Monkeypox Dashboard

Felix Huang, Saketh Dendi, Sakin Kirti
For CSDS 393
On 12/01/2022

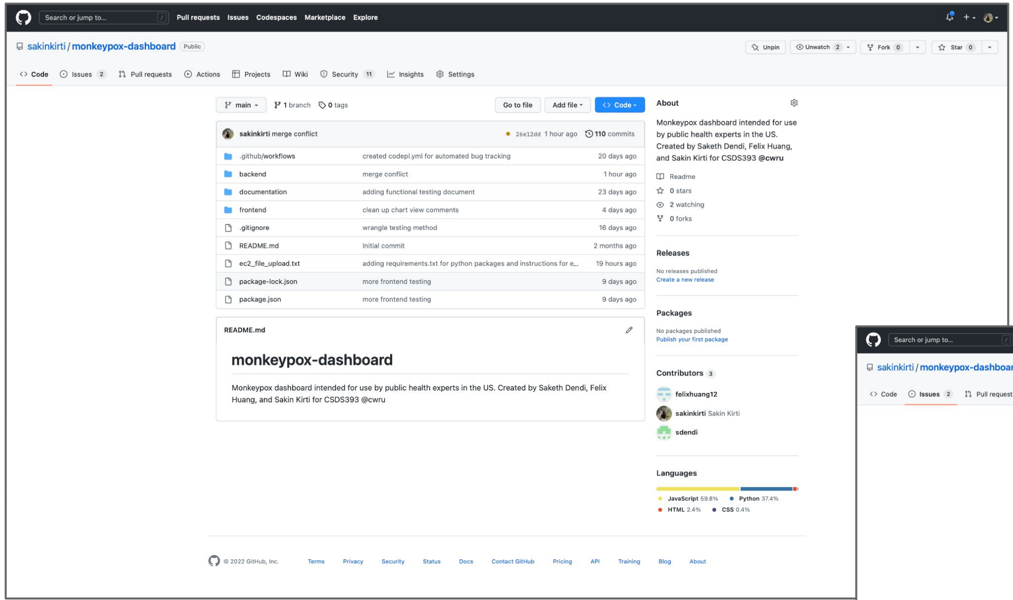
Project Background

Project Idea

Our idea was inspired by COVID-19 tracking dashboards that were created to track and visualize the pandemic. We thought that it would be a good idea to do this for monkeypox since the same resources aren't as widespread. Our platform takes the most up to date case data and aggregates it state by state. It also offers graphic representation of the cases by state and predictions for how the virus will spread in each state.

Why is it important?

When the COVID-19 virus hit we did not have the proper tools in place to respond to the virus properly. This led to an unnecessary loss of life and a much more difficult time getting through the pandemic. By creating tools like this for future viruses we can be better prepared if and when the next big pandemic hits.



sakinkirti / monkeypox-dashboard Public

Code · Issues 2 · Pull requests · Actions · Projects · Wiki · Security 11 · Insights · Settings

main · 1 branch · 0 tags · Go to file · Add file · Code

sakinkirti merge conflict · 26e12dd 1 hour ago · 110 commits

.github/workflows	created codepl.yml for automated bug tracking	20 days ago
backend	merge conflict	1 hour ago
documentation	adding functional testing document	23 days ago
frontend	clean up chart view comments	4 days ago
.gitignore	wrangle testing method	16 days ago
README.md	Initial commit	2 months ago
ec2_file_upload.txt	adding requirements.txt for python packages and instructions for e...	19 hours ago
package-lock.json	more frontend testing	9 days ago
package.json	more frontend testing	9 days ago

README.md

monkeypox-dashboard

Monkeypox dashboard intended for use by public health experts in the US. Created by Saketh Dendi, Felix Huang, and Sakin Kirti for CSDS393 @cwru

About

Monkeypox dashboard intended for use by public health experts in the US. Created by Saketh Dendi, Felix Huang, and Sakin Kirti for CSDS393 @cwru

