Hamilton County Improved Health Dashboard Data Portal

User Guide

Overview

The Data Portal serves a supporting role with the Hamilton County Improved Health Dashboard course project by acting as the store of data that our public facing dashboard pulls from. This communication occurs through a public API exposed by the Data Portal, which in turn interfaces with a SQL database. The API is read-only, and while the Data Portal is deployed on the public web, access to it is locked behind user authentication so that CRUD operations cannot be performed via the web interface unless the user is authorized. Using the Data Portal the user is able to influence the content display on the public facing dashboard by creating/modifying diseases and their categories, as well as by submitting and editing case, death, and hospitalized reports.

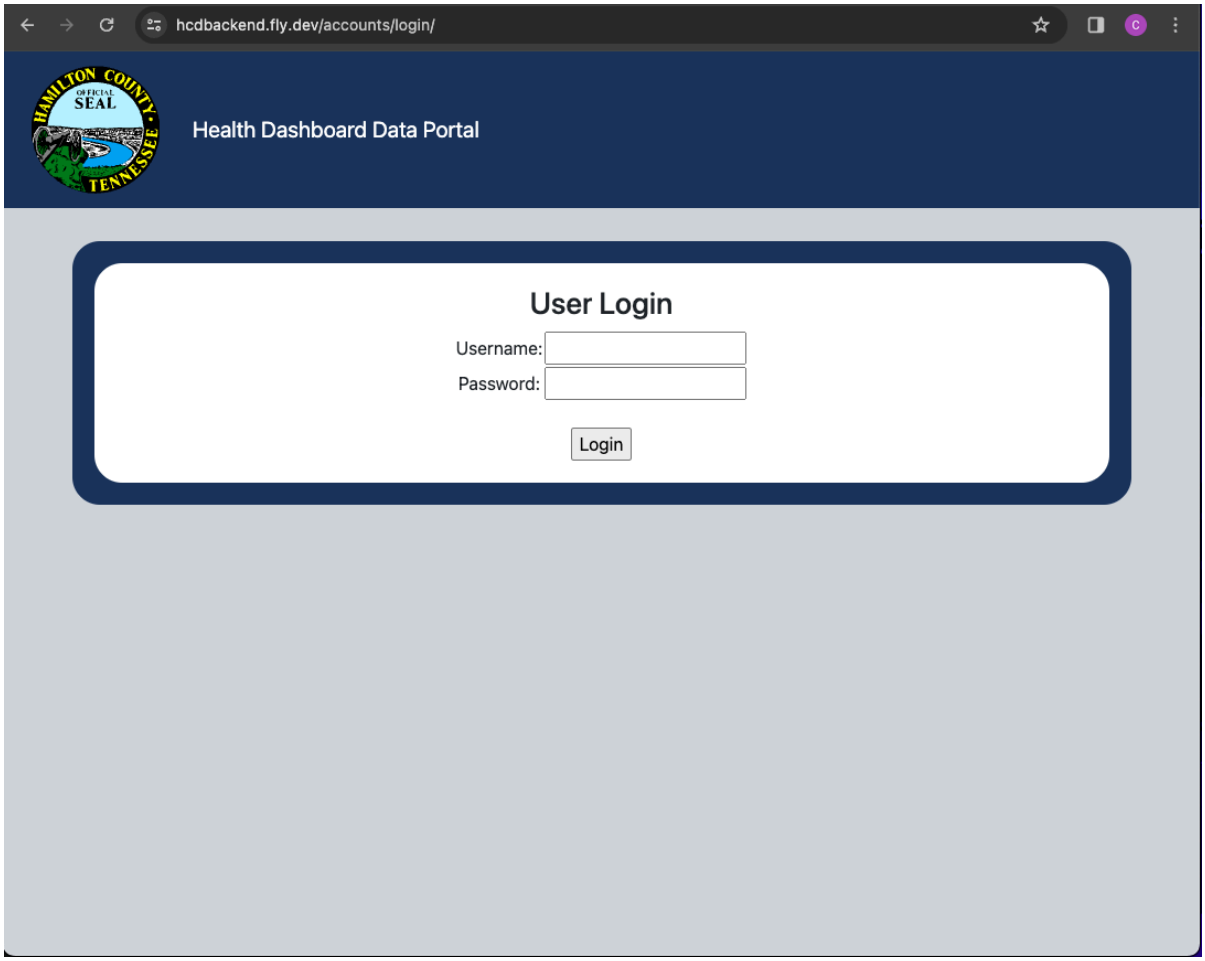
The Data Portal was written in Python using the Django web framework. It is hosted and deployed using the cloud-service provider Fly.io, which streamlines various containerization and networking steps required to deploy a website accessible on the public web. PostgreSQL is used as the database and it is hosted through AWS using their Relational Database Service, so both the DataPortal and accompanying database exist on separate web servers using different cloud providers. The data contained in our database is a mix of real and mock data. COVID-19 data was pulled using the same endpoints that the current Hamilton County Health Dashboard uses, while data for the other diseases was generated by our team.

