

University of Arkansas – CSCE Department

Capstone I – Final Proposal – Fall 2020

AltruSight

Rafael Toche Pizano, Ben Guthrie, Andrew He, Tanner Edwards, Seth Piepergerdes

Abstract

Charitable giving is a selfless act that can impact the world in different ways, from environmental conservation to human welfare, many individuals donate in hopes of having a positive impact in the world. In 2017 donations to charitable causes were estimated to be \$410.02 billion or 2.1% of the US’s GDP [1], but how can donors be sure that their money is being put to good use, and that is in fact being used for charitable causes? The US government has laws and regulations for nonprofit organizations, but it is up to nonprofit organizations to build trust with their donors and donors must trust that their donations are being put to good use [2]. Currently, the US government has no centralized system where donors can track how nonprofits they donate to spend their donations with transparency. Our objective is to create a proof-of-concept website that gives donors a centralized system where they can track how nonprofits use their donations. Ultimately, this would create transparency within nonprofits and give better insight as to how nonprofits operate and whether they live up to their mission. On top of this, we want to streamline the donation process into a new social experience that will ideally increase the frequency and consistency of charitable giving, all while providing donors the tools to analyze their own personal giving as well as the previously mentioned transparency into how their donations are being spent.

1.0 Problem

Donors tend to not track how their donations are spent due to the complexity of the process. Oftentimes, when someone donates to a nonprofit, the donor will never find out how the money was spent. In the US many nonprofits collect money but do not use the money for the intended purposes. A list put together by the Tampa Bay Times and The Center for Investigative Reporting reported 50 “nonprofits” that collected up \$1.3 Billion from 2008 -2018 in donations but sent \$970 million to the collectors [3]. This type of behavior discourages the population from donating to nonprofits and ultimately hurts legitimate nonprofits and reduces the meaningful impact of legitimate nonprofits.

Nonprofits are also not obligated to disclose who donates to their organization [4], and although this is normally not a problem, not disclosing big donors makes it easy for nonprofits to be used as money laundering intermediates. Most nonprofits have their own donation sites, and although this in itself is not an issue, it makes it hard for potential donors to verify the authenticity and transparency of the nonprofit. Although some thorough research would help establish the authenticity of a nonprofit, some donors simply want to donate without the hassle of having to do in depth research. A lack of transparency on behalf of nonprofits makes it easier for them to get away with illegal activities such as fraud or money laundering. An example of this is the money laundering scandal involving the Fédération Internationale de Football Association (FIFA) where an Israeli bank and a Swiss bank reached a resolution with U.S. prosecutors to pay \$30 million for conspiring to launder more than \$20 million. [7]. Officially, FIFA is registered as a non-profit organization that administers football events such as the FIFA World Cup and Copa America [8].

As a result of not having a centralized system for nonprofits, opinions and ratings for nonprofits are scattered throughout the internet. Different websites may use different rating methodologies which makes it difficult to assess whether a nonprofit aligns with a donor's views and values. In addition to this, donors would then have to research if the website rating nonprofits is reputable, as ratings from a non-reputable website could be inflating a fraudulent nonprofit's ratings to make it seem legitimate. Approximately one in three donors research nonprofits before donating [5]. In other words, approximately 66% of people who donate do not look into the charities to which they want to donate. A centralized government regulated system, for example, would allow users to go to one specific place where they can find information about nonprofits and make an informed decision before donating.

2.0 Objective

Our objective is to create a proof-of-concept website that serves as a centralized system where all nonprofits must register so that donors can track how their donations are being used. Our objective is to create a proof-of-concept website that pulls information on multiple non-profits with the help of an API. These non-profits will be displayed on our web-application and they will be separated into two categories, verified and not verified. Non-verified non-profits will simply have their information displayed on the web-application, however, they will not be considered 'legitimate'. Theoretically, in order for a non-profit to be labeled as verified, a non-profit representative would have to request for the non-profit to be registered as legitimate. Ideally, the government would then run a rigorous background check on the non-profit in order to ensure its legitimacy, and if this is proven to be true, the non-profit would then be labeled as 'verified'. The web-application would then let users donate to 'verified' non-profits and the non-profit representative would have more control as to the type of information that the non-profit posts on their site.