Readme · 0 stars · 2 watching · 0 forks

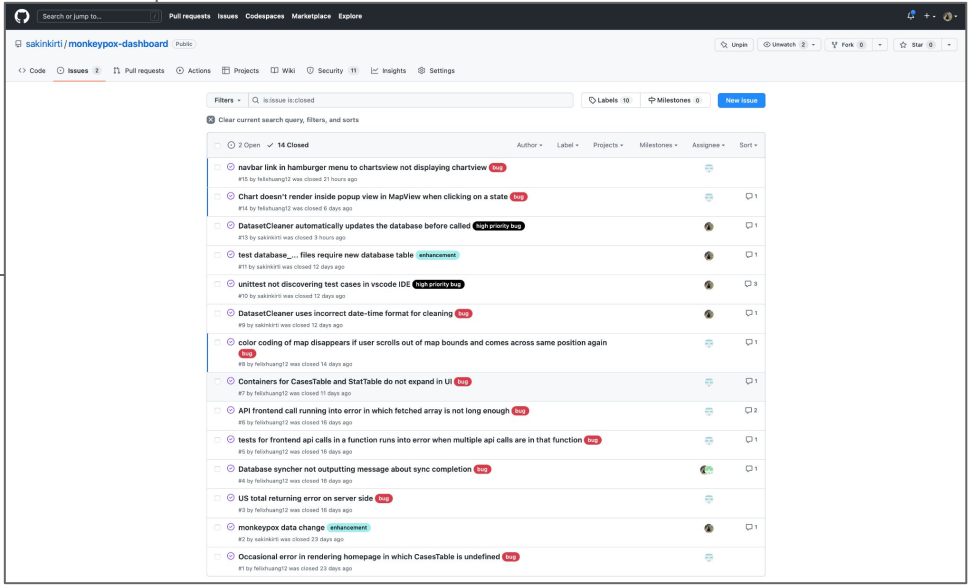
Releases: No releases published · Create a new release

Packages: No packages published · Publish your first package

Contributors 3: felixhuang12 · sakinkirti Sakin Kirti · sdendi

Languages: JavaScript 59.8% · Python 37.4% · HTML 2.4% · CSS 0.4%

© 2022 GitHub, Inc. Terms Privacy Security Status Docs Contact GitHub Pricing API Training Blog About



sakinkirti / monkeypox-dashboard Public

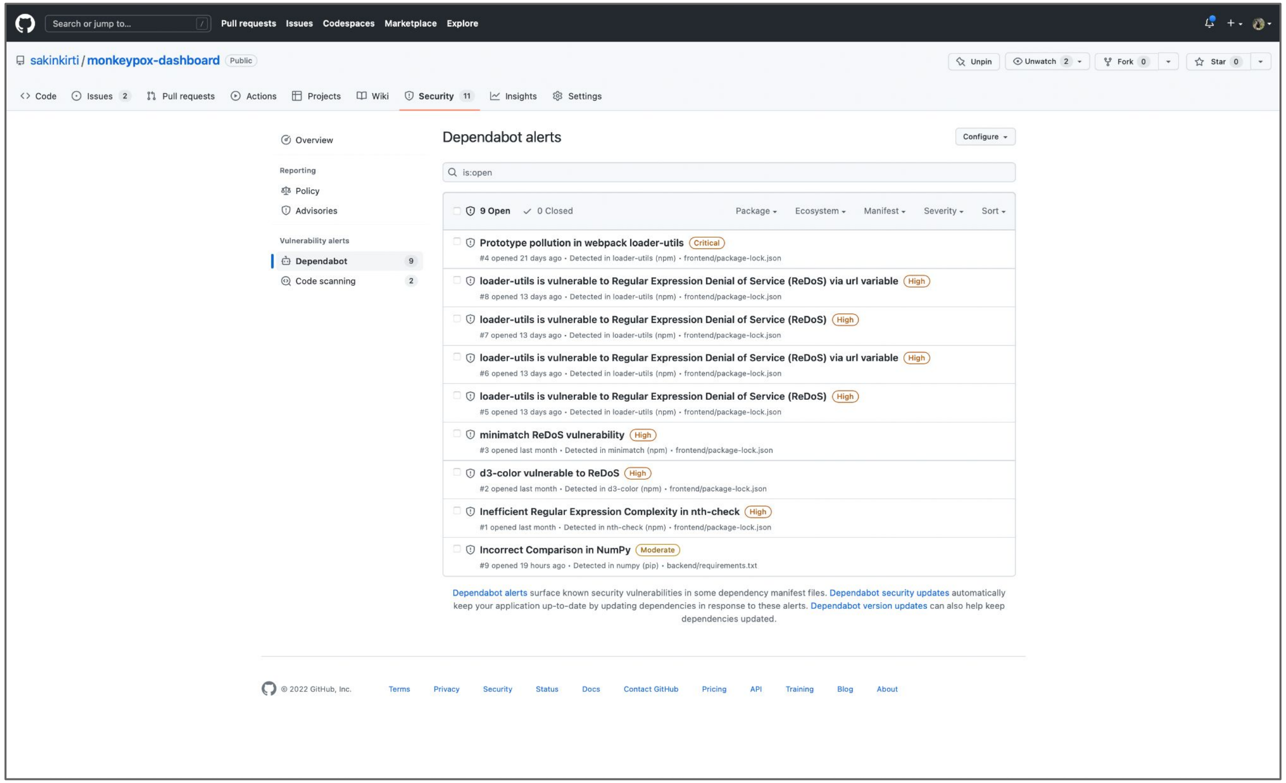
Filters · is:issue is:closed · Labels 10 · Milestones 0 · New issue