A link to our Data Portal is found here:
https://hcdbackend.fly.dev/dataportal/

Supporting information on the technologies used by the Data Portal may be found below:

Django
https://www.djangoproject.com/
Fly.io
https://fly.io/
AWS RDS
https://aws.amazon.com/rds/
PostgreSQL
https://www.postgresql.org/

Authentication / Authorization

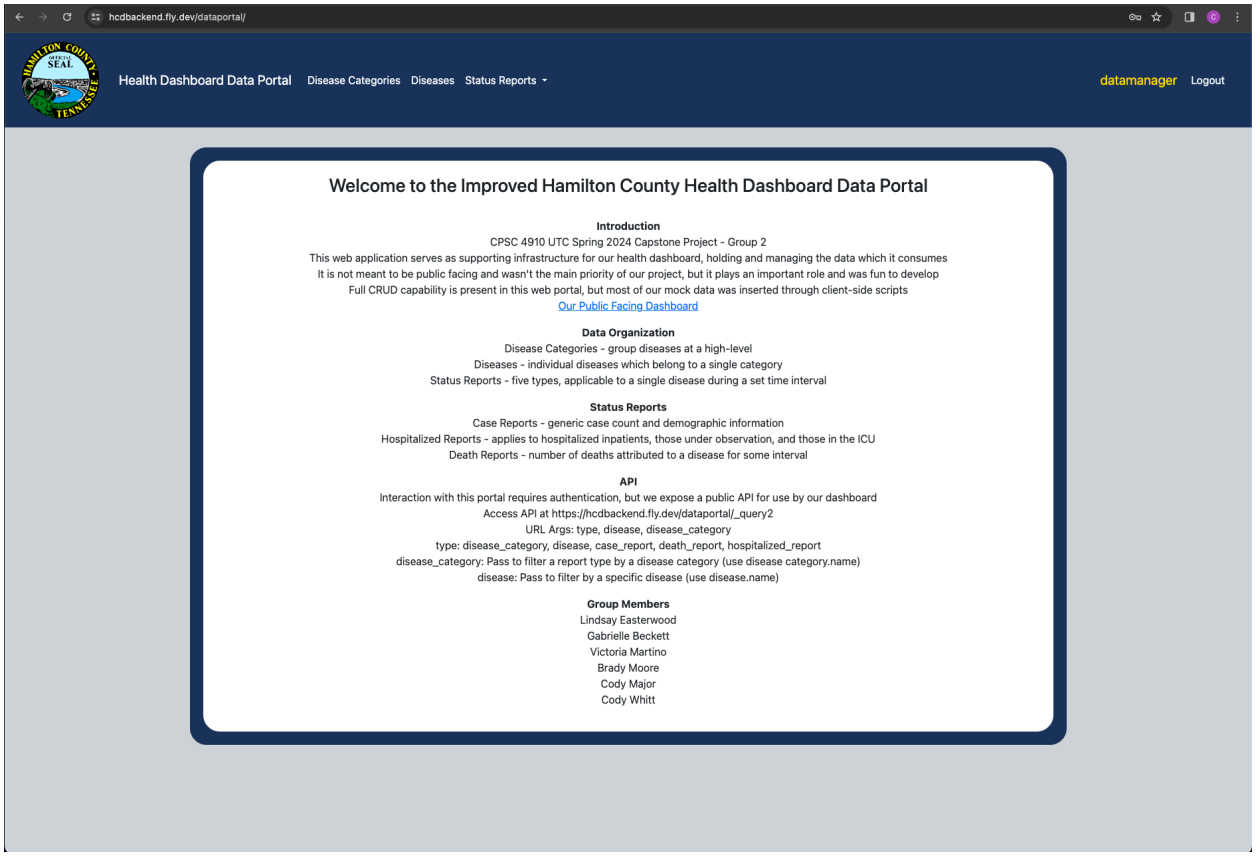


Accessing the Data Portal will require you to first login. This can be done using the following credentials.

