Client: Kati.io

Talanda Williams

CST 499: Capstone

Summer 2018

Table of Contents

Executive Summary

Kati.io, henceforth referred to as Kati, is a lean startup working to deliver a platform for web annotations for journalism. The main features of this platform is two-fold: an extension and a website. The extension allows users to create and manage annotations for a news article or site and provide a ranking of bias and subject tags. The website is the user interface for these annotations as well as providing a searchable database of a community provided pool of reviewed news articles and their corresponding details.

Projects like Hypothesis and Genius already have shown the user interest in annotating the internet. Web annotations are now a W3C standard and have shown that their importance and use in this digital age is needed. This cornerstone of this project is the creation of the user website and frontend, to be integrated with the backend MySQL database and extension component. Due to this complementary nature, features of the project may need other related aspects completed before additional work can be done, providing a potential time conflict. Assuming that does not occur, the deliverable for this project is the front end completion of the user-interactable website.

Project Description

Kati is in the very early stages of creating a product that is not only helpful but crucial in the age of digital journalism. With the wide range of news sites, articles, and (mis)information, Kati is poised to be a central hub for finding, commenting on, and searching for articles. The project consists of two main features: a website and an extension. The extension will be used for interactions with the articles themselves, through a right-click context menu or direct icon selection on the tool/address bar.

The extension will be how annotations and tags to an article are both initially made as well as viewable by other users. The website is how users will search for articles via the subject tags, look at their prior annotations and feedback, along with any additional features introduced during the design phase. There will be a mix of front and back end development, where needed, to solve modular steps of in an agile sprint fashion.

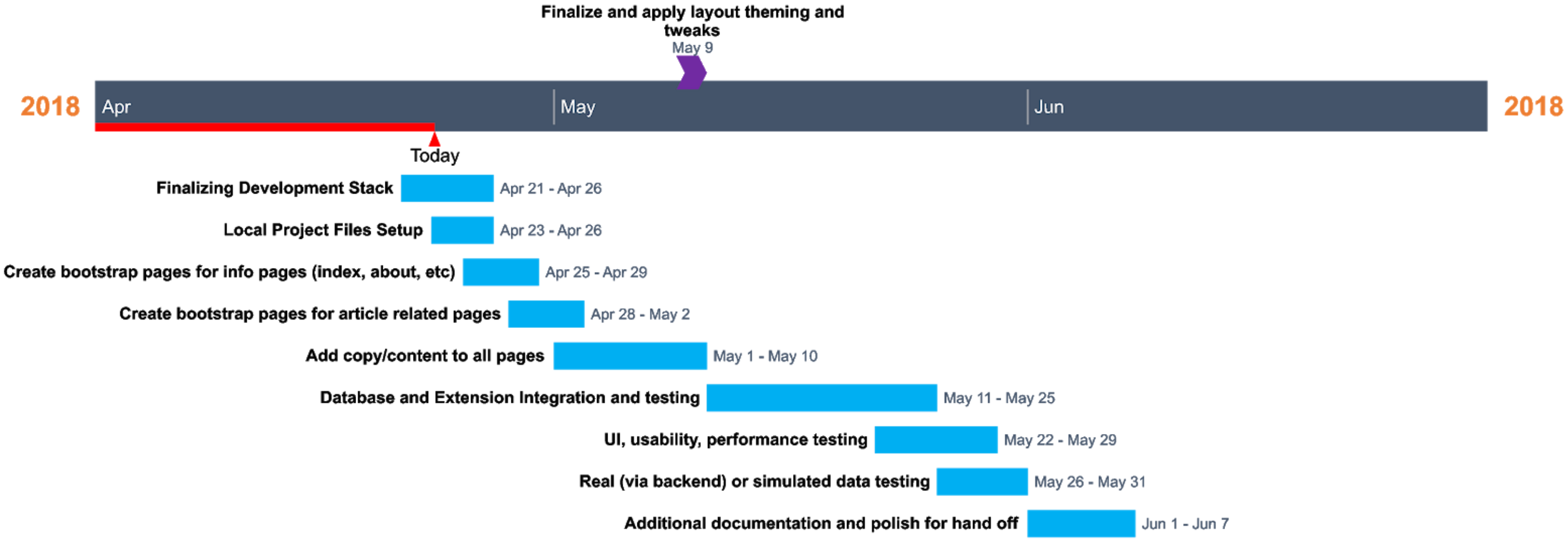
The front end of the user website will be launched on Amazon Web Services and developed primarily using HTML5/CSS3 and JavaScript. Bootstrap 4 was chosen as a front end library for prototyping and development of the user interface. The website will encompass a user login and database integration, not developed in the scope of this project, but the pages themselves will be created. The creation of the extension nor the development of related features or the database is not included in the scope of this project, however, the development status of these aspects will affect the ability to work on aspects of the website which should interact with them.

