessary.

Service Account

Of note, in order for the Shiny application to have permission to write to a Google account, a service token must be generated and placed on the Shiny server. This can be done through the Google Developers API, and an example of this setup is available on our internal test server. This is also presumably needed to authorize confirmation e-mails to be sent from the service account after a submission, which is a requirement of the library.

GitHub

As part of the library architecture, we envision having an Official Record be a standalone open source repository, existing outside of the R Shiny applications. This repository is to hold the OHDSI-sanctioned and publically available library of approved phenotypes and all associated metadata. Content from the Staging Area gets promoted by the librarians to become part of the Official Record.

We envision this process as a comparison between what exists in the staging area to what exists in the official record. Git is a version control software that can automatically report on these differences and allow for librarians to commit changes to the library. We choose GitHub because OHDSI already has over 150 GitHub repositories, so it is natural to tap into this existing environment. In fact, before this work began, a repository already existed specifically for phenotyping, so we are proposing to place the Official Record at that location.

When a librarian commits changes, he/she publically takes ownership of the change. Using the Git/GitHub ecosystem, the details of the change are automatically taken care of, including the history and nature of the changes taking effect, as well as the calculation of the differences between the Staging Area and the Official Record.

Conclusion

In the Requirements Document released on August 20, 2019, we described all of the desired features of the OHDSI Gold Standard Phenotype Library and set forth an architecture we believed would allow for the library to aptly function. In this document, we focused on the technology required to drive the components of the architecture. We discussed the use of R Shiny applications to view and submit data to the library, the Auth0 authentication service, Google Drive to facilitate the Staging Area, and GitHub to support the Official Record. As has been the case throughout, the Gold Standard Phenotype Library Working Group remains open to OHDSI Community suggestions.

Relevant Links

Gold Standard Phenotype Library Working Group Forum

In August of 2018, the requirements development began for the library. In January of 2019, the Gold Standard Phenotype Library Working Group was founded and began meeting regularly. A record of this development is available here:

http://forums.ohdsi.org/t/requirements-development-for-the-ohdsi-gold-standard-phenotype-library/4876

Gold Standard Phenotype Library Working Group Wiki

In addition to a forum thread, this working group has a wiki, which contains links to some presentations given about the library:

http://www.ohdsi.org/web/wiki/doku.php?id=projects:workgroups:gold-library-wg

Requirements Document

Prior to the development of the Technical Specifications, we have developed a Requirements Document that explains the features of the library and data captured by the library.

https://forums.ohdsi.org/t/requirements-development-for-the-ohdsi-gold-standard-phenotype-library/4876/99?u=apotvien

Phenotype Definition Forum Post

On May 8, 2019, James Weaver posed the question about what a phenotype is in the context of observational research. This discussion guided our development of definitions and book/chapter structure:

http://forums.ohdsi.org/t/what-is-a-phenotype-in-the-context-of-observational-research/6796

Software Demonstration Forum Post

On September 5, 2019, Aaron Potvien released an updated architecture diagram in advance of the 2019 OHDSI Symposium Gold Standard Phenotype Library Software Demonstration.
https://forums.ohdsi.org/t/requirements-development-for-the-ohdsi-gold-standard-phenotype-library/4876/109

APHRODITE

The Gold Standard Phenotype Library houses computable phenotypes. Dr. Juan Banda has developed Automated PHenotype Routine for Observational Definition, Identification, Training and Evaluation (APHRODITE), a methodology and software for developing computable phenotypes:

https://github.com/OHDSI/Aphrodite/

PheValuator

Validating cohort definitions is an important component of the Gold Standard Phenotype Library. Dr. Joel Swerdel has developed methodology to validate a computable or rule-based phenotype algorithm by checking its performance against extremely specific or sensitive cohorts:

https://github.com/OHDSI/PheValuator/

Interface Prototype

An early prototype of the viewer application was built and is hosted via a Shiny Server at the following address:

http://data.ohdsi.org/PhenotypeLibraryViewer/