This would provide donors with a centralized system that they can use for transparency, as well as for analytics. Creating a centralized system for nonprofits would make it easier for donors to choose the right nonprofit to donate to as the amount of research they would have to conduct would be reduced. The website could also have social media like functionality in that users can comment, share, and like nonprofit organizations that appeal to them the most. This would provide a layer of transparency as users would have one place where they can comment on the nonprofits' use of funds and whether the nonprofit is staying true to its mission. This website could also provide users with analytics such as providing reports or graphs that help visualize how a nonprofit is distributing its funds.

Since the goal of this website would be to provide more transparency about how nonprofits spend their funds, our website would provide information such as who is giving to nonprofits and potentially how much (showing how much a user donated could be an option that the user controls). Doing so would reduce illegal behavior and make money laundering more difficult as a centralized system for nonprofits would make it easier to track the donations going into a nonprofit.

Ultimately, our objective is to create a system that makes it easier to hold nonprofits accountable and makes it easier for donors to see how their donations are being put to use. In theory, all nonprofits would have to register to our system as it would be government regulated, and nonprofits would have to undergo a rigorous background check before being registered to our website. This would provide donors with a sense of security as no fraudulent organizations would be registered to our website and every nonprofit would follow a general format for providing useful information such as ending bank balances, balance sheets, or a generalization of how money is to be spent.

Layered on top of our goal to provide donor transparency and analytics into non-profit usage of donations is also the functionality to make the act of donation a public and shareable event. Humans, at our core, are social creatures, so other human beings tend to have an influence on our personal decisions. We want to leverage this into a live feed of public donations, complete with customized captions for the donations, which will bring a new level of information to a donor's fingertips, as well as ideally encouraging people to donate more. We also want to provide the ability to integrate personal goals and analytics for an individual's personal donations, similar to the functionality of many budgeting/spending tracker applications.

3.0 Background

3.1 Key Concepts

We plan to use the Angular framework for our web application. Angular is a framework that allows us to build single page web applications using HTML and TypeScript. TypeScript is not

required to use Angular, but the advantages of TypeScript make Angular a really nice framework to use. TypeScript allows for static typing which will make debugging less of a headache as opposed to trying to debug plain JavaScript. Static type checking will help us catch more errors up front, as opposed to us finding unexpected behavior in our application when testing, and it also enforces clean code in terms of having to plan for what type of information each variable will truly represent. Angular's framework allows for the user interface to consist of independent components. These components control views with the use of HTML templates. Angular components are grouped into modules. These modules help define what components can interact with other components and even other modules in the application.

Alongside Angular, we plan to use Heroku, PostgreSQL, Node.js and Express. Heroku has built in support for PostgreSQL, which is one of the main reasons we are choosing to use it for our databases. We could have used Heroku along with something like MongoDB as opposed to Heroku Postgres. Heroku Postgres is a database-as-a-service provided by Heroku. Our web application will take advantage of Postgres' relational database model. We also wanted to utilize Postgres over a NoSQL alternative (like MongoDB, which is common in the MEAN stack - Mongo, Express, Angular, NodeJS) because the more structured schema will make it easier for us to both store and query data when needed. PostgreSQL also comes with very powerful features such as a very intuitive schema system, strong concurrency support, and even support for JSON types among the standard SQL types. All in all, it is a very flexible, powerful, and robust database that we believe would be best utilized for our project needs.

We need Node.js with an Express server to help build and continuously serve our web application to the public. We will only need to make use of the Express Server on the github branch that we want to host the production environment. Express will help know what data to serve to users depending on the URL path the user is currently visiting in our web application. For testing we don't need the Express Server. Express seems to be the best and most used web application framework for Node.js, and we chose this tech stack because it will have a lot of documentation and support online during the development cycle. We could have gone with something like Maven and Spring Boot but we felt that since more of our team had experience with Node.js, that would be the best option.