Project Goals and Scope

The main project goal is the creation of the layout for the website aspect of the Kati interface. This includes the creation of the pages and the structure which will later integrate with the database via API or extension details, to the data itself for each user or article reference. The management of the MySQL database for the user and article data to be stored, as well as the related annotation details for it, is not covered but if time permits may be included in later stages.

Project Objectives and Timeline

- Design and development of a user profile page for showcasing user annotations and five-star rankings of articles.
- Design and development of the general website structure, including user account management, article rating details, search page, and call to action for the extension download.
- Integrate with an analytics tool, such as Google Analytics, to gain data about the usability of the website.
- Interface with the MySQL database using NodeJS to load the relevant data into the website, if the backend is available and time permits.
- Extra: Implement a tag-feature for the overall article to be displayed on a searchable articles directory page.



Client Background

Kati is a discussion platform to verify statements in news articles, using crowd-sourcing to hold journalism accountable and find the most trustworthy articles on any subject. Kati started off as a branch of Hypothesis, a self proclaimed nonprofit on a mission to bring an open conversation over the whole web. Hypothesis is used to hold discussions, read socially, organize your research, and take personal notes. Kati plans to serve the area of journalism to allow users to highlight statements of bias in news stories, apply a rank to the article, and insert subject tags, all aggregated from users who add information about each article.

After additional research and evaluation, Kati decided to not use Hypothesis as a base and to start from scratch on the Kati project without the added complexity of the Hypothesis tool. The client details are still in a discovery phase, with new direction being decided on throughout the project itself.

Problem to Solve & Evidence of Need

The internet is a giant collection of all of human's vast knowledge, and annotating it provides a framework for discussion and enlightenment amongst us all. The need for web annotations is not a new discovery, nor is Kati the first to tackle it. In 1993, web group annotations saw its first start with a server test for the Mosaic browser. The creators had figured that users of the web would want to annotate across websites and tested the possibility. The issue was scale; the server would need to hold an enormous amount of data for all user annotations across the entire internet, and so the project was scrapped. Since 2009, there have been a number of groups working towards web annotations. One of which was the World Wide Web

Consortium (W3C), the international community which develops web standards. The W3C Open Annotation Community Group worked to build a draft of specifications built of the work of the Annotation Ontology and the Open Annotation Collaboration.

In February 2017, the W3C released annotation standards in the form of “recommendations”. This included the Annotation Data Model, the Annotation Vocabulary, and the Annotation Protocol, together providing the foundational material for annotation tools. This is a key milestone toward having built in browser support for annotations in a standard format and native functionality. The model provides a framework where annotations are separate from data and reunited and re-anchored when the relevant document is presented. This puts the data in the control of the reader and removes the need for “reinventing the wheel” by developers wanting to replace the current style of comments needing to be embedded into individual sites. The years of work put into the development of this standard and the ideals behind it highlight the case for annotations as a whole on the web.

In addition, journalism is certainly a key industry where having the ability to use web annotations while viewing various perspective and news sites is important. The pathways to news online are steadily increasing, in 2016 it about 4 in 10 Americans mostly get their news purely online. 72% of Americans in 2016 consumed the news on a mobile device, over half preferring mobile. Combining this with the notion that fewer than 25% of Americans have trust in social media sources for news; people want a way to get information that they trust. A fluid solution is needed combining the news and the prospects of annotations. Sites are unable to do this alone, as even a useful comments section on one site does not correlate to an increase use across other news sites as a cohesive system. Additionally, many sites cannot have another annotation system on top of a current comment or sharing system such as Facebook. A third party is needed to combine these aspects of news collection and sorting, as well as annotations and the sharing of that user feedback and knowledge. With Kati, a user wondering about the legitimacy or bias of a news article would not need to leave the page to learn more and discuss with other users qualms or questions they might have. They would be able to have a dashboard of information that they've either gathered themselves or via the network of users, fully ranked and reviewed.

