the political grid project

The Political Grid Project

Design Documentation
&
Application Programming Interface

Project by:
Tanyoung Kim

Developed by:
Tom Bellitire
Michael Dandy
Thomas Lester
Justin Roberts

February 1, 2011

Georgia Institute of Technology
Atlanta, Georgia 30332

Product Vision

Motivation and Opportunity

Political Grid Project is a web application that allows a user to rate politicians' tweets based on their validity and trustworthiness. As the most popular micro-blogging service, Twitter has played a significant role in defining social trends and influencing people's political stances. The popularity of a Twitter user, including politicians, oftentimes is measured by the number of tweets, retweets, mentions, and/or followers. We believe that these parameters do not provide enough information about where the public stands on a political remark. To follow a politician does not necessarily imply that he or she supports that politician. Moreover, the number of retweets and mentions may not suggest high level of consensus. In order to overcome these limitations and to further analyze the complexity of Twitter, Political Grid Project provides a mechanism in which a user can vote on political tweets based on two parameters: agree/disagree and important/ not important. Unlike other Twitter applications on the Internet, this app presents a graphical representation on how the public thinks and feels about certain issue rather than merely listing the most popular tweets. Furthermore, it allows the user to examine his or her political stance and accordance with politicians, parties, and other political bodies. Currently, there are no other web applications that aim at this goal. Hence, we build Political Grid Project in order to fill this niche.

Problem Statement

The problem with the current method of viewing a politician's popularity on Twitter is that it projects misleading information to the majority of public observers. This miscommunication can cause the statistics drawn from twitter to be error-prone and inaccurate. We believe that a successful solution would be an application that allows users to voice their opinion on a political remark posted to Twitter based on whether or not they agree with it and whether they believe that the comment is trustworthy. This solution would provide a means for graphically displaying the users' collective opinions on a topic, allowing a more reliable source of information for the basis of instantaneous political opinion.

Product Position Statement

For all public observers who follow and read politicians' blogs on Twitter, our system is a web application that allows user to further analyze the politicians' tweet based on its validity and trustworthiness and provides information graphics about majority agreement. Unlike other Twitter apps that merely show the most popular tweets based on Twitter criteria such as number of tweets, retweets, mentions, and/or followers, our product provides extra dimensions to further inspect the complexity of Twitter in defining social trend and political stance.

User Demographics

With respect to system user, there is only one type of user in this Political Grid Project: the System Admin. As an admin, the user can view all information stored in the database, in which it will be represented in some sort of information graphics that can be analyzed

such as grid, network graph, or chart. Our client, Tanyoung, will use this web application mainly as part of her thesis research. She will analyze how social networking can affect public opinion. As a System Admin, she is able to navigate through the Web and analyze and interpret the informational graphics presented by the application.

With respect to targeted audience, there are two types of user: those who participate in evaluating political comments and those who observe the majority consensus of a political comment.

- **Political Participants** can utilize our application to criticize a tweet based on two parameters: agree/disagree and important/not important. This will help them to define their support of a politician. The participants will be able to navigate through the Web and understand basic and common web UI.
- **Political Observers** would be able to view where the majority stands on certain political remarks and how the public views a politician's trustworthiness. Similar to participants, the observers also have some knowledge on how to navigate through the Web. Furthermore, they can be expected to be able to interpret information graphics.

Features

See Pivotal Tracker stories at https://www.pivotaltracker.com/projects/457151.

Constraints

- Dependency: Application depends on Twitter API, JavaScript, and JQuery. Users are expected to have JavaScript enabled on their Internet browsers.

- Availability: Application must run 24/7 and avoid traffic congestion caused by the Twitter server, i.e. carefully manage Twitter API calls so that it will not surpass the given bandwidth (150 calls/ hour).
- Portability: Application must run in IE 9, Firefox, Safari, and Chrome on any desktop/ laptop.