Clear current search query, filters, and sorts

2 Open ✓ 14 Closed

Issue	Label	Closed	Comments
navbar link in hamburger menu to chartsview not displaying chartview	bug	#15 by felixhuang12 was closed 21 hours ago	
Chart doesn't render inside popup view in MapView when clicking on a state	bug	#14 by felixhuang12 was closed 6 days ago	1
DatasetCleaner automatically updates the database before called	high priority bug	#13 by sakinkirti was closed 3 hours ago	1
test database_... files require new database table	enhancement	#11 by sakinkirti was closed 12 days ago	1
unittest not discovering test cases in vscode IDE	high priority bug	#10 by sakinkirti was closed 12 days ago	3
DatasetCleaner uses incorrect date-time format for cleaning	bug	#9 by sakinkirti was closed 12 days ago	1
color coding of map disappears if user scrolls out of map bounds and comes across same position again	bug	#8 by felixhuang12 was closed 14 days ago	1
Containers for CasesTable and StatTable do not expand in UI	bug	#7 by felixhuang12 was closed 11 days ago	1
API frontend call running into error in which fetched array is not long enough	bug	#6 by felixhuang12 was closed 16 days ago	2
tests for frontend api calls in a function runs into error when multiple api calls are in that function	bug	#5 by felixhuang12 was closed 16 days ago	1
Database syncher not outputting message about sync completion	bug	#4 by felixhuang12 was closed 16 days ago	1
US total returning error on server side	bug	#3 by felixhuang12 was closed 16 days ago	
monkeypox data change	enhancement	#2 by sakinkirti was closed 23 days ago	1
Occasional error in rendering homepage in which CasesTable is undefined	bug	#1 by felixhuang12 was closed 23 days ago	

Github made things really easy…



sakinkirti / monkeypox-dashboard Public

Dependabot alerts

is:open

9 Open ✓ 0 Closed

- Prototype pollution in webpack loader-utils (Critical)
  #4 opened 21 days ago • Detected in loader-utils (npm) • frontend/package-lock.json
- loader-utils is vulnerable to Regular Expression Denial of Service (ReDoS) via url variable (High)
  #8 opened 13 days ago • Detected in loader-utils (npm) • frontend/package-lock.json
- loader-utils is vulnerable to Regular Expression Denial of Service (ReDoS) (High)
  #7 opened 13 days ago • Detected in loader-utils (npm) • frontend/package-lock.json
- loader-utils is vulnerable to Regular Expression Denial of Service (ReDoS) via url variable (High)
  #6 opened 13 days ago • Detected in loader-utils (npm) • frontend/package-lock.json
- loader-utils is vulnerable to Regular Expression Denial of Service (ReDoS) (High)
  #5 opened 13 days ago • Detected in loader-utils (npm) • frontend/package-lock.json
- minimatch ReDoS vulnerability (High)
  #3 opened last month • Detected in minimatch (npm) • frontend/package-lock.json
- d3-color vulnerable to ReDoS (High)
  #2 opened last month • Detected in d3-color (npm) • frontend/package-lock.json
- Inefficient Regular Expression Complexity in nth-check (High)
  #1 opened last month • Detected in nth-check (npm) • frontend/package-lock.json
- Incorrect Comparison in NumPy (Moderate)
  #9 opened 19 hours ago • Detected in numpy (pip) • backend/requirements.txt

Dependabot alerts surface known security vulnerabilities in some dependency manifest files. Dependabot security updates automatically keep your application up-to-date by updating dependencies in response to these alerts. Dependabot version updates can also help keep dependencies updated.