3.2 Related Work

CharityWatch is an organization that sets out to "maximize the effectiveness of every dollar contributed to charity." [6] This website allows for users to retrieve info about a charity of their choosing. A charity's page will have metrics including a "program percentage," and "cost to raise \$100." The program percentage of a charity is the amount spent on charitable programs relative to overhead costs. The cost to raise \$100 is how much a charity spends to get \$100 in donations. A higher program percentage with a lower cost to raise \$100 are signs that point towards a good charity. These two metrics along with many other factors allow for CharityWatch

to give each charity a grade on a scale of A+ to F. The page has lots of other information including the tax status of the charity, a list of tax forms that were analyzed by CharityWatch, and information showing how transparent said charity is. CharityWatch is a hub for information about charities, but their website does not have a social aspect, or anyway to actually donate money to charities through their site. This is where our website will be different from CharityWatch. For one, we want users to be able to leave comments on a charity's page. Ideally this will allow for all users to have more unique and informed opinions on charities. With our website users won't have only metrics and a grade to form opinions on, but the input of other users.

Another key thing missing from CharityWatch is the ability for users to donate to a charity directly through the website. When we came up with this idea to be able to donate through our website, we thought of Venmo. Our website doesn't relate to Venmo in that we want users to be able to send each other money, but it does relate to the social aspect of venmo, where users can see payments made by other users is what we think would be a good fit for our website. We would not show amounts donated, but we think that users should be able to see a feed of donations made to charities through the app by other users. Ideally users would be able to donate privately and not share to the public feed if they so desired. This public feed would allow users to see who their friends and others are deciding to donate to, and might be another way to help users take that final step in deciding to donate to a cause they deem worthy.

Our web application won't actually facilitate real payments, as we don't think we have the skills to securely implement this quite yet, but we can mock the payments, and still show a feed of "donations" on our site. It would look to the user as if they sent a payment, and we will store relevant payment information in our databases to help create the donation feed and display other relevant information to users about their donation history through our web application. If we ended up wanting to actually try to implement payment processing on our website, Stripe might be a good option, but for this project actually processing payments is not a requirement.

4.0 Design

4.1 Requirements and/or Use Cases and/or Design Goals

- Support a myriad of nonprofits and automatically populate a list of nonprofits from an API, examples include [Charity Navigator](#) API.
- Users should be able to register an account on AltruSight's website.
- Users should be able to login and logout of AltruSight with ease.
- Users should be able to search for, and view individual nonprofits.
- Users should be able to view the donation activity made public by other users.
- Users should be able to make (mocked) donation payments.
- Users should be able to add nonprofits to a list of favorites for easy access

- Users should be shown accurate and relevant information regarding individual nonprofits
- Users should be shown accurate, fair, and transparent ratings of nonprofits