Design Architecture

Software Architecture

The Political Grid Project implements a multi-tiered client-server architecture, in which the client only displays data that has been queried to the server. This model is the best choice since web applications are cross platform; not only should our application run on any operating system, but it should also supports both desktop and mobile viewing. Furthermore, this architecture does not require users to have any advanced computer knowledge. This is demonstrated in that our web application only requires a JavaScript enabled browser, while a desktop application requires the users to go through an installation process. As a result, this project can target a wider audience. Finally, Twitter has an API that can be accessed through the REST protocol. By extending the architecture model, the server can also act as a proxy that communicates to the Twitter servers.

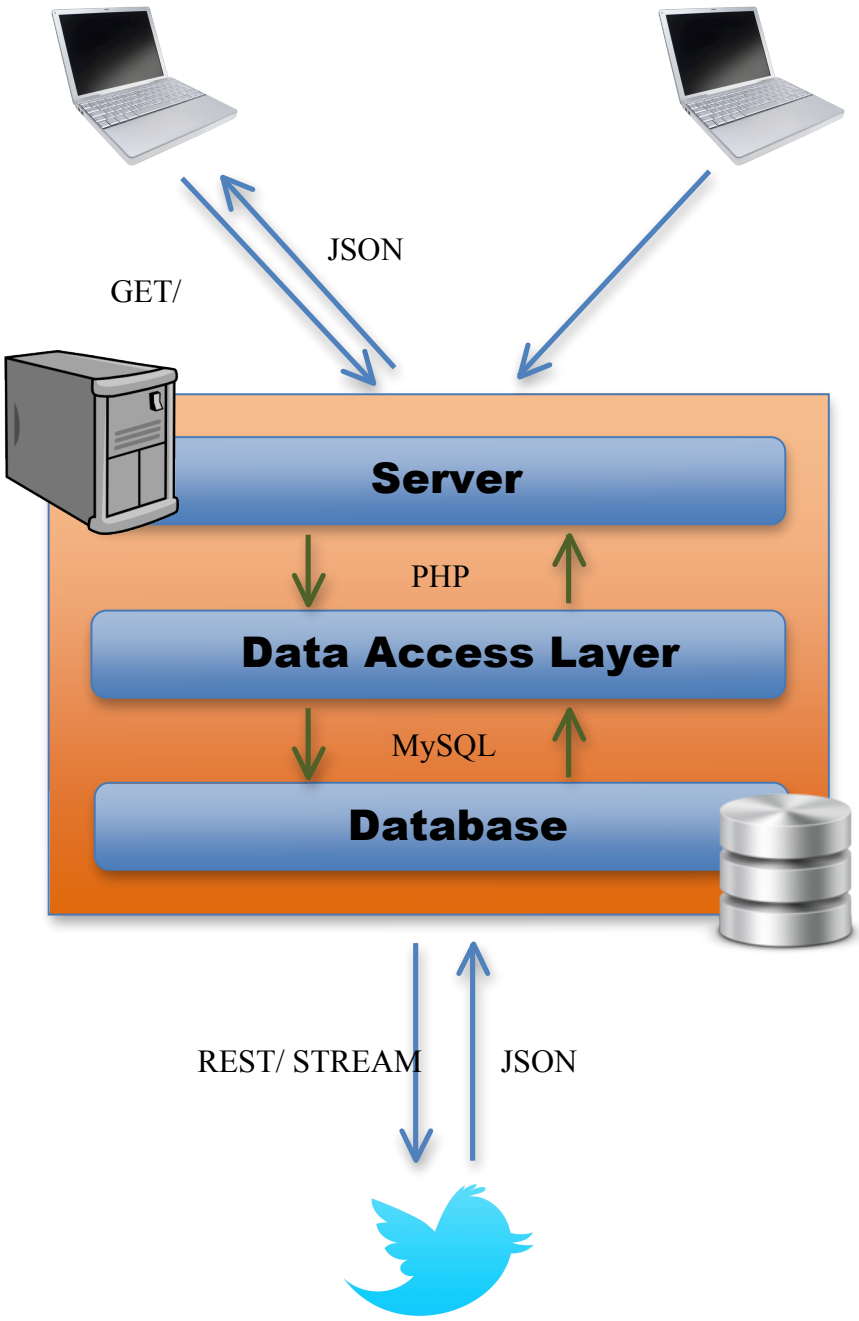
Design Rationale