Username: datamanager
Password: mocs4910

This security is required since the Data Portal allows create/update/delete operations to be performed via the web interface.

There are currently no other user accounts or ways to generate user accounts outside of Django's built-in admin portal. Since the Data Portal plays only a supporting role in the project, this functionality was not deemed necessary, but the groundwork for expansion in the area exists.

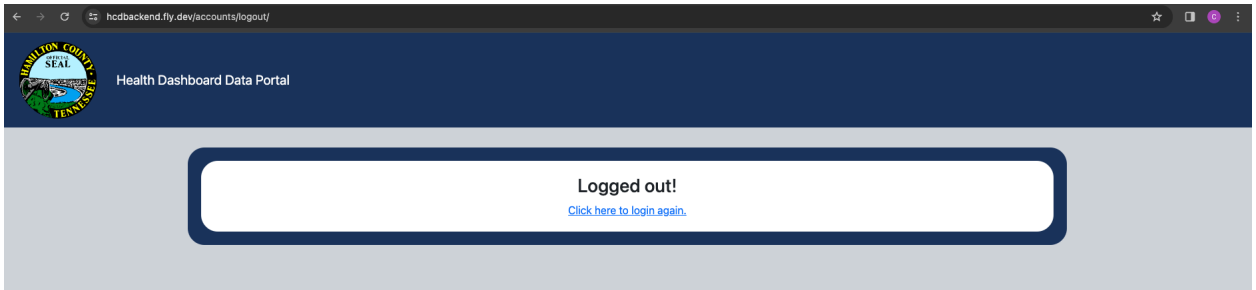


Once you are logged in, you are redirected to the Data Portal's home page. In the upper right of the navigation bar you will see your user account name along with a link to log out of the portal.

While authenticated, you are able to access all areas of the portal and perform CRUD operations to interact with the backing SQL database. Navigation to those resources occurs through clicking elements displayed in the navigation bar.

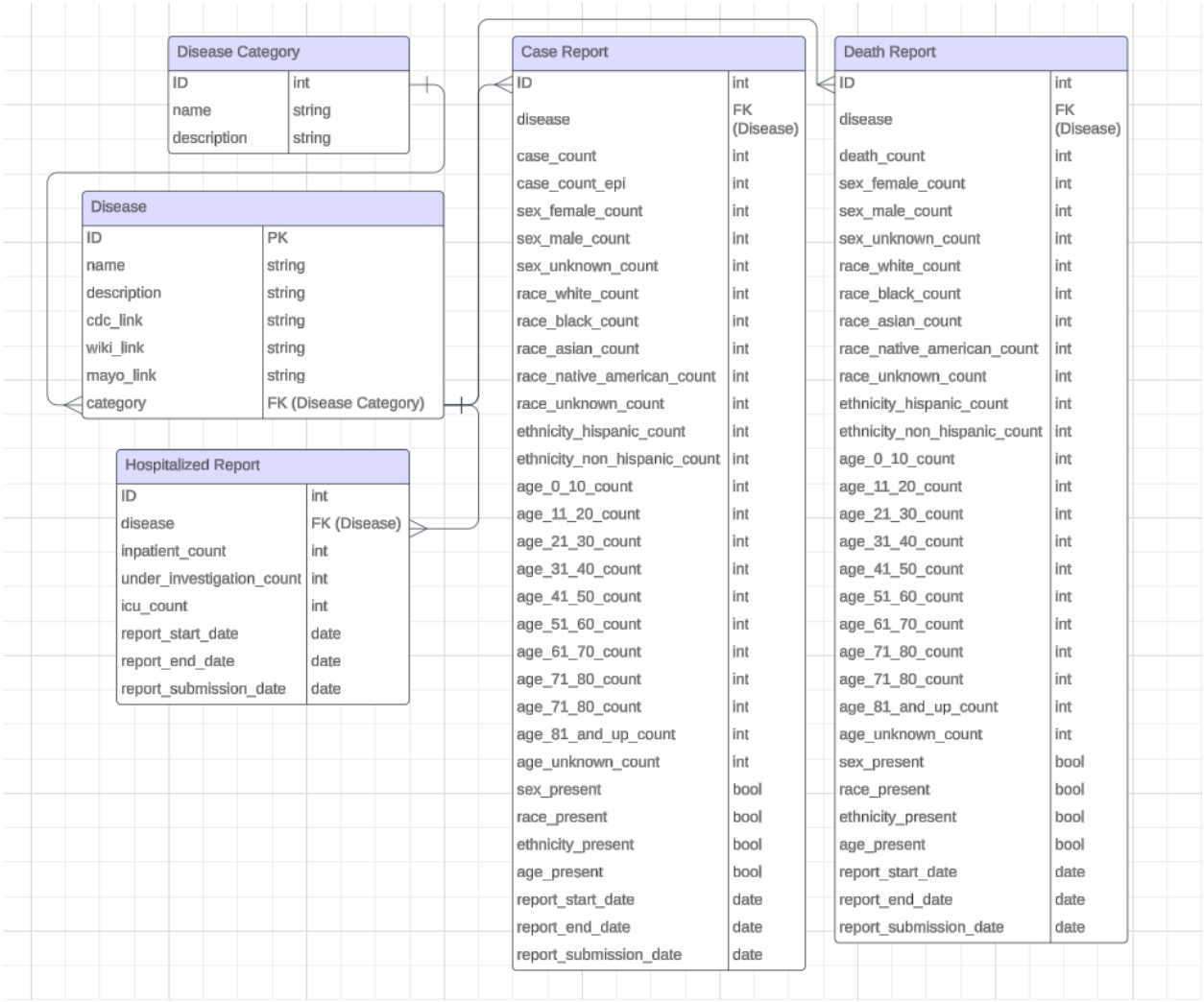
Authentication state will store in your browsers cookies if enabled.

