crowd.ed Documentation

Release 0.0

The crowd.ed Project

August 24, 2014
Contents

1 What is crowd.ed? 3
  1.1 Our Mission .......................................................... 3
  1.2 Our Name ............................................................ 3
  1.3 The Problem .......................................................... 3
  1.4 What Can We Do About It? ........................................ 3
  1.5 What Can You Do ..................................................... 4
  1.6 Similar Projects ....................................................... 4

2 Phases 7
  2.1 Phase 1. Platform Development ........................................ 7
  2.2 Phase 2. Deployment ................................................... 7
  2.3 Phase 3. Data Collection .............................................. 7
  2.4 Phase 4. Data Mining/Processing ..................................... 8

3 Contributing 9

4 Developers 11
  4.1 Getting Started ....................................................... 11

5 Indices and tables 13
What is crowd.ed?

1.1 Our Mission

Leverage the contributions of the masses to empower ordinary citizens in the struggle to keep their elected officials accountable to those whom they represent.

1.2 Our Name

At the most basic level, crowd.ed comes from the words crowd and (ed)ucation, but it is far more than just a combination of words. crowd.ed is an idea. The idea that ordinary citizens can donate just a little bit of their time, energy, skills, and/or money towards helping to educate and inform others about the issues that impact us the most.

1.3 The Problem

This project was started out of frustration with the Nation’s current political situation. Congressional approval ratings are the lowest they’ve been in 40 years (a mere 16%), yet we continue to see reelection rates above 80%, according to the Center for Responsive Politics. At what point do we begin to take responsibility for continually sending the same people back to represent us on the Hill?

To be fair, this is likely due to the major barrier to entry that has developed over the years. In the 2012 election cycle, the average cost for a challenger to even have a chance to win out over the incumbent in a House election was $2.4 million dollars, a 2300% increase since 1974. If a challenger can’t raise the money, the odds of victory are considerably lower, with odds of 293:2 if your campaign furnishes less than $1 million.

With numbers like these it’s no wonder that so many have lost hope in the system, and distanced themselves from politics wherever possible. For what hope does a single voice have of being heard in a room filled with attack ads, big business, and dirty money?

1.4 What Can We Do About It?

This is where crowd.ed comes in. A network of connected and informed citizens, actively participating in the collection and dissemination of information. Thomas Jefferson once said in a letter to Richard Price:

“Whenever the people are well informed, they can be trusted with their own government; that whenever things get so far wrong as to attract their notice, they may be relied on to set them to rights.”

—Thomas Jefferson (1789)

It is no longer sufficient to sit idly by and watch our elected representatives make a folly of their entrusted positions. “Every government degenerates when trusted to the rulers of the people alone.”

If we are to have any hope of steering this country back onto the course intended by our Founding Fathers, it is necessary that we act.

This is a project driven by the people, for the people.

1.5 What Can You Do

Simply put, you can get involved! crowd.ed is a project driven by the people, for the people, and will go nowhere without your contributions.

We encourage contributions from any and all who are interested and able. We only ask that you take on a professional and respectful attitude while working with us. We strive to provide factual information that is unbiased in nature, that can be used by all people to make educated and informed decisions about where they place their trust.

“Few will have the greatness to bend history itself, but each of us can work to change a small portion of events. It is from numberless diverse acts of courage and belief that human history is shaped. Each time a man stands up for an ideal, or acts to improve the lot of others, or strikes out against injustice, he sends forth a tiny ripple of hope, and crossing each other from a million different centers of energy and daring those ripples build a current which can sweep down the mightiest walls of oppression and resistance.”

—Robert F. Kennedy

“It is the greatest of all mistakes to do nothing because you can only do little - do what you can.”

—Sydney Smith

Feeling inspired? Find out how you can Contribute

1.6 Similar Projects

There are several existing projects that are similar in nature to crowd.ed. Each project is quite valuable in its own right, however we felt that each of the projects was missing at least one small component that is vital for enabling the changes we need.

1.6.1 The Vote Smart Project

This is an excellent project with many fantastic features, and a wealth of useful information. The information on the site is collected by members of the organization during internships. University students along with members volunteer to contribute to the cause for a pre-determined amount of time during the internship. The organization has a full-time staff that works out of offices in the beautiful Montana Rockies.