Although a fat-client may have better performance over thin-client for the amount of work can be distributed evenly among the clients, this project will be based on thin-client model. It is due to Twitter API limitation. Not all of the Twitter API supports JSONP that allows cross-domain AJAX request. Therefore, information that the clients can retrieve from Twitter is limited. A server is needed to request such information on behalf of the client in order to circumvent this limitation. Furthermore, this client-server model will be based on a multi-tier architecture, in which presentation, logic, and data act as separate

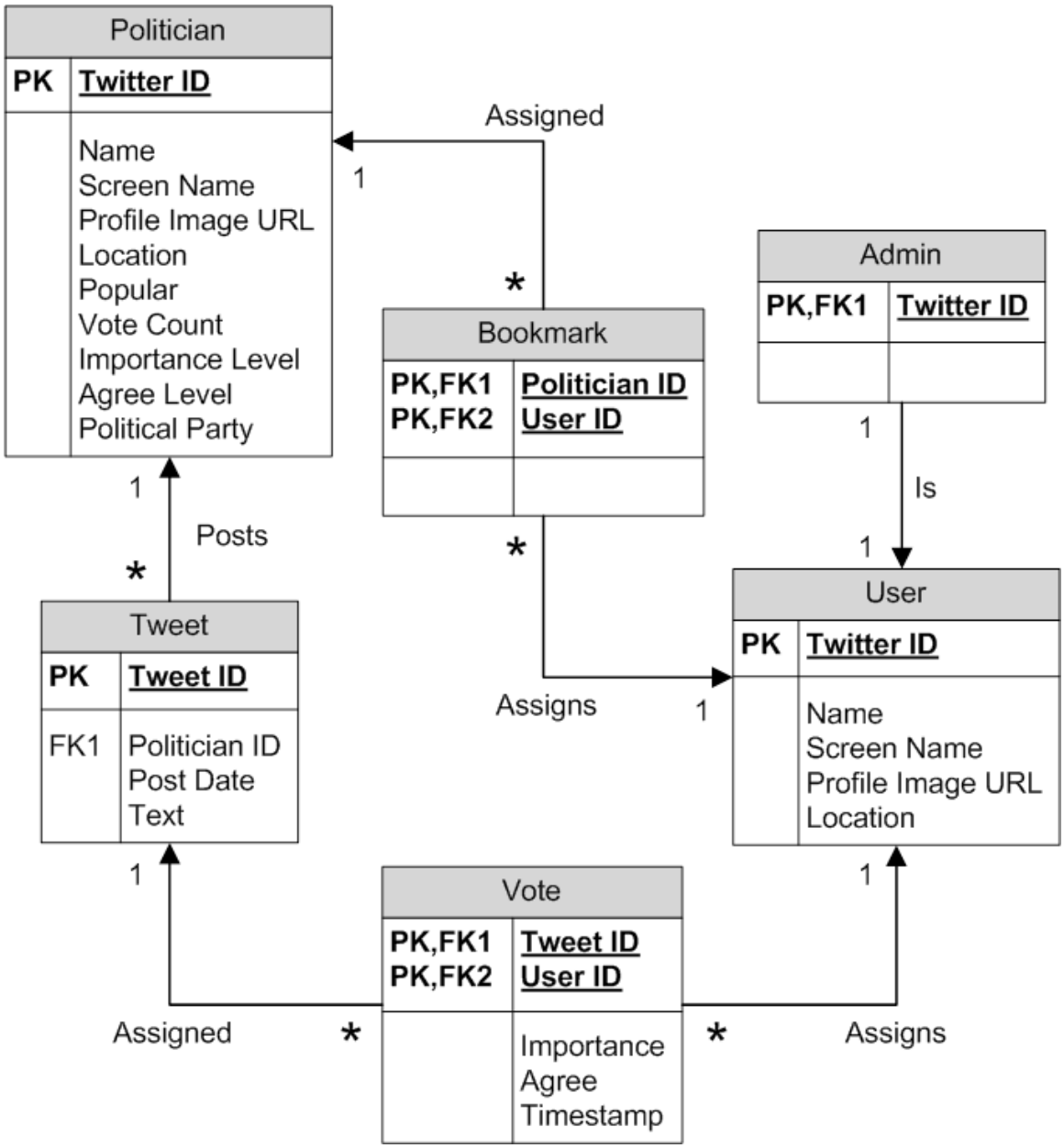
modules. Each module is secluded from the other in order to increase maintainability, i.e. if one module is broken, it will not severely impact the other module. Moreover, each module can be developed independently, making the design process easier. In this project, the presentation layer will be written in HTML, the logic will be written in PHP, and the data access layer will be written in PDO/MySQL.