Clicking "Logout" will redirect you to the following page:



Data Schema

ERD Diagram



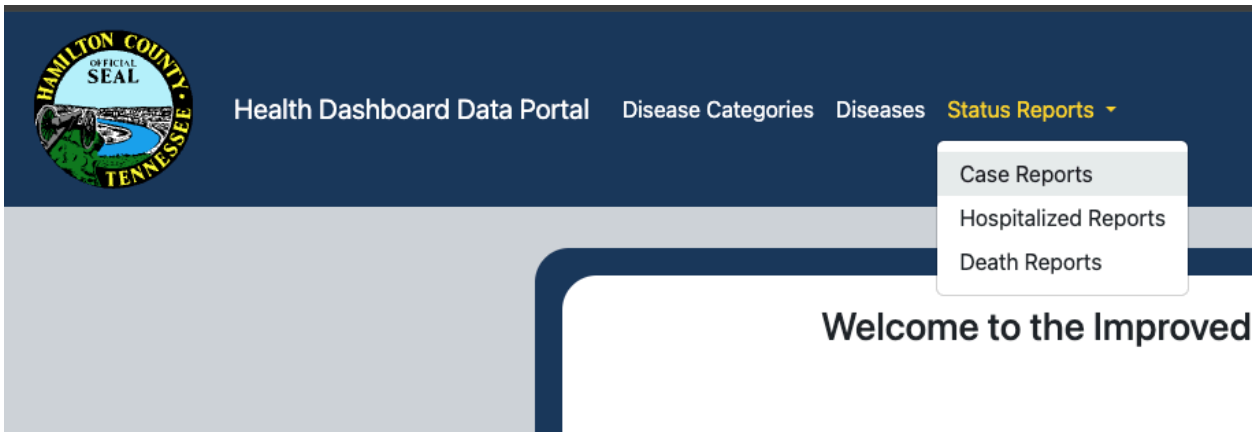
Data Organization Notes

Reflecting the nature of disease classification, the Data Portal enforces a hierarchical structure where related diseases are grouped into categories, and individual status reports are tied to specific diseases. This allows status reports to be easily queryable by either specific disease or the overall disease category. Start and End dates are flexible to allow reports to cover any period of time desired by the user.

Demographic data is optionally present on two of the report types, with <demographic_group>_present bit fields available to reflect whether that specific area of demographic data is populated or should be ignored.

