Project Name: Pay it Fwd

Project Members:

Jonah May [jrmay@princeton.edu](mailto:jrmay@princeton.edu) (front end)

Yasmeen Almog [yalmog@princeton.edu](mailto:yalmog@princeton.edu) (back end)

Christopher Giglio [cgiglio@princeton.edu](mailto:cgiglio@princeton.edu) (middleware)

**Section 1: Overview**

The goal of this project is to provide an efficient way for verified charitable causes to connect with the maximum number of givers possible. Oftentimes, worthy charitable causes go viral on social media; however, end users of these platforms often find themselves unable to donate to the cause. Either no donation option is available, or social media users have to click through multiple links and input their payment information in order to give charitably to a cause, providing a barrier to entry for most people. With this project, we want to simplify and streamline the donation process in order to help small charitable causes become more visible, increase returns for charitable causes, and make the donating experience more enjoyable. In order to achieve this goal, we plan to deliver a mobile iOS application called “PayItFwd” (name still being worked on). The app will consist of a feed (picture, video, and text) of causes that users can like and comment on. Most importantly, users will be able to donate using a “one-click” donation feature. In addition to an efficient donating platform, we hope to create a philanthropic community on the application, and ultimately help people.

**Section 2: Requirements and Target Audiences**

Our project hopes to solve the problem of interested individuals not donating to causes due to inconvenience. In particular, we want to target users who would share or like posts from accounts such as [Humans of New York](https://www.facebook.com/humansofnewyork), a social media sensation that garners 6 figure likes and 5 figure shares regularly. Our product will benefit users on both sides of the donation: would-be donators get a faster and easier way (and in some cases just *a* way) to give to the causes that they care about, while organizations can reach a broader range of potential donors. As it currently stands, just about every charitable organization has an online donation system through their website. However, each of these sites are disparate and require separate payment information. There are sites, such as Indiegogo or Kickstarter, that consolidate this process and allow users to give to a wide range of organizations. These sites, however, are not specific to charitable causes. Our project aims to create a platform for giving only to charitable causes. The intended users of our platform include individuals who are interested in becoming donors. On the other side, our users will be organizations looking for donations. We will first focus on large organizations, in order to ensure their credibility. However, on the platform, these large organizations can submit multiple “causes”. For example, the Make-a-Wish foundation can create separate funds for different campaigns.

**Section 3: Functionality**

**Use Case Scenarios:**

**Scenario 1: Daisy the Donor**

Daisy is a junior at Princeton university. Daisy is quite active on social media, having a Twitter, Facebook and Instagram account. She loves following accounts like Humans of New York, and always watches viral videos in her news feed. One day, her friend Megan tells her about PayItFwd, an app that puts together all the feel-good posts and videos she loves in her news feed, and makes it easy for her to donate to these people if she feels so inclined. This piques Daisy’s interest, so she decides to make an account.

Daisy downloads the app, and creates a profile for herself. Her profile includes a username/handle (she uses the same handle that she has on Twitter and Instagram), a profile photo, and a small bio. After she starts donating to causes, scrolling through her “profile” will also show the posts that she has donated to.

After her profile is created, it is time for her to build her newsfeed and start exploring the different causes and posts that exist in the app. Daisy will be asked to “subscribe” to a few basic threads so her newsfeed can be established. This can be done in a number of ways. She can select countries or geographic regions in the world, types of causes (i.e. hunger, disease, global warming), or “friend” individual people (likely similar to friending on Facebook). Daisy can subscribe (or unsubscribe) to new places/causes/friends at any time. These subscriptions outline the types of posts that will populate her newsfeed.

Once her newsfeed is populated, it is time for Daisy to become an active user. As she scrolls through her newsfeed, Daisy can both “like” and “comment” on posts. She can “share” a post (meaning that it will appear in her friends’ newsfeeds, much like a “retweet”). Finally, she can “donate” to a post. The donate button will be a small button located directly beneath the media of the post, next to the “like” and “comment” buttons. By pressing on this button, it will expand into different possible dollar values (10 cents, 25 cents, 50 cents, $1, and $X). Daisy merely selects the amount she would like to donate, uses her fingerprint for verification, and the transaction is complete.

**Scenario 2: NYCause (potential charity)**

NYCause is an organization that is based in New York City. It consists of blog posts that are usually photos accompanied by a powerful story. Often times these photos depict individuals who are working through struggles and have been down on their luck. Although NYCause has gained a lot of traction on different social media platforms, it has struggled to raise enough money. Some of its posts receive millions of likes - just think, if each “liker” merely donated 10 cents, the subject of that post could be $100,000 richer. It could change their lives forever. NYCause was feeling extremely bummed about their inability to change the lives of the people whose cause they share to the world, until someone told them about PayItFwd.