Besides layered architecture, we also considered event-driven architecture. However, we concluded that this method would not be the best option because the main function of our web application is providing real-time information such as popular politicians, recent tweets, and users' vote. Our user is provided a stream of information despite their requests for such data. In other words, no event is required to update the state of the application and the user will always be kept up to date with the data that our application provides.

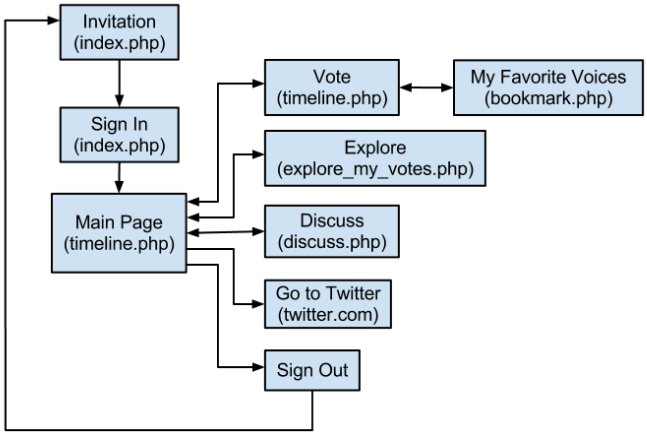
System Context



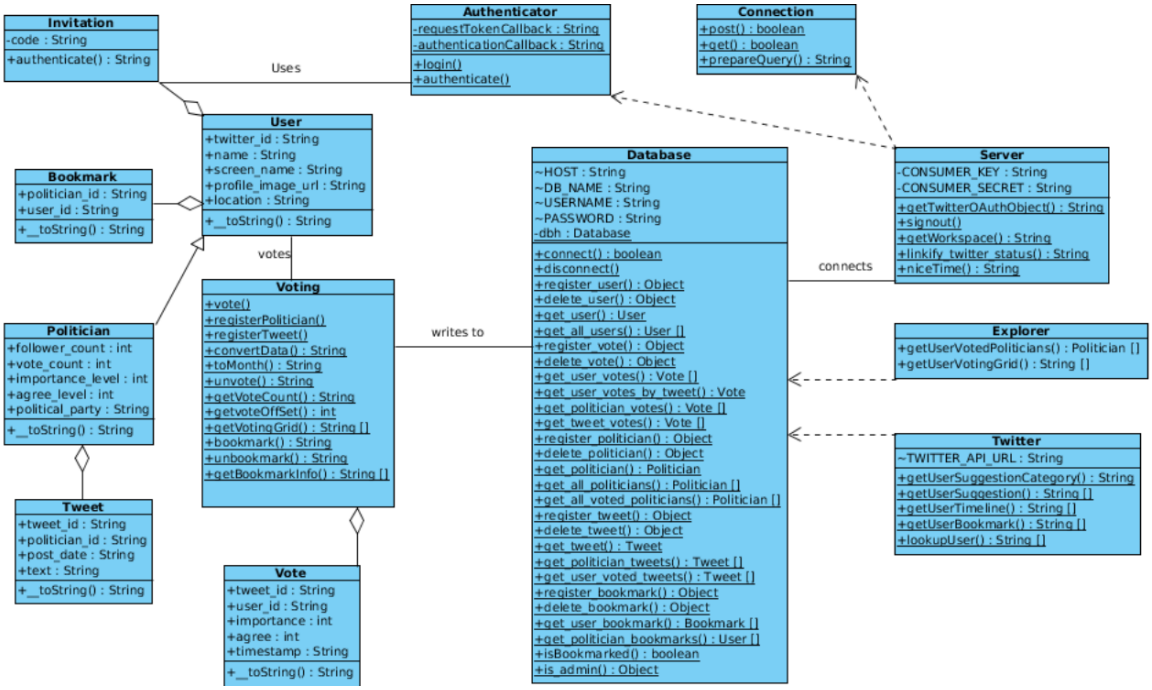