Community Impact & Ethical Considerations

Articles included in the Kati user profile page will come from a variety of sources, and we need to ensure that we do not cause any legal issues or concerns around this data. The articles themselves will be linked to the original site from the user profile, making sure to avoid any hot linking or iframe issues with embedding data from one site into another site. Additionally, it may show through user provided feedback that certain sites are geared toward a certain perspective; such as biased politically left or right, very biased in other aspects, very liked or unliked, etc through the use of the various user provided rankings and the bias score provided by the system to each article.

Considerations must also be made to ensure that the system does not produce libel as a byproduct or related claim. Future precautions must be taken before the wide adoption of the site, to ensure that spam reviews and other poorly rated responses and annotations do not negatively impact the bias score, ranking, or other aspect of Kati. In terms of the rating and order of the articles on a user profile, this is all populated based on the articles they have already looked at. In the search section, there must be considerations to not seem biased as a platform itself; via particular political leanings or otherwise during this politically charged time where a majority of the articles reviewed may be political in nature.

Current Kati plans involve the platform being used via Chrome, Firefox, and Edge extensions for actual annotations, with other interaction systems made through the website and profile via any browser. This requires a scan of current breakdown of technologies across different areas and the percentage running on each platform above to make Kati as accessible as possible for the community. Additional community considerations may involve ensuring the website for general accessibility standards, potentially to be included in the scope of the creation of the user profile page.

Environmental Scan

Currently the main companies in this field are Hypothesis and Genius in terms of web annotation leaders. Hypothesis as previously explained due to the mission of general web annotations and Genius as a very popular website for lyric explanation and built in annotations, which they have made usable in other websites. Hypothesis is the closest in terms of direct competition, but does not cover the extra niche areas of journalism that Kati targets. The ranking system and users ability to have a centralized dashboard for searching for reviewed articles sets it apart.

Genius has a web annotator project similar to the style of Hypothesis, where it can be used as a Chrome extension, embedded into individual sites, or using a proxy such as genius.it/ before the url name. Their news section, their self-proclaimed ‘flagship’ of the project has not been updated in 2 years and does not appear to have an active base. An additional Historical Survey developed by Hypothesis is included in the Appendix.

Risks & Dependencies

The main resources needed relate to feedback on direction and progress from the Kati team. There is only the founder and one engineer who make up the start up, so feedback may be easily obtainable, but may also be fragmented due to other items. At the beginning stage, there is a need for the server to host the website, the domain it will be accessible by, and framework decisions for the start. As of now, there is an AWS instance set up and a empty Kati.io domain landing page, but work will mainly be on localhost until those can be properly configured for access.

The success of the website/front-end portion of the Kati project is heavily dependent on the ability for the extension to exist, provide articles, and save them to the database so that the user page has data to populate the site with. Since the majority of the database work is for the articles, user notes, and related data- the decision of the stack and architecture must be decided before the implementation of anything secondary such as user authentication and profile saving. The backend structure is not currently set at the time of this proposal, but it is expected to consist of using NodeJS -> API -> MySQL database. Despite this due dependency, this proposal is laid out such that as long as the front end web portion of the project is prepared for those later details to be added, the basic template is created, and copy can be easily added, that the project is complete without these extras for the company itself to roll out the solution.

Usability Testing & Evaluation

Using the tools included in the Appendix, the website portion of Kati will be evaluated using modern web application standards of performance. More details of direct requirements of what constitutes each test will be constructed in May pending the decision on the full stack Kati will use. For manual testing, I plan to use the Guru99 checklist consisting of testing: usability, functional, compatibility, database, security, and performance, as available. Usability is basic layout and user design and interactions. Using a tool like HotJar or MouseStats free plan for analytics, data can be brought in about user using the site via heat maps, clicks, timings, recordings, and etc built into the JavaScript code.

Most of the checklist will be testing throughout the project, not just at the end. This will ensure that there is a comprehensive check of all aspects including links, form fields, entries, etc on various devices and screens including Windows, Mac, and Android phone. This will help also cover compatibility testing and in various browsers across those devices. Functionality testing may depend heavily on the completion of the backend aspects, but will also include any JavaScript validation whether client or server side via NodeJS, text fields and making sure nothing is bypassable, etc for any sort of calculations.