4.2 High Level Architecture

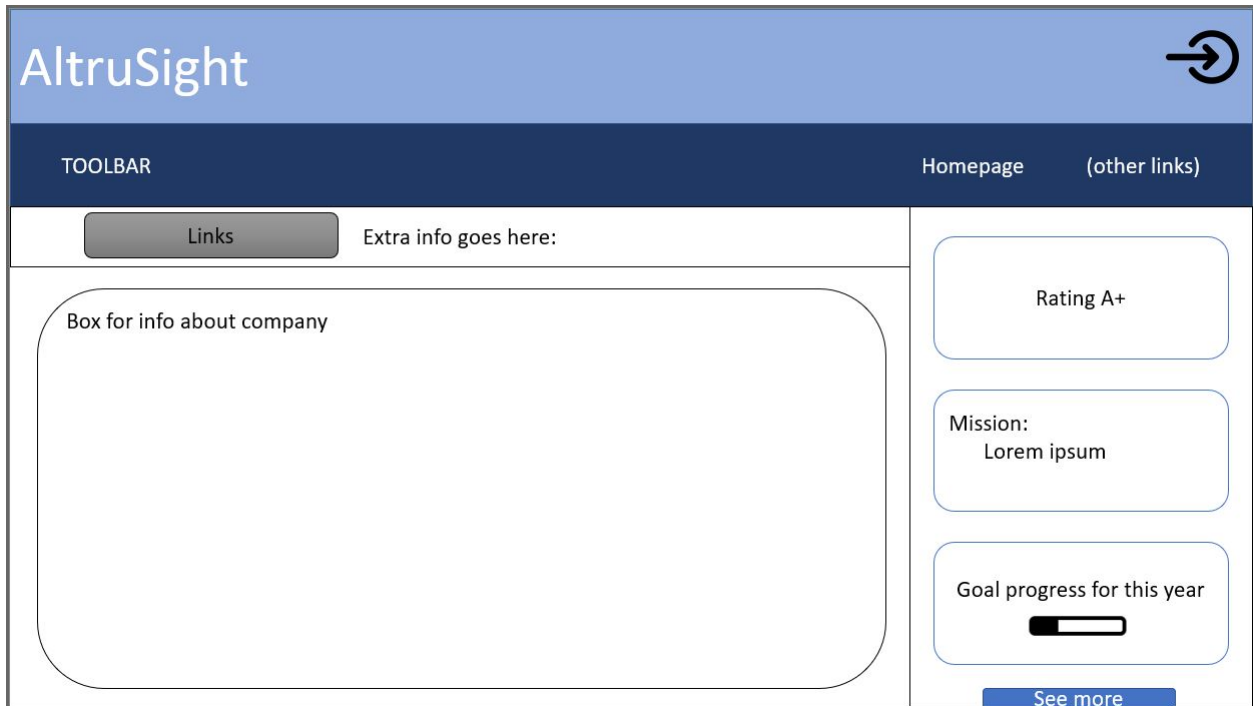
For this project there are no hardware components required outside of a machine for each group member to work on. We plan to use Heroku, NodeJS / Express, and Angular to make up our backend and framework. The hierarchy of web pages could be thought of as Facebook meets Yelp in a way. The home page will be different depending on if the user is logged in or not, and each nonprofit/charity will have their own page that displays their ratings, mission statement, among other things. An individual user's page (not a charity page) will look very similar to a homepage one might expect to find on any social media platform, whereas the charity pages will favor the look of a yelp page.

For the backend server we plan on using PostgreSQL. There are thoughts about adding a link through the site where users can donate to the various charities and their donations will show up on their page. To implement this we must first figure out how to have secure payments that can connect to the various charity payment locations, at the moment PayPal or Venmo are the most promising. If this idea does not come to fruition though, it would always be simple to add donation links to the charity's website.

