Team 8: The Political Market

Technical Report

GitLab URL: https://gitlab.com/kevinchenftw/thepoliticalmarket/

Kevin Liang, Kevin Chen, Diyuan Dai, Anisha Kollareddy, Vaishnav Bipin

1. Motivation

Money in government has always been a significant political discussion point, and sometimes seems to have gained traction over the last few years. As a team, we were particularly curious about the data surrounding this topic, and felt we could fulfill a civic duty by educating the interested public about the relevant data. Elected members of Congress are expected to represent their constituents in their home state. But at the same time, it would also be in their interest to consider the major companies in the state as well. Industry success creates jobs and stable economies, so there is already a major connection between companies and politicians.

Another metric we were interested in was the stock market. We had the original idea to compare market trends with Congressmember portfolios, since they are required by law to disclose them. But there wasn't as much transparency as we hoped in perusing the data, as a lot of investment portfolios were with external companies and sources. But the idea of the stock market as a whole proved to be valuable, as it was a metric that could directly evaluate a company and compare them to other companies and current events. The market is indeed always fluctuating, but the hindsight of aggregated data is absolute.

Finally, the government as an entity can spend significant money on a variety of contracts. Likewise, it is a great benefit to a company to be selected for a government contract. A Congressmember is still part of the government, and as an elected official, they also benefit in their campaigns to receive money. Campaign financing is also a form of money in government, and can come from many sources. We hope to take the data collected from our three models and delve into their connections; the stock market, government contracts, and Congressmember finances all certainly relate closely to one another, and we are very interested to see what sort of interconnectedness we can identify and visualize.

2. User Stories, Phase I

DoggieDB User Stories:

We provided User Stories for the DoggieDB Team. Their website is meant to help users find where they can adopt a specific breed of dog, and ideal places for that dog to live in. We asked them for basic website content, like an about page and navigation bar. As Texas residents, we also asked them for dog breeds found in shelters in Texas and for dog breeds compatible with Austin.

1. **Gitlab Statistics on About Page:** As users, we would want to see each user's contribution displayed on the About page of the website. The data should be accurately obtained from the GitLab API and consider all commits relevant to the production. It should also be clear what type of contributions are being measured.

2. **Navigation Bar:** As users, we would like to see a navigation bar so we can easily go from dog breeds, to shelters, to cities, etc. We want to be able to navigate between models quickly if we are trying to piece together many attributes to make a decision. We also want to make sure we don't miss any important aspects of the data if they are not clearly labeled on a navigation bar.

3. **Dog Breeds Model with Cities:** As users, we want to be able to see dog breeds with their compatible cities displayed on the site. We have our own preferences and favorite breeds and being able to see all their corresponding compatible cities would help make educated decisions. We might also be interested to see if this data corresponds with the breeds of dogs we see most often if we live in these cities.

4. **Compatible Breeds for Austin, TX:** We would like to see the dog breeds compatible with living in Austin, TX, or other Texas cities. As students here, we are curious about the alleged consensus that Austin is a dog city. From our daily activities and our other Texas hometowns we have a local connection, and are curious if our experiences seeing dogs aligns with the data presented.

5. **Dog Shelters in Texas:** We would like to see the dog shelters of Texas on the site, as a user who is trying to make the decision to adopt a rescue. We would consider data about the shelters like their location and hours in order to visit at the proper times. We also might be curious about data like adoption rates or other reviews from fellow dog enthusiasts.

The Political Market User Stories:

We were asked by our “users” for an about page with contribution statistics as well as instances for each of our data models.

1. **About Page Statistics:** As your client, I want to be able to see gitlab statistics in order to see each user’s contribution to the project. If possible, I would like to see these in cards. This will show me how active the team is working on the project.

2. **Campaign Finances Model:** As your client, I want to be able to access campaign finances and filter by contributors, amount, and time. This could be displayed as a table or as a grid. This would allow users to understand the impact of donations on political campaigns.
3. **Government Contracts Model:** As your client, I want to be able to access a "government contracts awarded" component page. It would be helpful to have access to instances with data pertaining to contractors, amount received, etc.. I would love if this could be in a grid/table.
4. **Stock Market Instances:** As your client, I want to be able to access stock market data. It would be helpful to have this data in the form of a table. This would enable users to understand stock information for different politicians
5. **Links between data models:** As a client, I want to be able to link between model pages. For example, I should be able to go to a government contract from a related stock. This would allow the user to understand the connection between contracts and stocks

We were able to complete all our user stories without much hassle, though one story had to be adjusted due to our decision to omit a less-productive model from our RFP step. We all definitely gained important insight by thinking about developing or requesting these issues from the user perspective.