There have not been clear evaluation goals set from the client, but I intent to follow the Google Developers guidelines for a clean loading and site experience. This involves avoid landing page redirects, enabling compression, improving server response time, leverage browser caching, minifying resources, optimizing images and CSS delivery, prioritizing visible content, and removing render-blocking JavaScript. According to MachMetrics Speed blog which compiled different industry research, 53% of users will abandon a page if it takes longer than 3 seconds to load. Therefore, the front end profile will have a goal pagespeed rendering of less than 2 seconds, as asynchronous calls could still be loading for additional seconds after the page has appeared to the user. The average page size in the US is 1.88MB, which correlates to loading speed and bloat time for those with slower internet connections. The goal for the front end profile will be less than 1Mb. The average number of resources, numbers of files the browser has to download to render the page, in the US is 115.6 files, the front end profile will aim for 50-80 resources. Lastly, the average server delay in terms of time to first byte, which plays an important role in Search Engine Optimization, on average in the United States is about 2.11 seconds and the front end profile will have a goal of less than 1.3 seconds. These values will be tested using WebPageTest, Pingdom, and GTmetrix.

Summary of Goals with Numeric Values:

Page Load	< 2 seconds
Page Size	< 1 megabyte
Number of Resources	< 80 resources
Server Delay	< 1.3 seconds

An evaluation report will be included on the results from the three testing sites during the end of the project. If real user data has also been gained by this time, an analytics report including Google Analytics and MouseStat or HotJar will show real and test session data. The website, once complete, will be handed off as a zip file with all accompanying code, images, files, and data obtained during this project.

References

Andreessen, M. (1993, May 31). Group annotation server guinea pigs? Retrieved from http://1997.webhistory.org/www.lists/www-talk.1993q2/0416.html

D. (2017, February 24). Annotation Is Now a Web Standard. Retrieved from https://web.hypothes.is/blog/annotation-is-now-a-web-standard/

W. (2017, February 23). Embedding Web Annotations in HTML (T. Cole, Ed.). Retrieved from https://www.w3.org/TR/annotation-html/

Fischer, S. (2017, March 21). Social media sources for news, ranked by user trust. Retrieved from https://www.axios.com/social-media-sources-for-news-ranked-by-user-trust-1513301079-91a3154c-671f-416d-a796-5d80f9024f9d.html

Gooding, S., Aldrich, C., Schinkel, M., & McCan, D. (2017, March 03). Web Annotations are Now a W3C Standard, Paving the Way for Decentralized Annotation Infrastructure. Retrieved from https://wptavern.com/web-annotations-are-now-a-w3c-standard-paving-the-way-for-decentralized-annotation-infrastructure

Home. (n.d.). Retrieved from https://web.hypothes.is/

Mitchell, A., Gottfried, J., Barthel, M., & Shearer, E. (2016, July 07). 1. Pathways to news. Retrieved from http://www.journalism.org/2016/07/07/pathways-to-news/

Sanderson, R., & Cole, T. (2017, February 23). Making it Easier to Share Annotations on the Web. Retrieved from http://w3c.github.io/web-annotation/admin/AnnoRecsPublished-BlogEntry.html

Appendix

List of tools which will be used to benchmark the performance and accessibility of the website portion of Kati.io. It will also be integrated into Google Analytics for usability testing and interaction.

1. W3C Website Markup Validation Service: https://validator.w3.org/
2. Google PageSpeed Insights: https://developers.google.com/speed/pagespeed/insights/
3. Can I Use… and HTML5test, checking for browser specific abilities: https://caniuse.com/#comparison and https://html5test.com/results/desktop.html
4. Checklist: https://www.guru99.com/complete-web-application-testing-checklist.html
5. PageSpeed Insight Rules: https://developers.google.com/speed/docs/insights/rules

Further reading and details about related tooling and systems for creation and optimization of the project as well as additional insights on web annotation advancement.

1. Cloudflare content delivery network: https://www.cloudflare.com/
2. Server side Javascript: https://nodejs.org/en/
3. W3School's Browser Statistics: https://www.w3schools.com/browsers/default.asp
4. Credibility Coalition: https://credibilitycoalition.org/
5. Hypothesis Historical Survey: Google Drive viewable sheet
6. MachMetrics Speed survey: https://www.machmetrics.com/speed-blog/average-page-load-times-websites-2018/