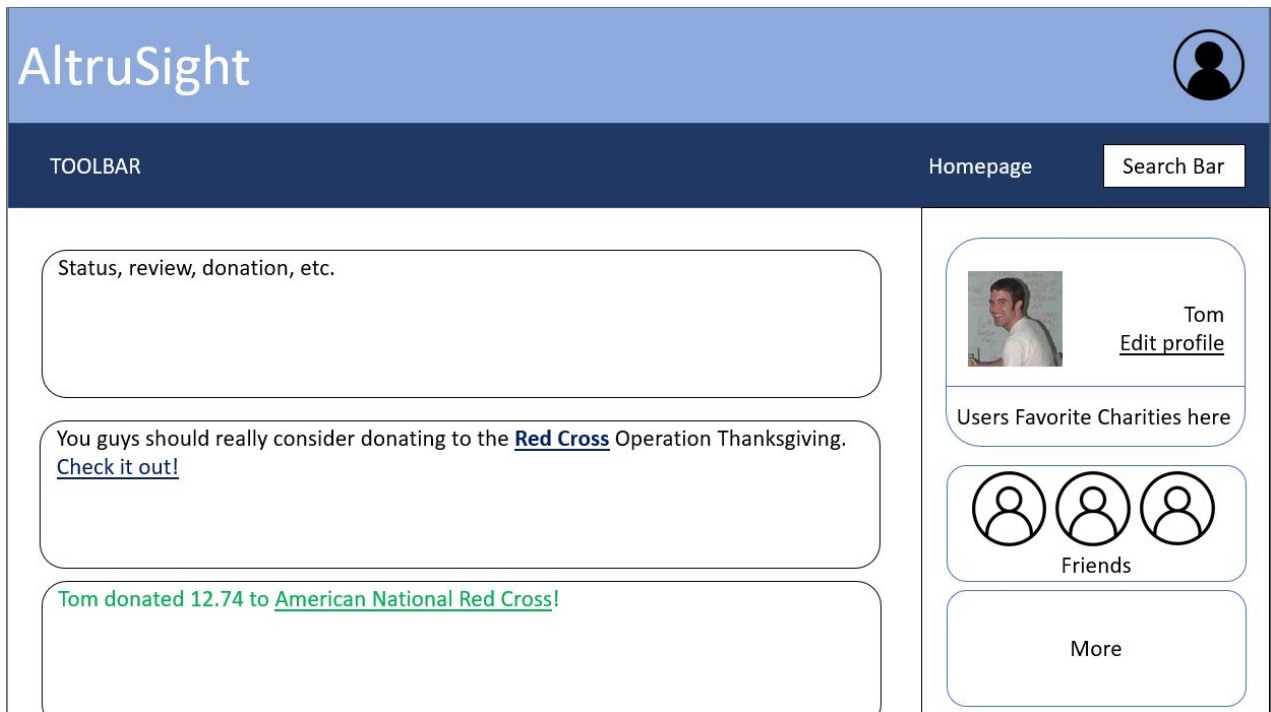
- Below are two examples of what the site could look like depending on if the user is logged into their account or not.

AltruSight

Page accessed when looking at a specific non-profit



Below is a mockup of what a logged in user might see



4.3 Risks

Risk	Risk Reduction
<p>Charity Integrity: There is a risk that some Charities will not be honest in reporting accurate numbers, including amount of donors, total donations, etc.</p>	<p>In order to reduce this risk, the team will ensure that as the API adds nonprofits to the application, the ratings for each charity will be accurate. The ratings will explicitly tell the user how reliable or honest the charity is. Moreover, any accounts of charity fraud will be listed on the page.</p>
<p>Monitoring the comments: Since comments will be allowed to be posted on the charity’s page, they should be monitored.</p>	<p>The team will implement a profanity filter that disallows comments with extreme profanity. Moreover, to reduce bot activity as well as rude comments, email authentication will be required to comment.</p>
<p>Security Risks: Since the project has the potential involvement of personal data, security measures must be taken to ensure privacy and protection.</p>	<p>Because of the proposed centralized donation system, private user data would need to be stored. Security measures will be taken in order to ensure user privacy such as data encryption. Furthermore, weak authentication and session management that leads to sensitive data leaks will be mitigated by focusing on user management.</p>

4.4 Tasks - We will execute the following tasks:

Ramp Up:

- The team should become familiarized with the tools used in our tech stack, mainly Angular which will be used as our front-end, and our back-end which will be composed of a NodeJS server and our cloud database utilizing Heroku PostgreSQL.
- The team should become familiarized with the overall design of our project and understand the use case as well as the features that will be implemented.
- The team should understand the general requirements that nonprofits have to fulfill in terms of government documentation and spending, and develop a basic idea on how to evaluate nonprofits based on the information that is publicly available.
- Locate an API with nonprofit information listed that will fit our use case

Tasks Relevant to the Design of the Front End

AltruSight

- Design and implement our NodeJS backend API and host on Heroku
- Design and implement the home page. This includes:
 - Designing a logo
 - Designing a splash page or alternate views for individuals who are logged in vs. logged out
 - Designing a toolbar with clean dropdowns, a search bar, and login/logout buttons
 - Designing the payment feed (Venmo style)
 - Designing a social sidebar within the payment feed view (horizontally shrinks or stretches the payment feed when expanded)
 - Alternate views for either a “social home page” (feed/socials) vs. “nonprofit home page” (trending nonprofits, featured nonprofits, etc.).
- Design and implement the registration / login forms for users.
- Design and implement the nonprofit pages including:
 - Ratings / quick evaluation sidebar
 - Overall box for information on nonprofits
 - Links to relevant sites and information
- Design and implement the users profile page
- Design the schema for our PostgreSQL database.
- Design and implement the look and feel for a user “donating” through our website.