Database Entity-Relationship Diagram



Sitemap

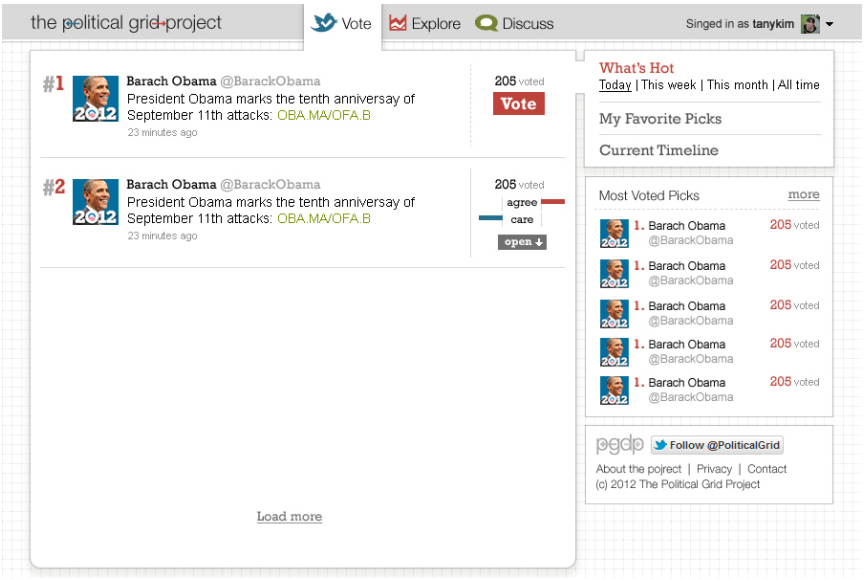


Server UML Diagram

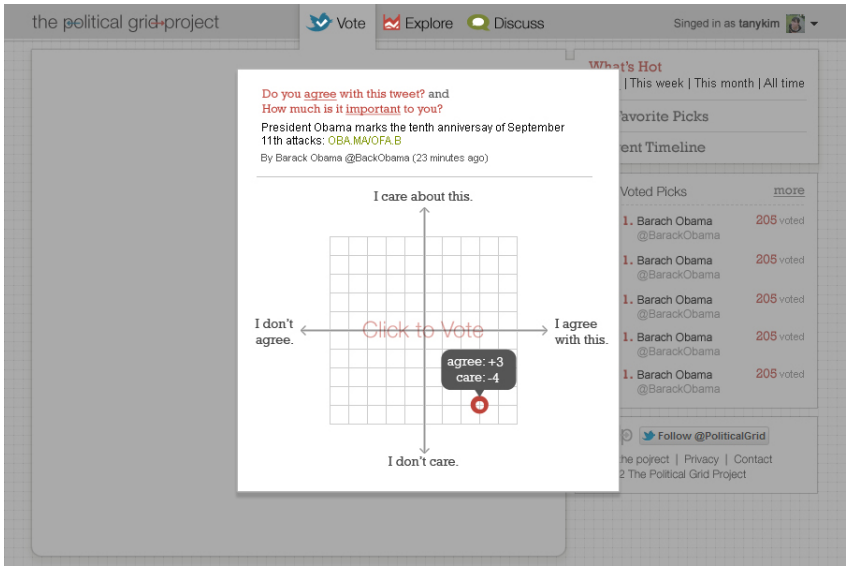


GUI

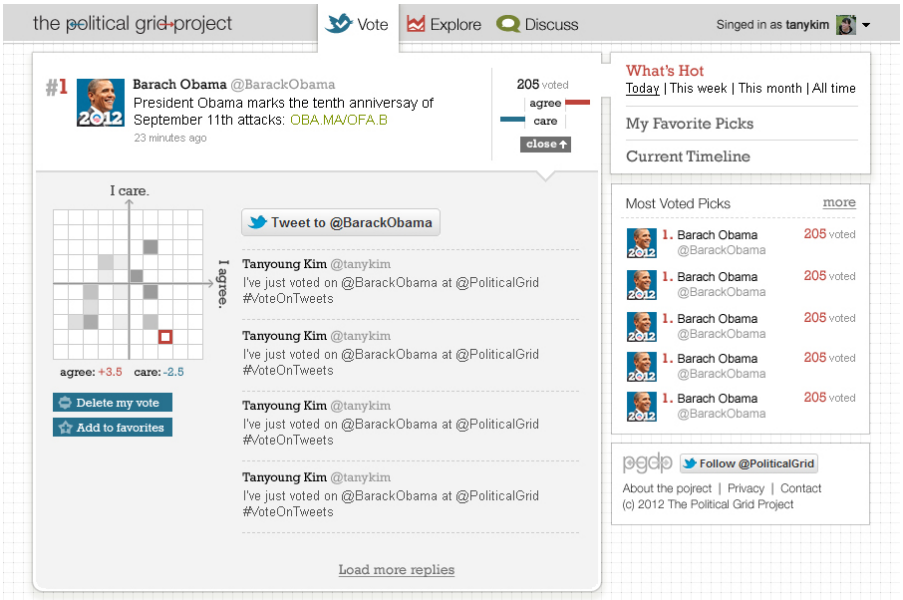
Tweet Timeline



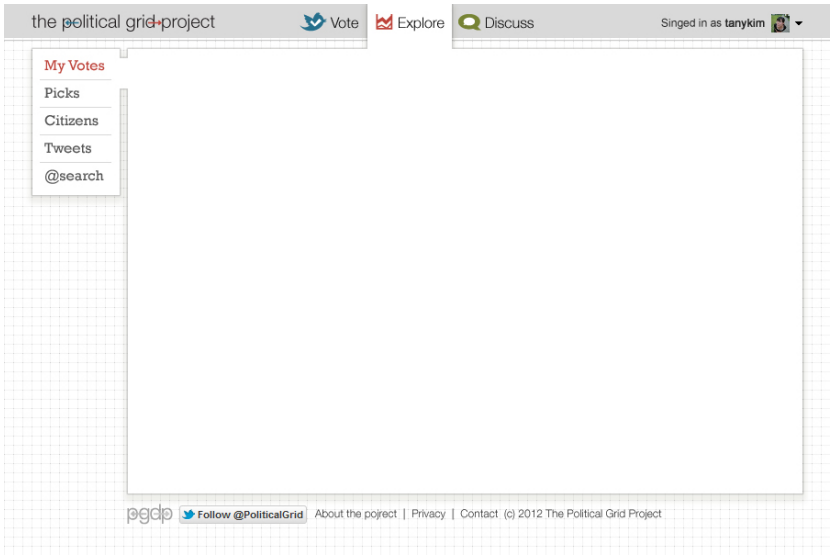
Voting Panel



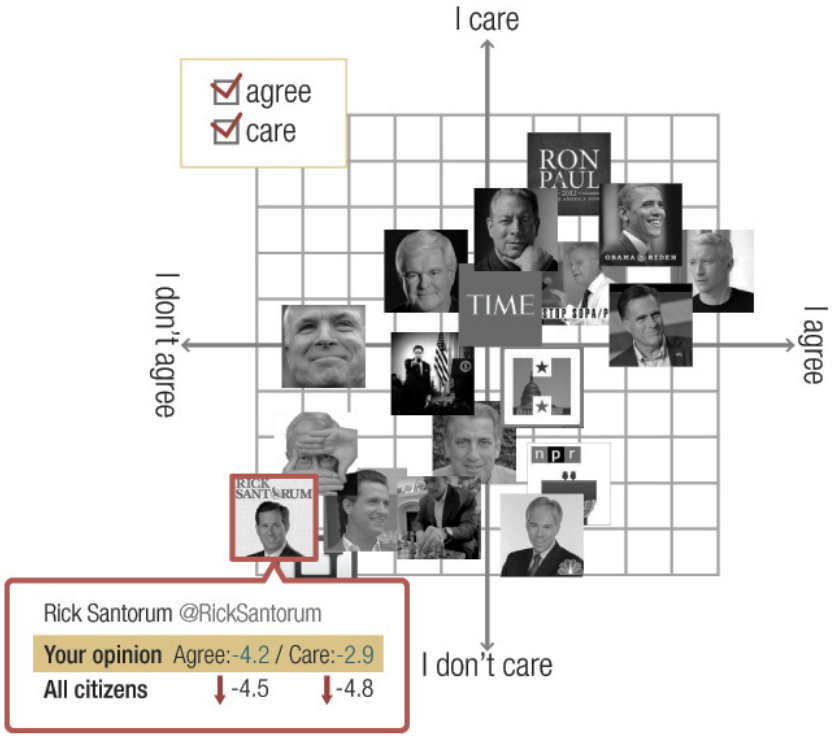
