

Final Report

Team Members:

- Baldwin Mugri
- Berkeley Poulsen
- Francis Ogbennah
- Leo Giovanni
- Nicholas Dougherty
- Sodbileg Ganbat

Link to our [Food Inspector App](#). Click to open.

Information Problem our team is trying to solve:

The problem that this application tackles is the lack of information available to the public on a food establishment's compliance with health and safety regulations. The goal is to present the inhabitants of Prince George's County with good and clean places to eat. These inhabitants are the key stakeholders, but anyone living in Maryland or passing through the area can seek to benefit from the application. Our solution to the information problem was to provide users with a visual representation of good places to eat based on their standards and the establishments compliance with the terms of our dataset.

Targeted Stakeholders & Browsers:

The Food Inspection Registry Application is targeted to be used across Android, iOS, and online browsers. We want this application to be available to as many users as possible in as many ways as possible. As of right now, the application is available through online browsers (Chrome, Safari, Firefox, Opera), but the long term goal is for there to be apps on the App Store and Google Play Store for this application.

Our Dataset:

The Prince George's County Health has provided us with a dataset that covers investigations into the county's food service facilities. It is a hearty dataset with 31,994 unique entries and consists of 29 columns that can be divided into the following categories: Establishment Description, Inspection Details, Address, Food Safety, Proper Holding Temperature, Safe Cooking Temperature, Safe Food Handling, and Facility Hygiene. This dataset was chosen because of its comprehensive overview and how it relates to the prominent audience of PG County inhabitants, whom of which we are targeting.

Chosen Strategies & Solutions for the Problem

Our solution to the information problem was to create a web-based application that would provide users the ability to search and filter through categories to find food providing facilities. The search output of facilities show whether the facility was in-compliance with the health & safety policies set by Prince George's County Health Department.

Technical System Decision Rationale:

As a result, our team decided to build a web application similar to Yelp or Google Maps that would provide users the ability to locate and navigate to nearby restaurants based on their category selected. Furthermore,

depending on the starring system used, users will be able to select clean and safe dining facilities, and avoid ones that are not in-compliance with the health & safety standards set by PG County.

For our project we decided to use the following systems:

- *Javascript/HTML/CSS*
 - The bulk of both our back-end and front-end coding. We used Javascript, HTML, and CSS because the majority of us had the most experience with these languages, and as such, would allow us to collaborate on the project efficiently.
- *Google Map API/Map Javascript API*
 - Was our initial choice to create our web application map, but was later scrapped to use Leaflet instead as it was simpler and better when integrating with the rest of our design.
- *Leaflet*
 - Lightweight and user-friendly Javascript library used to build web mapping applications. We decided to use Leaflet over Google Maps when building our geolocation web map.
- *Heroku*
 - Heroku is a cloud platform that we used to host our Food Inspector Application website. We decided to use Heroku over other service providers such as Amazon AWS because it is relatively simple to use, free, and has a GitHub integration that helps a lot. Furthermore, it was the recommended application to use in class.

How our System Addresses the Problem:

Our final system helps to address the problem we were trying to solve by giving users the opportunity to search for restaurants that they can trust in. While rare, there is the chance of being the person that goes to a restaurant that is not health code compliant and no one wants to be the unlucky customer to get sick from one of these problems. With our system, a user can make a decision on where they want to eat and feel confident that it was the right decision.

Challenges & Changes in Design:

We ran into the following issues while working on our project prototypes:

- Using/connecting to Heroku
- User Information Overload
- Aggregate Data Values
- Adding a Map with a Marker

a) Leading up to our final presentation our group encountered some difficulty hosting our pages on Heroku. We were thinking of alternatives such as Amazon AWS, but decided to stick with Heroku. Eventually, one of our teammates realized we were missing a key code in our files that was causing our Github Integration to result in webpage being timed-out.

b) Another challenge we faced while entering into our final design was how much information we wanted to appear to the user at face value. We were going for a design that would not put too much information in front

of them so that they would not be overwhelmed. In terms of our design, when we want to give health code compliant information, it can be challenging to display that information without it feeling like too much. Our solution was to make large charts. This would allow for it to maintain a structure around the restaurants listed without making it look like an excel sheet. We also included that only a certain number of results would be shown at a time. The result is a design that is easy to look at and reduces the overwhelming possibility of too many options.

c) Our third challenge was using aggregate data values. These kinds of values may misrepresent the truth behind the data. Using visualizations presents much more information in a single object than a bunch of objects with both aggregated and multi-dimensional data. The comparison of two aggregated multi-dimensional numbers is like comparing fruits. Both objects are fruits and they might both be apples, but one might be a tart granny-smith while the other is a sweet Fuji apple. The characteristics of both apples vary as much as the variables of two aggregated numbers. As a result, we decided to leave out the aggregate metrics from our final design. Furthermore, we decided to present our information at the overview-level, which minimized unnecessary information overload, and provides users ability to interact with the application without cluttering.

d) Our fourth and final large obstacle in our design process was creating a geolocation map. We initially attempted to use the Google API Key and Maps Javascript API to create a map centered on Prince George's County, Maryland, United States of America. Our team worked on this issue for quite a while, and we finally found a solution that worked. However, when we implemented the code into our files, even though our "Click to use current location" button was working, the map itself was not displaying correctly on our webpage. As a result, the team decided to use Leaflet which was a much easier solution to utilize and implement into our design.

Possible Future Development on Problem:

There are many possibilities for the future of this design. While we only currently show three health code compliances, there are so many more that we could offer. With continued changes in the future, in terms of design and information, a user will be able to determine what places they should eat at based on the options, allergies, a handbook worth of health code compliances, and other filters such as ratings by other users of the application. Furthermore, as mentioned before, adding a Yelp-like feature or Google Maps-like feature that provides the user restaurant details including: Opening Hours, Images, Address, and Ratings. We believe providing a Yelp/Google Map customer ratings in conjunction with a possible In-Compliance Health rating would provide a good holistic review of each restaurant to provide quick and easy-to-grasp information to users.