Tasks Relevant to Integrating Functionality with Front End

- Implement registration, login, and logout functionality for users. Also implement the ability to edit user information such as password or other aspects of the profile.
- Implement mock payment for users. No transactions will actually go through but we will save relevant information in our DB.
- Implement mock payment feed for users based on the mock payments made stored in our DB.
- Pull in non profit information from our API of choice and use that information to populate relevant components (mainly in the nonprofit page)
- Implement search functionality for users (users should be able to search for nonprofits)
- Implement the ability to “favorite” nonprofits.

Wrap Up: Polishing existing features and creating documentation

Additional Stretch Goals:

- Implement charts and statistics for users and nonprofits
- Implement a way to automatically grab non profit information in order to create default pages for new nonprofits in a way that provides value to users (move away from manual curation)
- Implement a way to parse government documentation to extract relevant information from forms.

4.5 Schedule

Task	Summary	Date
Ramp Up		11/16 - 01/25
Gain familiarity with tech stack and development standards	<ul style="list-style-type: none"> - Front-end: Angular + HTML + SCSS - Back-end: NodeJS + Express - Database: Heroku PostgreSQL - Sprint cycle expectations 	
Gain familiarity with overall project design	<ul style="list-style-type: none"> - Planned features - Use cases for our project - Overarching architecture 	
Learn more about nonprofits	<ul style="list-style-type: none"> - General requirements fulfilled by nonprofits - Government documentation and spending - Develop basic idea on how to evaluate nonprofits based on available information - Locate an API with nonprofit information listed that will fit our use case 	
Design database schema	<ul style="list-style-type: none"> - Evaluate all the information we will need to store and query - Create diagram for easy visualization 	
Phase I: Adding Core Features		01/25 - 02/22
	<ul style="list-style-type: none"> - Create the structure for the endpoints exposed in our backend 	

AltruSight

Design and implement backend API	<ul style="list-style-type: none"> server - Implement backend API using NodeJS - Host server using cloud services such as Heroku 	01/25 - 02/22
Continue to design and implement database schema	<ul style="list-style-type: none"> - Polish, plan and implement the structure for our database - Create documentation on standards for our codebase on how to access and push information 	01/25 - 02/08
Begin Home Page	<ul style="list-style-type: none"> - Toolbar with a search bar, dropdowns, easy access buttons (visible across all pages) - Payment Feed (Venmo Style) - Hardcoded - Social Sidebar 	01/25 - 02/08
Implement user authentication services	<ul style="list-style-type: none"> - Create forms, pages, and modals for user login and registration - Implement a robust user authentication; for example, Firebase Auth - Ensure user login status is persisted immediately thru the web app 	01/25 - 02/08
Begin nonprofit pages	<ul style="list-style-type: none"> - Page with ultimately filterable and searchable list of all nonprofits, pulled in from external API - Page for individual nonprofits containing a ratings / quick evaluation sidebar, main panel for more detailed information, and quick links to relevant sites and information 	02/08 - 02/22
Begin user profile pages	<ul style="list-style-type: none"> - Contains initial login information - Ability to edit certain user information - Ability to add and remove profile picture - Option to logout or delete account 	02/08 - 02/22
	Phase II: Add functionality to core features	02/22 - 03/22