• Pros
  – Data is entered by people, leading to an easy-to-read analysis of most issues
  – Several great tools to help voters determine their stance on pivotal issues
  – Volunteers and interns get to work at a beautiful ranch in the Rockies

---

2 Thomas Jefferson: Notes on Virginia Q.XIV, 1782. ME 2:207
• Cons
  – In order to contribute (anything but money), you must take a minimum of two weeks to travel to the
    Montana Rockies. While this would be a wonderfully beautiful experience, who wants to spend their
    hard-earned vacation reading through congressional voting records?
  – Closed source software platform, and data is copyrighted
  – The organization seems to have difficulty in keeping up with the work, especially during non-summer
    seasons.

1.6.2 Open Congress

Open Congress is an open software platform developed by the Sunlight Foundation. The platform uses automated
scraping software to collect information from numerous sources, and prevents them for display on the website. The
organization also supports several Application Programming Interfaces (APIs), to allow the data they collect to be
used by other developers for other projects. The main drawback here is that the project seems to be no longer under
active development, and is limited in scope as to the information it provides. The platform does not allow for users to
enter data directly, which leaves us only with limited analysis and summaries of many issues. We hope to be able to
leverage much of the data already collected by the Open Congress project in the development of crowd.ed.

• Pros
  – Tons of good data in one location
  – Provides users the ability support comment on and show support for candidates/issues

• Cons
  – User contributions to the site are limited to comments and showing support. They cannot contributed to
    the site content directly.
  – The site/data collected generally lacks a human element, making it somewhat difficult to parse through the
    information at times.
The project will progress through several phases; each of which requiring contributions from various persons, with a wide range of skills.

2.1 Phase 1. Platform Development

Primary contributors during this phase will be software developers. During this phase we will work to develop the software platform that will serve as the foundation for the remainder of the project.

**Deliverables:**

1. A software platform that supports the needs of the following phases.

2.2 Phase 2. Deployment

During this phase, we will deploy the software platform, making it available to the general public. This phase will likely require a substantial financial investment to provide the server processing power and up-time required by the platform. Again, software developers will largely be contributing during this phase, however, we may (and probably should) reach out to the general public for financial support.

It is vital that we get the word out about the project during this phase. Possible avenues are social media, and other crowd-sourced information sharing sites (and possibly investment sites) such as kickstarter.

**Deliverables:**

1. A publicly accessible platform that users can access and interact with.

2. Enough funding to cover hosting, and any other fees/costs associated w/ running a website.

2.3 Phase 3. Data Collection

Contributions during this phase will largely come from the general public. It is during this phase that we are collecting information on various public representatives. At the very least, we should be tracking congressional voting records, bill contents/summaries, etc, all of which will be coming from data input from users.

During this phase, software developers continue to develop the platform, adding new features and addressing issues as they arise.

**Deliverables:**
1. A collection of information regarding elected officials, their voting histories, and any other information that is deemed useful to the end user.

### 2.4 Phase 4. Data Mining/Processing

Once a substantial amount of data has been collected, we can begin using data mining techniques to further analyze the data. The goal here is to provide citizens with a more interactive experience, and more valuable information.

**Deliverables:**

1. The platform can provide the user with candidate suggestions, potential outcomes of votes, etc.
Since crowd.ed is a crowd-powered project, things only get done when people pitch in. We are always in need of support from software developers to build and expand our software platform. If you are interested in joining the development effort, check out the **Developers Page**.

Unfortunately, since we don’t have the platform completed yet, we’re not quite ready to accept major contributions from *non-technical* types. However, don’t be discouraged! Once we have the platform up and running, there will be plenty of work to go around. If you would like to stay informed of our progress, you can join our mailing list, and we’ll let you know when we’re ready for some more help!

Finally, if you don’t have the time to donate but still would like to contribute to the cause, we do accept financial donations to help keep the lights on. Since all of our work comes from volunteers, you can be rest assured that your money is always going to supporting the project.
4.1 Getting Started

crowd.ed is hosted on Github, but all of our development is done via Gerrit. Gerrit is a code review tool developed by Google for the Android Operating System Project, and offers a much more powerful code control mechanism than Github. Gerrit is used by many open projects, and has already been well documented, so we won’t get into that here. If you’d like to learn more about Gerrit, check out this site.

