

# COVID-19 back2normal Dashboard

CAPP 30122 - Winter 2021

## Group Members

Christie Ibaraki - *ibarakicj*

Cole Frank - *cvf*

Shai Slotky - *slotky*

## Project Overview

From public health impacts to economic downturn, the COVID-19 pandemic--and requisite social distancing measures--have altered the lives of many Chicago residents. Using data from a variety of sources at the ZIP Code level, we built a dashboard that visualizes some of the changes in the daily lives of Chicagoans over the course of the pandemic. This project is primarily focused on observing the ways in which daily activities have and have not begun to return to their pre-pandemic forms in different parts of the city.

The data available for each ZIP Code includes:

- Demographic data - *race, employment, age, poverty level, and insurance coverage*
- Daily COVID-19 case and vaccine data
- Measures of daily activity
  - Five different categories of foot traffic (Restaurants, Bars, Parks/Beaches, Schools/Libraries, Grocery)

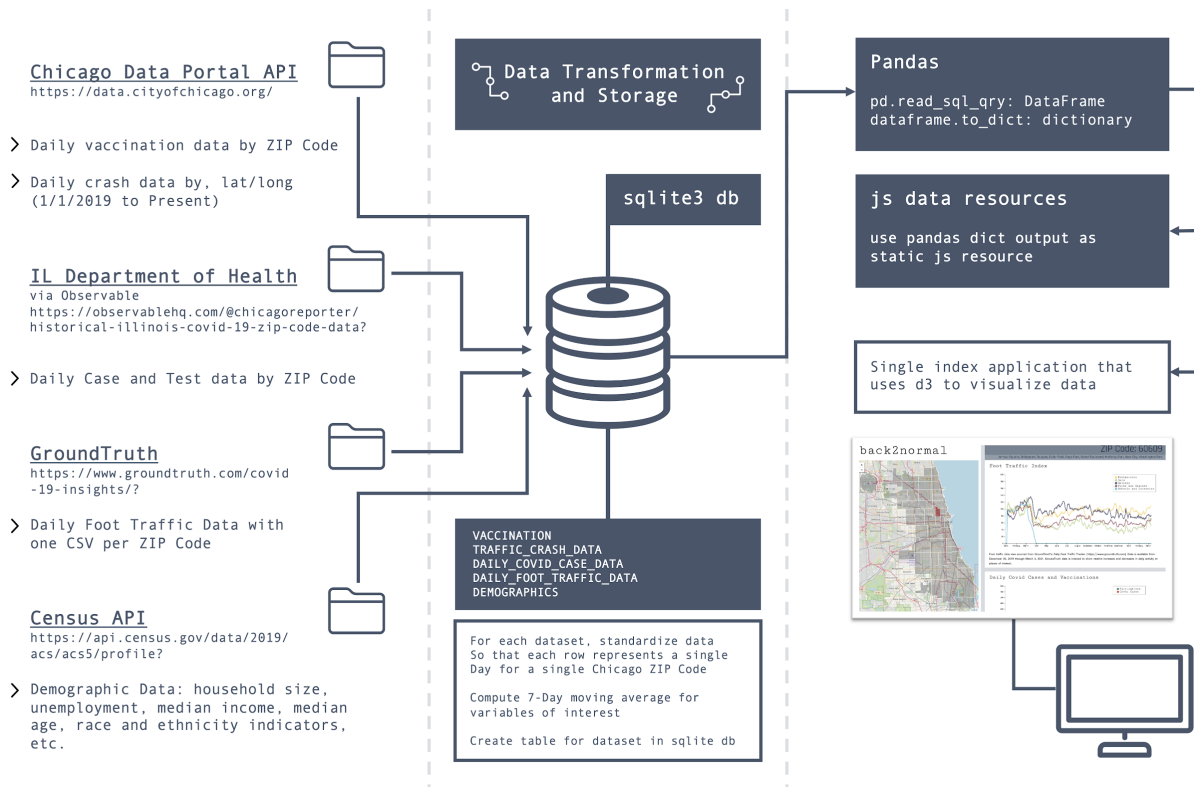
Overall, the goal of this project is to provide information on whether life in each Chicago ZIP Code is “back to normal” and if there are changes in this return to form according to demographic differences across ZIP Codes.

## Software Architecture

The backend of our project consists of an sqlite database of five tables. All of the code used to build the database is in the **core** directory.

As detailed in **the diagram below**, the first step was reading each of the respective data sources into a Pandas dataframe. The data came from the following sources:

- City of Chicago Data Portal - *accessed via API*
- The U.S. Census Bureau - *accessed via API*
- The Illinois Department of Public Health - *accessed via API*
- GroundTruth - *accessed via CSV download*
  - GroundTruth is a location-based marketing and advertising technology company that provides indexed foot traffic daily foot data at the ZIP Code level through the GroundTruth [COVID-19 Insights Daily Foot Traffic Tracker](#)



The Pandas dataframes were subsequently cleaned and standardized using functions we wrote in **core/data\_transformations.py** and then each was inserted into its appropriate SQL table. To facilitate this process we constructed classes for:

- 1) Handling the creation of or connection to an existing sqlite database (**core/data/dbclient.py**)
  - 2) Making requests to the city of Chicago Socrata API (**core/data/socrata/socrata\_api\_requests.py**)
  - 3) Building queries for the city of Chicago Socrata API (**core/data/socrata/soda\_data.py**)
- [Note: the **core/data** directory contains modules to retrieve data for all datasets including census, IDPH, and GroundTruth]

The code for the frontend of our project is in the directory **back2normal\_app**. We first queried the SQL database to create one static javascript data resource for each plot in our dashboard (see the .js files in **back2normal\_app/static/js** that start with 'data\_'). These static data sources are imported into our app via the HTML file (**back2normal\_app/home.html**) Each plot has a corresponding javascript file (see the .js files in **back2normal\_app/static/js** that start with 'plot\_').

The data is plotted using the D3.js library. Aside from the four graphs we plot with data from the database, we also constructed an interactive map of Chicago's zip codes using geojson data from the City of Chicago data portal. All HTML code is in **back2normal\_app/home.html** and all CSS code is in **back2normal\_app/css/back2normal\_app.css**. The result is a single index javascript application displaying the map and 4 graphs--all of which are built to update with new data when a new zip code is clicked on the map.

## **Project Goals and Assessment**

Our group hoped to track and visualize the ways in which COVID-19 has changed life across Chicago's many different neighborhoods, with the goal of answering the broad question: are things back to normal? Moreover, as a technical goal, we hoped to display our findings through an interactive dashboard, which we achieved. Ultimately, we built an interactive web app that allows the user to visualize trends in any of Chicago's ZIP Codes.

Initially, we envisioned this application as a django project that would make live queries to the SQL database we constructed. This would allow us to more easily regularly update the visualization. Due to time constraints, however, we instead built a javascript application that reads in data from static resource files in the project directory.

We also originally hoped to include more measures of activity from a wider range of data sources. For example, we pulled two years worth of Chicago car crash data (300,000+ records) from the city of Chicago API but ultimately did not have enough time to incorporate this data into our visualization.