AltruSight

Improve home page (alternate views for social / nonprofit based content)	<ul style="list-style-type: none"> - Logo in toolbar - Alternate views for either the social home page (feed/socials) and a nonprofit based home page (trending nonprofits, featured nonprofits, news, etc.) 	02/22 - 03/08
Add alternate views for logged in vs. logged out users	<ul style="list-style-type: none"> - Ideas include splash page on home page, blurred out content, removal of dropdowns in toolbar - Redirection from all URLs non accessible to users back to default home page 	02/22 - 03/08
Implement search functionality	<ul style="list-style-type: none"> - Search with live filter and dropdown results for nonprofits both in main toolbar and nonprofit list page 	02/22 - 03/08
Implement mock payment for users	<ul style="list-style-type: none"> - Locate a secure and robust external payment API to use for future iterations - Create buttons, forms, and responses “mocking” a payment page and confirmation - Save “payments” data in DB - Easy access from main pages and nonprofit pages 	02/22 - 03/08
Implement ability to “favorite” nonprofits	<ul style="list-style-type: none"> - Store favorited nonprofits and associate with users in DB - Display favorited nonprofits in relevant locations such as home page and profile page - Show special display / status when on a “favorited” non profits page 	03/08 - 03/22
Implement mock payment feed for users	<ul style="list-style-type: none"> - Use saved “payment” data in DB to populate social payment feed for users 	03/08 - 03/22
Phase III: Stretch Goals		03/22 - 04/19
Differentiate between user and “nonprofit” accounts	<ul style="list-style-type: none"> - Nonprofits can sign up for an account using an authorized email from the organization (probably won’t be able to implement this 	(If applicable) 03/22 - 04/05

AltruSight

	<ul style="list-style-type: none"> - part) - While in nonprofit account mode, a nonprofit user can edit their specific page information 	
Implement donation progress bar	<ul style="list-style-type: none"> - Users can set individual goals for a time period and track their “donations” through a progress bar - Nonprofits can set goals for fundraising and track their progress through a progress bar 	(If applicable) 04/05 - 04/19
Implement charts and statistics for nonprofit information and user information	<ul style="list-style-type: none"> - Charts summarizing nonprofit spending - Charts summarizing user donations through time - Other statistics can be visualized as well 	(If applicable) 04/05 - 04/19
Parsing relevant government documentation	<ul style="list-style-type: none"> - Extracting information from publicly available documents released by government / nonprofits 	(If applicable) 04/05 - 04/19
	Wrap Up	04/19 - End of Semester
Polish documentation on project		
Fix major issues		
Create final presentation and project report		

4.6 Deliverables

- Web application with 4+ main views: a home page with interactive social feeds, a customized page for each nonprofit supported, a user profile page, and a page devoted to the aggregation of nonprofits.
- Database schema diagram designed for Heroku PostgreSQL
- Mockups describing the main pages of the web application, similar to the two mockups provided under the High Level Architecture section.
- Description and schema of the backend API designed in NodeJS and Express
- Description and/or diagrams of interactions between the Node / Express backend, Angular frontend, and PostgreSQL database
- Link to repos for front end and back end development

- Updated schedule reflecting true progress throughout the sprint cycles
- Description of external APIs used
- Description of core design decisions made such as decisions on frameworks and technologies used, UI components, and overall user flow
- Final project presentation as well as the created codebases in a .zip format

5.0 Key Personnel

Rafael Toche Pizano - Toche Pizano is a senior Computer Science major in the Computer Science and Computer Engineering Department at the University of Arkansas. He has completed courses that are relevant to this project such as Software Engineering, Programming Paradigms, Database Management Systems, and Programming Foundations I & II, among others. Toche Pizano has relevant experience in the field of software engineering as a result of interning with companies like Ability LLC (Software Engineer Intern), J.B. Hunt Transport Services, Inc (Software Engineer Intern), Deutsche Bank (Technology Analyst), and Walmart (Data Science/Machine Learning Intern). Toche Pizano was responsible for the abstract, problem, and objective sections. Toche Pizano assisted in the PowerPoint presentation video and will help in creating the foundation for the backend and database that will be used in our web-application.