© 2022 GitHub, Inc. Terms Privacy Security Status Docs Contact GitHub Pricing API Training Blog About

Github made things really easy…

Frontend Tools

- HTML: markup language for displaying documents in a web browser
- CSS: styling documents and React components
- React.js: open-source front-end JavaScript library for building user interfaces via UI components
- ChakraUI: library providing more accessible and cleaner React components
- Recharts: charting library built on React components
- Leaflet: JavaScript library for building web maps

Backend Tools

- All code written in Python
- PostgreSQL database (stored on AWS)

PostgreSQL

pandas

Psycopg2

NumPy

Flask

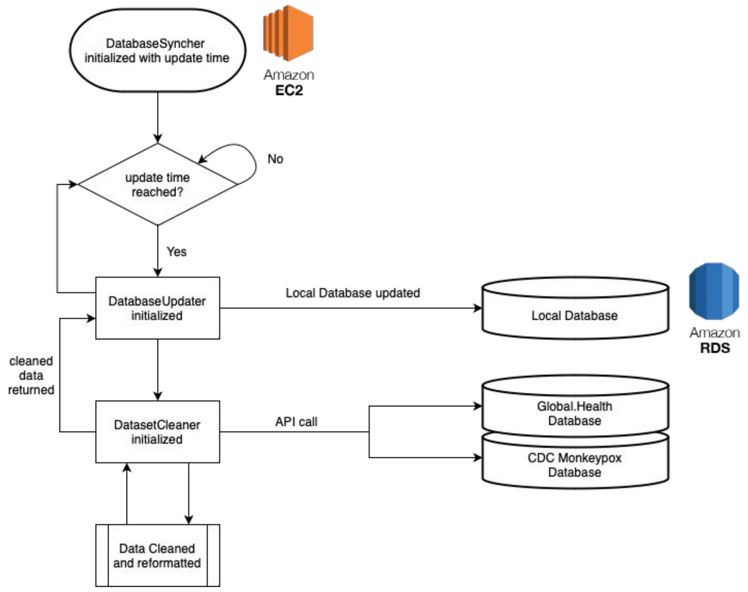
API Development

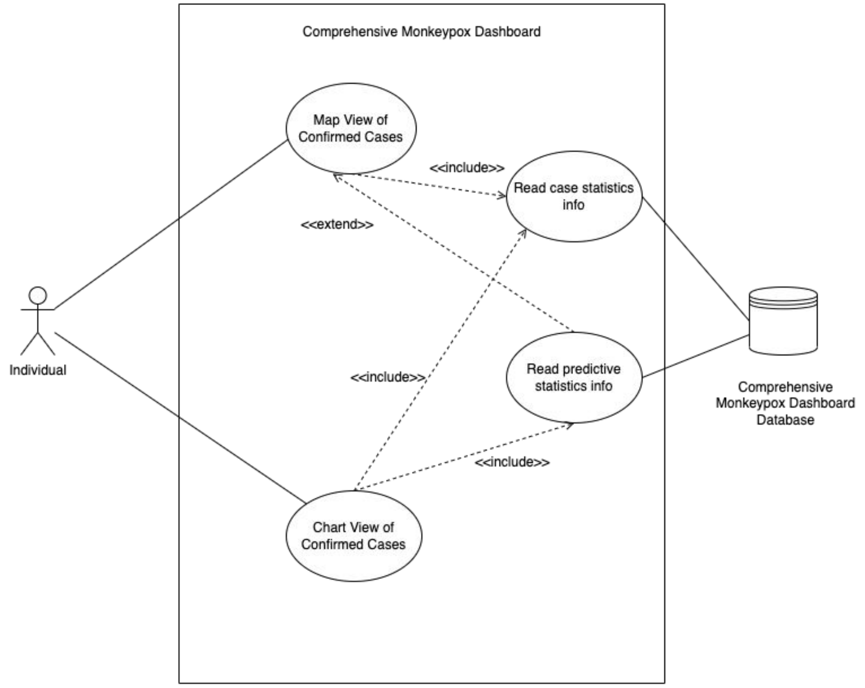
- Endpoints for the following:
  - Retrieving cases per day for a specific state from our database
  - Retrieving cumulative cases on each day for a specific state from our database
  - Retrieving U.S. cumulative cases
  - Retrieving 14-day case count predictions for a specific state
  - Retrieving public health statistics predictions