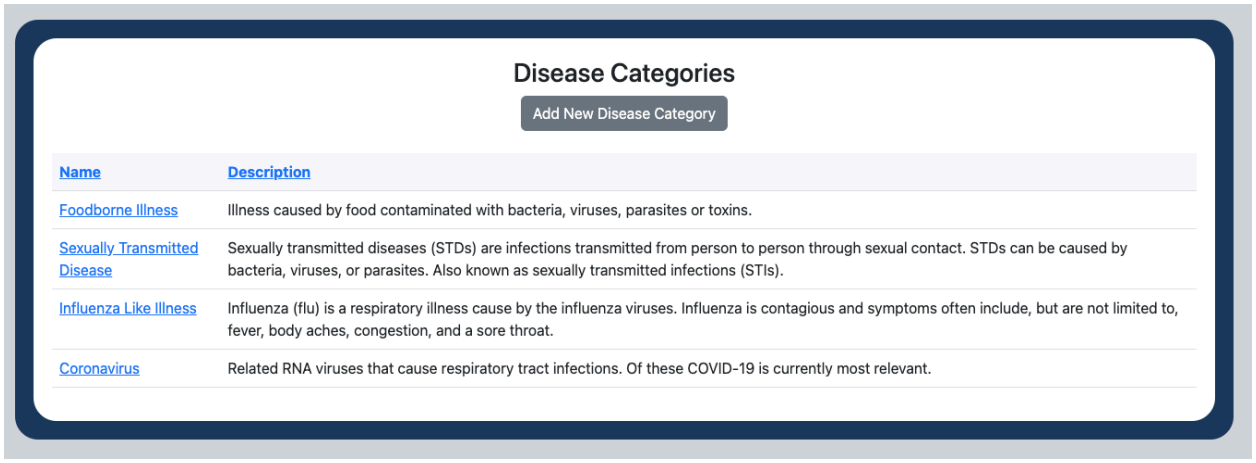
CRUD Views

Navigation



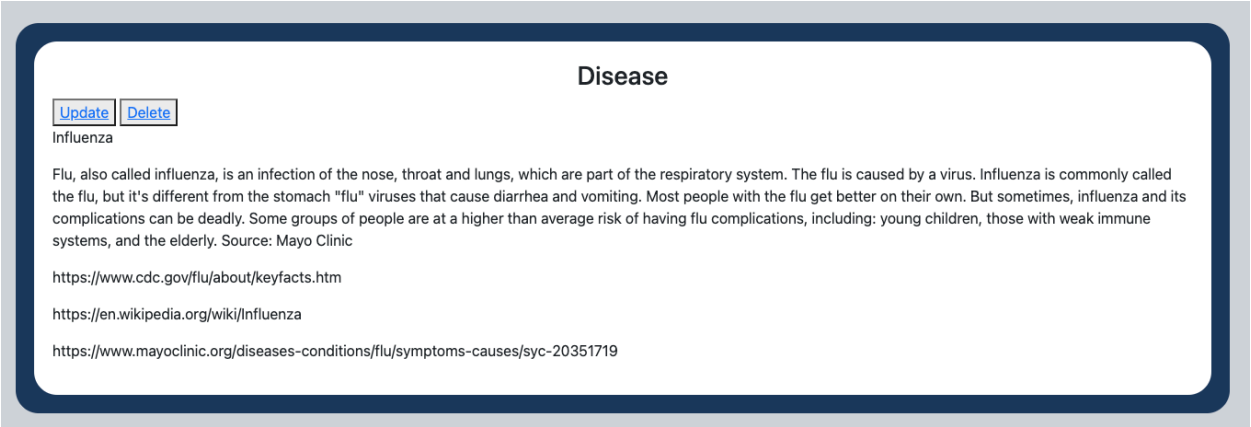
Use the Navigation Bar to access the various data resources managed by the portal. Selecting Disease Categories, Diseases, or one of the three Status Reports will take you to a page that will allow you to create, update, view, or delete resources of that type.

View All



When navigating to a data resource from the navigation bar, you will be presented with a table view showing all currently existing items of that type along with certain attributes of each item. On this page you are given the option to create a new item of this type through the Add New button, or click an item's hyperlink in the table to see a detailed description of the particular item.