4.1.1 Project Setup

To get going on the code, you’ll first need to set up an account at gerrithub.io. The site uses your Github credentials, so if you don’t already have an account with them, you’ll need to sign up there as well. During the sign-up process, you’ll be asked if you want to import your Github repositories. Importing repositories is completely optional, but doing so will let you use Gerrit with your other projects.

Once you have your gerrithub account set up, you can clone the project by executing the two-part command below (substituting \textit{<gerrithub username>} with your username):

\begin{verbatim}
git clone ssh://<gerrithub username>@review.gerrithub.io:29418/montaguk/crowd.ed && scp -p -P 29418 <gerrithub username>@review.gerrithub.io:hooks/commit-msg crowd.ed/.git/hooks/
\end{verbatim}

The first command will clone the crowd.ed source code, and the second will set up a git-commit hook that generates a unique \textit{Change Id} for each commit. This ID is used by Gerrit to track a specific change across cherry-picks and rebases. For more details on this mechanism, see the Gerrit Documentation.

4.1.2 Making a Change

Once you have the repo set up, you’re ready to start making changes! Go ahead and modify the code to your hearts desire. Once you are satisfied with your changes, commit them to your local repository as you normally would:

\begin{verbatim}
git commit -a
\end{verbatim}

Once that is done, you’ll need to submit your changes to Gerrit for review. To do this, run the following command (substituting \textit{<branch name>} with the target branch):

\begin{verbatim}
git push origin HEAD:refs/for/<branch name>
\end{verbatim}

In most instances, the target branch will be \textbf{master}, but this will likely depend on the nature of the change.

Now you just sit tight and wait for someone to review your change on gerrithub. You can view the status of your pending review by selecting your review from the \textit{Outgoing reviews} list here. While your waiting for someone to review your change, why not make some comments on another change!
4.1.3 Reviewing Changes

Every registered user has the ability to review a change. You may comment on the change and give it either a +1 or a -1 vote. Only project integrators have the ability to give a +2 or -2 vote. In order for a change to make it to the target branch, it must receive at least one +2 review. If a change receives a -2 review, that change is cannot ever be merged into the target branch. See this page for more details on the Gerrit scoring system. ..note:: gerrithub only supports the Review and Verified labels.

4.1.4 Verifying Changes

For a change to be submitted to the target branch, it must first be marked as verified. Normally, pushing to HEAD:refs/for/* would cause a CI server to build the proposed changes automatically, but gerrithub does not currently support this. As the project grows we would like to move towards a fully automated continuous integration implementation, but for now changes must be verified manually.

4.1.5 Submitting Changes

Once your change has received a “Reviewed: +2” and a “Verified: +1”, it will then be submitted to the target branch. When requested, a submit will cause Gerrit to merge the change into the target branch, and sync the change back to the repo on Github. At this point, Github triggers a continuous integration build on drone.io.

4.1.6 Continuous Integration

crowd.ed uses drone.io CI builds to monitor the state of each branch. This helps us to catch any issues/bugs with the code as soon as they are introduced, and ideally prevent them from making their way into the code in the first place. Since gerrithub is currently limited in its implementation, our current CI situation is not ideal. Perhaps one day we’ll have the resources to manage our own Gerrit server, but for now we’ll just have to make due.

4.1.7 Documentation

The majority of crowd.ed is written in python, and uses Sphinx to document the code. The docs are hosted on ReadTheDocs and are rebuilt each time master is updated. When making changes to the code, we request that you also make the necessary changes to the documentation, as one of our goals is to keep the code-base as easy to maintain as possible. Sphinx tracks what percentage the code is documented, and we’d like to see that number stay as high as possible.

4.1.8 Testing

In order to fully utilize the benefits of continuous integration, we use a couple different testing frameworks. The web code is tested using Django’s built in testing framework, while we use the python nosetest package to test non-web code (scrapers, API interfaces, etc). If you’re adding new tests, make sure to add the appropriate calls to build.sh, which is the script that drives the CI build.
Indices and tables

• genindex
• modindex
• search