AWS RDS + EC2

Relational Database	Elastic Compute Cloud
- stores case counts, organized by state and date - stores public health statistics, organized by date - stores 14-day case count forecast, organized by state and date	- automates database updating (every 24 hours we pull data from CDC) - EC2 gives us a dedicated compute node

Backend and Frontend Process



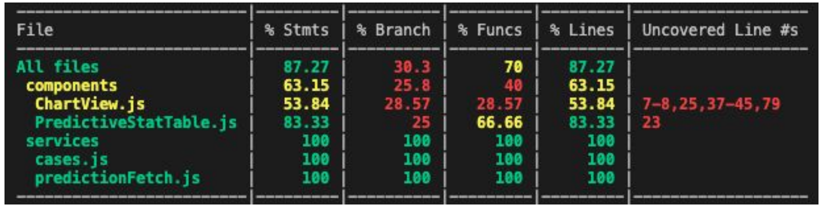

Comprehensive Monkeypox Dashboard
Map View of Confirmed Cases
<<include>>
Read case statistics info
<<extend>>
Individual
<<include>>
Read predictive statistics info
Comprehensive Monkeypox Dashboard Database
<<include>>
Chart View of Confirmed Cases

Testing and Code Coverage

<u>Frontend</u>

Framework

Coverage

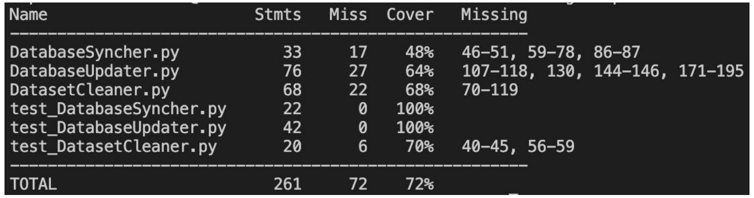


```
--------------------------|---------|----------|---------|---------|-------------------
File                      | % Stmts | % Branch | % Funcs | % Lines | Uncovered Line #s
--------------------------|---------|----------|---------|---------|-------------------
All files                 |   87.27 |     30.3 |      70 |   87.27 |
 components               |   63.15 |     25.8 |      40 |   63.15 |
  ChartView.js            |   53.84 |    28.57 |   28.57 |   53.84 | 7-8,25,37-45,79
  PredictiveStatTable.js  |   83.33 |       25 |   66.66 |   83.33 | 23
 services                 |     100 |      100 |     100 |     100 |
  cases.js                |     100 |      100 |     100 |     100 |
  predictionFetch.js      |     100 |      100 |     100 |     100 |
--------------------------|---------|----------|---------|---------|-------------------
```

<u>Backend</u>

Framework

Coverage



```
Name                        Stmts   Miss  Cover   Missing
---------------------------------------------------------
DatabaseSyncher.py             33     17    48%   46-51, 59-78, 86-87
DatabaseUpdater.py             76     27    64%   107-118, 130, 144-146, 171-195
DatasetCleaner.py              68     22    68%   70-119
test_DatabaseSyncher.py        22      0   100%
test_DatabaseUpdater.py        42      0   100%
test_DatasetCleaner.py         20      6    70%   40-45, 56-59
---------------------------------------------------------
TOTAL                         261     72    72%
```

Key Lessons

- Setting up Dataframe to Database connections take a lot more time than we originally thought
- Make sure you are using the most efficient solution to any given problem otherwise you will waste precious time and effort
- It’s best to delegate tasks at meetings and keep lines of communication open otherwise the task at hand will get extremely muddled