Seth Piepergerdes - Piepergerdes is a senior Computer Science major in the Computer Science and Computer Engineering Department at the University of Arkansas. He has completed courses relevant to the development of this project like Database Management Systems, Programming Paradigms, Software Engineering, and others. Piepergerdes has some relevant experience in the tech industry through interning with Cerner (Software Intern). Piepergerdes was responsible for the background section, including the key concepts and related work sections of this proposal. Piepergerdes also helped with preparing the powerpoint and presentation before recording it. Piepergerdes will be responsible for helping develop the entirety of AltruSight including working with our entire tech stack.

Ben Guthrie - Ben Guthrie is a senior Computer Engineering major in the Computer Science and Computer Engineering Department at the University of Arkansas. He has completed courses that are relevant to the development of this project such as Programming Foundations I & II, Programming Paradigms, Software Engineering, and others. Guthrie has relevant experience with web development from the final project in the Software Engineering course. Guthrie was responsible for the risks section. Guthrie will be responsible for helping in full-stack development on each section and task listed above.

Andrew He - Andrew He is a senior Computer Science major at the University of Arkansas. Courses completed include Programming Foundations I & II, Programming Paradigms, Software Engineering, Database Management Systems, among others. Relevant experience includes a summer research experience at University of Arkansas for Medical Sciences studying the

viability of graph databases in a medical setting, an Applications Development internship at J.B. Hunt where he helped develop a mentorship application based in the Angular Framework, and a Software Development Engineer internship at Amazon where he created tools for analysis of customer queries to Amazon Prime Video and the impact of changes made to a specific understanding of queries within the service. For the preliminary report, Andrew set out a basic tasks list alongside the schedule and deliverables. Andrew has also been responsible for contributing to the initial presentation video as well as maintaining the page for AltruSight on the Capstone website. During development, He will be highly involved in full stack development for the web application.

Tanner Edwards - Tanner Edwards is a senior Computer Science major at the University of Arkansas. He has completed courses relevant to this project such as Programming Foundations I and II, Programming Paradigms, Software Engineering, etc. Edwards has relevant experience through a software internship with Cerner where he was responsible for uplifting the rest API used by the company as well as updating the dependencies. Edwards was responsible for the high level architecture section in the preliminary report.

5.0 Facilities and Equipment

Our team does not need any extra or special facilities or equipment. This project can be developed with laptops/PCs and internet connection.

6.0 References

- [1]. Your Guide To Intelligent Giving. (n.d.). Retrieved November 10, 2020, from <https://www.charitynavigator.org/index.cfm?bay=content.view>
- [2] Price, W., & Price, N. (2019, January 04). Who Governs Nonprofit Organizations? Retrieved November 10, 2020, from <https://www.boardeffect.com/blog/governs-nonprofit-organizations/>
- [3] Addressi, F. (2018, May 21). The 50 Worst Charities in America- How to Keep from Being Scammed. Retrieved November 10, 2020, from <https://smartasset.com/mortgage/the-50-worst-charities-in-america-how-to-keep-from-being-scammed>
- [4] Where can I find out who has donated money to a particular nonprofit organization? (n.d.). Retrieved November 10, 2020, from <https://learning.candid.org/resources/knowledge-base/researching-nonprofit-donations/>
- [5] Mariah Ross, Katelynn Rusnock, Informed Giving: Information Donors Seek Before Donating, Charity Navigator. Retrieved November 10, 2020, from

https://www.ftc.gov/system/files/documents/public_events/982093/charitynavigator_information_donorsseekbeforedonating.pdf

[6] CharityWatch Mission & Goals. (n.d.). Retrieved November 11, 2020, from <https://www.charitywatch.org/about-charitywatch/mission-goals>

[7] FIFA probe: First banks admit money laundering role. (2020, April 30). Retrieved December 03, 2020, from <https://www.usatoday.com/story/sports/soccer/2020/04/30/fifa-probe-first-banks-admit-money-laundering-role/111646194/>

[8] Who are Fifa and what do they do? - CBBC Newsround. (n.d.). Retrieved December 03, 2020, from <https://www.bbc.co.uk/newsround/32994511>