3. RESTful API: Postman

We documented our RESTful API based on the API calls we explored with our preliminary scraping of the data in our model sources. Other API calls reflect our plan to aggregate the data in hopes of easily accessing potential patterns areas.

Government Contracts:

- Government Contracts Endpoint
    - Returns list of government contracts awarded.
- Government Contract Endpoint
    - Returns a specific government contract specified by ID.
- Government Contract Vendors Endpoint
    - Returns a list of government contract vendors.
- Government Contract Vendor Endpoint
    - Returns a specific government contract vendor as specified.

Stocks:

- Stocks Endpoint
    - Returns a specific result based on a symbol provided.
- Stock Endpoint

Campaign Finance:

- Campaign Finance Model Endpoint
    - Returns a list of campaign finance models by congress members.

Candidates:

- Candidates Endpoint
    - Returns a list of candidates.
- Candidate Endpoint
    - Returns a specific candidate as specified by ID.
- Candidate by State Endpoint
    - Returns a list of candidates by specified state.

4. Models

All our models and specific instances are rich in attributes. We have listed a few that we can potentially capitalize from our API sources, and our direction after Phase I will likely depend on what trends and links we happen to notice as we work further with scraping and storing.

Government Contracts Awarded:

- Contractors - Information about various government contracts including their industry and homestate, and other relevant states
- Amount Received- Amount of money received per contract
- Service Provided - What was provided as a condition of the contract
- Time Awarded - What time was award given out to the contractor
- State- What state the contractor's HQ is located.

Stock Market Data:

- Company - including industry and other states of operation
- Volume - the number of shares or options traded

- Long-term performance - possibly expressed as percentage over time frame
- Quarterly Info - meet, exceed, or fall behind projected expectations
- P/S value - a ratio comparing market capitalization divided by sales
- Dividends - certain stocks that distribute a portion of earnings to investors
- Recommendations - analyst ratings including strong buy, buy, sell, etc.

Campaign Finance:

- Politician - including political affiliation and represented state
- Contributors - Contributors to their financial campaign
- Amount Received - Amount received for campaign
- Time Received - When contribution was received
- Repeated Contributions - Any repeat contributions
- Election Result - Whether or not said candidate won

5. Tools

The tools we used fall into three categories: development, backend, and frontend. Our development tools allow us to collaborate both as team members as well as with other teams. Our backend tools will eventually allow the site to perform programmatically intensive tasks like dynamically requesting data from APIs and processing this aggregated data in many ways. Our frontend tools allow us to construct the ideal user experience, and every visitor to our site should be able to easily navigate our models and conveniently view the data.

Development:

- Postman - creates and hosts ThePoliticalMarket RESTful API Documentation
- Gitlab - issue tracking, project management, version control, continuous integration

Backend:

- Docker - gives our specific environment for the backend for any user or provider
- wsgi - allows our web server to forward requests to a Python web framework
- Flask - the lightweight Python web framework communicating with through wsgi

Frontend:

- yarn - a package manager for our modules, allows us to locally compile and develop
- React - a library to develop and render our User Interface and visual components
- React-Bootstrap - a library to further construct and stylize our frontend interface

6. Hosting

Our site is hosted through Amazon Web Services, with Namecheap providing the domain name and relevant redirects. Within AWS we use S3 and Cloudfront. S3 is essentially a secure container that stores all our files, assets, and source code. Cloudfront is the Content Delivery Network that is linked with S3, meaning it can securely send the necessary content from our website to users. Namecheap provides the name of our project as a domain name.

We had to link the two hosting tools together by requesting a certificate from Amazon. Then, to validate the certificate, we then had to put specific CNAME fields into our Namecheap configuration. With the valid certificate, Cloudfront and Namecheap were able to essentially connect. Once a user navigates to our Namecheap domain name, our website would be smoothly delivered by Cloudfront.

7. Audience and Users

Even in this initial Phase I of the project, we all noticed a difference in having software developers as our audience. Their user stories were comprehensive, and as fellow developers, we had a very clear direction on how to meet the requirements so far. They were very realistic on their expectations, which was likely due to the fact they were also working on the same components of the project. There was also a good communication pipeline between us, and we were both familiar with how to navigate the GitLab issue system as well. Approaching the end of the phase, we all certainly appreciated the clarity and contents of the user stories they provided us and the convenience of the shared GitLab platform.

In the later phases, when we start to work with our backend and implement our API, these same strengths will certainly apply plus more. Since our audience will be querying our API to get data from our project, they should be quick to notice if there is anything missing or inconsistent. User stories might get more technical, but it might also help to step back and consider other aspects of the experience from a non-developer user as well.