Voting Summary



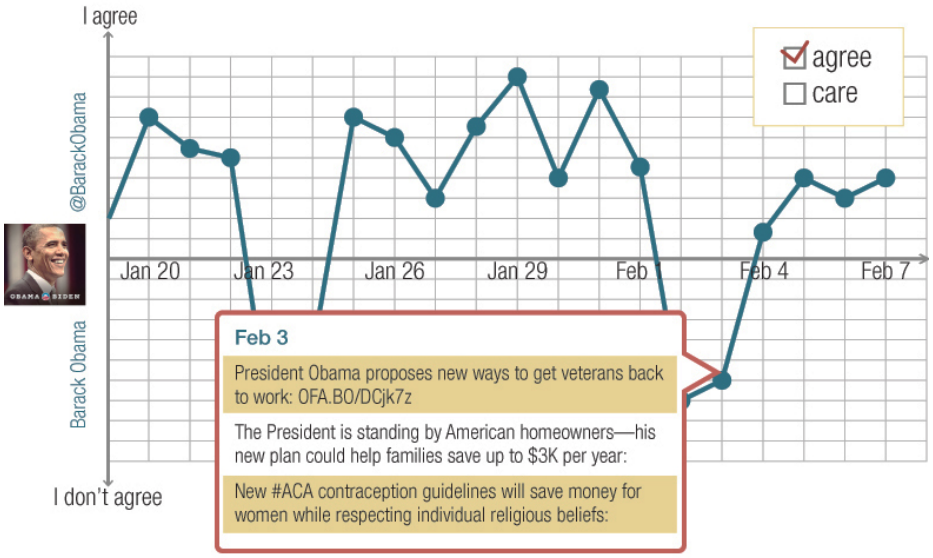
Explore Page



My Voices



Politician' Explore Page



Validation Plan

See Pivotal Tracker test case at https://www.pivotaltracker.com/projects/457151.

API Documentation

See http://corsair.dyndns-free.com/~cs4911/politicalgrid_v2/docs/annotated.html