NYCause decides to make a charity profile on PayItFwd. After registering their charity name, they undergo a week’s worth of background checks and confirmation. Once the charity becomes “verified,” its profile goes public. Much like a user profile, the charity’s profile consists of a handle, a profile photo, and a bio. However, rather than populate the profile with previous donations, the organization’s profile instead consists of the individual “stories” or “causes” that the charity supports. It is up to the charity to “post” new causes (similarly to attach these causes to a given geographical or cause-based stream). In other words, if NYCause posts a story of a single mother who is battling cancer, they post it to their account, and tag it to streams in NYC (which will subsequently tag to NY state, USA, and North America), as well as “disease” or “cancer” streams. Thus the post will be sent to the newsfeeds of all users who may take interest in it.

As users donate to a specific cause, the money will be sent to the verified organization. The stipulation is that the money may only go to the subject of the specific post, not to the organization as a whole. Thus, the organization is connected to the maximum number of people, and is able to actually raise a substantial amount of money for their cause.

**Description:**

We envision our app presenting a functionality similar to Instagram. Users will be presented with a newsfeed of causes and the option to donate. Another possible feature to implement would be the ability for users to share or promote a cause either directly to a friend or to those in their area (locality could be a parameter we use to determine newsfeed ranking). We could do login either via Apple ID, which would completely eliminate the need for any login screen and tie a user’s history to that, or we could link it with a social media account such as Facebook. Because we could implement the share feature as a simple sending of a link via Facebook Messenger and thus eliminate the need for a friends list, logging in via Apple ID might make more sense as it streamlines the experience.

Outside of the newsfeed, sharing, and donating functionality, we would like to implement search, search history, and donation history features. A step beyond these would be allowing this data to be shared via another social network such that we could draw more attention to charitable causes. A brief list of features on the donator and the charity side is shown below:

* Features
  + Donator
    - Like
    - Comment
    - Donate
    - Search and subscribe to a stream
      * By location
      * By cause
      * By specific organization
    - Follow other donors
      * Like twitter
  + Charity
    - Post
      * Video
      * Photo
      * Text
    - Tag posts to specific streams
      * Location
      * Cause

**Section 4: Design**

In terms of technical requirements, we would first and foremost need to build a front end in Swift for iOS. It would have to have support for our backend, which would store different user information, such as who is a donor and who is a receiver. Moreover, we would want to track information on popular causes as well as donation history. To handle payments, we are currently examining Apple Pay and Stripe. Apple Pay offers us the ability to have access to every user’s credit card information, guaranteed, meaning that we would not have to ask them for anything in order to begin accepting payments. Stripe, on the other hand, could ask the user to re-enter credit card information. Still, Stripe will ask for a smaller percentage of the donations, whereas Apple Pay could take up to 30% as our donations could qualify as in-app purchases.

For our backend, we will use relational database (MySQL) that would manage user objects, post objects, and their interactions. We have included a database schema mockup to show the relationships between charities, users, causes, and transactions. We would want to keep track of usage statistics to ensure that our news feed would provide the causes users would be most likely to donate to.

DBSchema.png

The User to Transaction, Cause to Transaction, and Charity to Cause links would all be one-to-many relationships.

Finally, our middleware will want to ensure proper communication between these two sides while making sure that we guarantee secure payments and verified causes. Just like Twitter verifies celebrity accounts, we want to verify our charitable cause posters. Our data structures should be such that user permissions are tightly monitored, insulated, and immutable by anyone but us. Not just our middleware, but our frontend and backend should all promote these properties.

**Section 5: Timeline**

* Project Start: March 15th
* Milestone 1: **March 21st**
  + UI Prototypes
  + Software Design Plan
  + Website
  + Repository Setup
* Milestone 2: **March 28th**
  + Preliminary Implementation of Frontend Navigation
  + Preliminary Implementation of Backend Services
* Milestone 3: **April 8th**
  + Connected Frontend and Backend for Most Basic Tasks
  + More Developed Frontend and Backend Implementations
* Milestone 4: **April 18th**
  + Alpha: Added Features and Stability (TBD)
* Milestone 5: **April 25th**
  + Beta: Added Features and Stability (TBD)
* Demo: **May 2nd**
* Final Report **Dean’s Date**

**Section 6: Risks and Outcomes**

One major source of risk comes from the need to learn a lot of new technology. Some of our group members have a good amount of experience with web development, but less with mobile development. As a result, we will have to learn a lot as we go. Potentially, this could lead to a slower development time and we could fall behind our proposed schedule. However, with a wealth of resources, we are confident we can quickly find solutions to issues that arise.

Our project also has a dependency on a database of organizations and causes for users to donate to. In terms of the long term feasibility of this project, this is absolutely critical. It would be stellar if we could bring live organizations onto the platform. However, this preferred path is unlikely during the course of this semester. For this project, we will focus on the software engineering and design, rather than customer outreach and philanthropic organization acquisition. Therefore, we will simply populate our database with premade organizations and causes. This will reduce our reliance on external data source, and should reduce our risk as well.

An additional risk comes from the fact that we will be performing financial transactions. If we were to carry out this project long term, it is important that we ensure the security and validity of all the financial transactions that occur. Using existing API’s such as Stripe or Apple Pay will help mitigate the risk of faulty financial transactions, however it is crucial that we ensure the security of our users so that our application is trustworthy.