View One

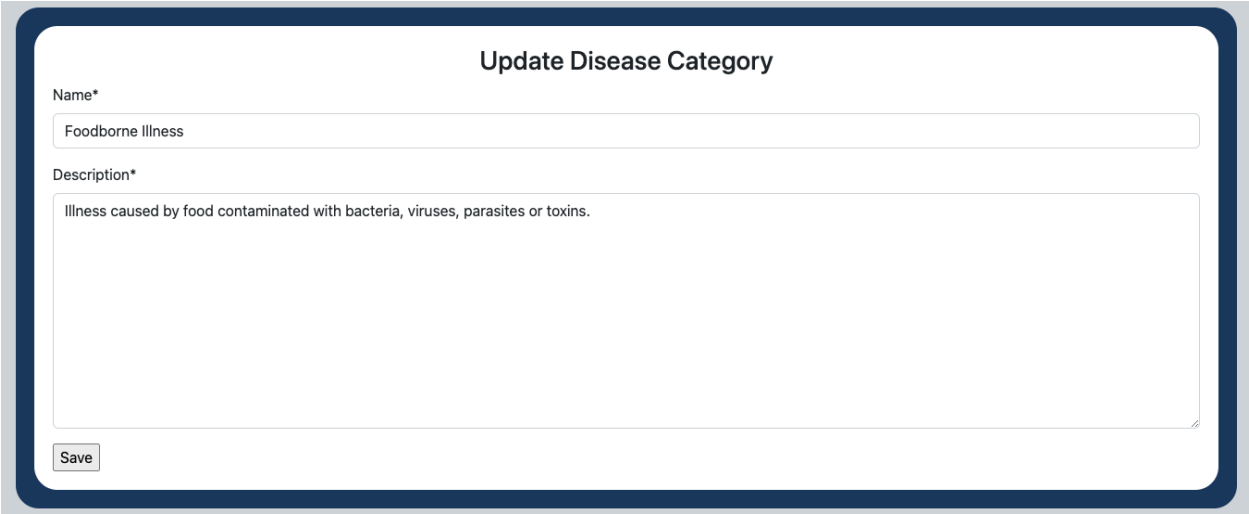


When viewing a detailed view of a single item you are able to potentially see additional information that is not present in the table view. From this page you also have the ability to update or delete the particular item through the Update and Delete buttons.

Create

Clicking the Add New button on a View All page for a resource will present the user with a form in which to enter information describing the new item. Once filled out, click the Save button to create the new item. The user will then be directed to the View One page of the new item.

Update



Clicking the Update button when on an individual item's View One page will allow the user to update information on that item. Simply modify the information in the form and click Save.

Delete

Clicking the Delete button when on an individual item's View One page will allow the user to remove the item. The user will be asked to click the Confirm button to perform this operation. If you do not wish to delete the item, simply navigate away from this page.

Note: Due to the hierarchical relationship between Disease Category -> Disease -> Status Report, the user will want to exercise care in deleting Disease Categories and Diseases, as deleting a Disease Category will also remove all existing Diseases mapped to them, and also all existing Status Reports tied to those Diseases.

API

The API (Application Programming Interface) for the Data Portal may be accessed at the following endpoint and is publically accessible (not requiring authentication/authorization).

https://hcdbackend.fly.dev/dataportal/_query2

The API returns a response in JSON format and may be used to query the Data Portal, and is used by the public facing Health Dashboard to pull data.

Combinations of URL Parameter query arguments are passed to the endpoint to pull the specific data the user is interested in. These include (key: <values>):

1. type
    a. disease_cateogory
    b. disease
    c. case_report
    d. death_report
    e. hospitalized_report
2. disease
    a. <disease_name>
3. diseaseCategory
    a. <disease_category_name>

Example: GET ALL Diseases

https://hcdbackend.fly.dev/dataportal/_query2?type=disease

Example: GET ALL Hospitalized Reports

https://hcdbackend.fly.dev/dataportal/_query2?type=hospitalized_report

Example: GET ALL Death Reports Belonging to a specific Disease

https://hcdbackend.fly.dev/dataportal/_query2?type=death_report&disease=COVID-19

Example: GET ALL Case Reports Belonging to a specific Disease Category

https://hcdbackend.fly.dev/dataportal/_query2?type=death_report&diseaseCategory=Foodborne%20Illness

Response format for status reports is designed to match as closely as possible with the existing Hamilton County COVID-19 Dashboard.