

Project Milestone 3: High-Fidelity Prototyping

Comp 3020 A01 Group 7

Julia Stoyko, Elizabeth Kenyon, Rebecca Tiessen, Chi Fung (Andy) Lun

For our project, we created an application to keep track of and allow communication between users in regards to who has borrowed from and lent you money. In our previous research we came up with a few key requirements to focus on. One of these requirements was to have a detailed and alterable list of who owes the user money and vice versa. It also needed to be easily accessible for users with different levels of computer knowledge. In the coming paragraphs we will explain how we managed to implement our expectations. We also felt that other “could haves” would be an asset to our program, such as a notification centre to send reminders, as well as a friendly atmosphere to help ease tension that comes with the potential awkwardness of money-lending.

We began our work by building a general horizontal prototype to be able to see the big picture and how the screens would interact. We used the Bootstrap framework in order to have a clean interface as well as allowing us to focus on more complex design. We then moved to our vertical prototyping where we used two of our three storyboards; confirming a payment and adding a “tab”. The major features we spend most our time on was the home screen display, including debts and debtors of the user, the ability to add a new tab and have it appear on the home screen, as well as the feature to confirm that a payment has been made.

Major Features

The home screen consists of a bubble-centered interface with bubbles that represent other people that owe the user, and people that the user owes. The bubbles are interactive, allowing the user to click on them to confirm that a payment has taken place. Our bubbles are color-coded black and red, where black means someone owes you money and red means you’re “in the red”, or owing someone else money. We chose red and black to accommodate possible color blindness in users, as well as providing a helpful visual cue.

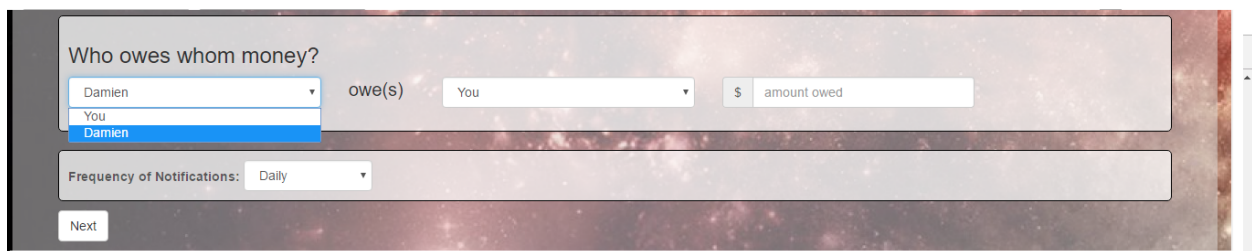
There is a navigation bar at the top of the screen. We decided to use icons instead of words, to represent the different navigable areas. We added a mouse hover feature as well, to give a text explanation to our navigation icons so that they weren’t ambiguous, and to help the user navigate around the site (see Figure 1).

Figure 1: A tooltip is shown to explain icons



When adding a tab, the amount owed is linked to the size of the bubble appearing on the screen, to provide a visual representation of money owed so it can be seen at a glance. Although the site would work best between users with existing profiles, there is an option to create a tab using someone's email so that you can keep track of the debt without the other person needing an account. We took advantage of dropdown menus in order to speed up the process and save the user needing to type names over and over again (see Figure 2). There is also a place to select the frequency of notifications sent, this was one of the features that was well liked in our storyboards. One of the other suggestions that we took from our storyboard evaluations was to put sending a personal messages in a separate step. We decided to add that feature to the notification center, so users still would have that option, but would not have to complete so many steps in the initial adding of the tab.

Figure 2: A dropdown selection of the “other user”

The image shows a web form titled "Who owes whom money?". It features two dropdown menus. The first dropdown menu is labeled "Damien" and has a list of options: "Damien", "You", and "Damien". The second dropdown menu is labeled "You" and has a list of options: "You", "Damien", and "Damien". To the right of these dropdowns is a text input field labeled "amount owed" with a dollar sign icon. Below the dropdowns is a section labeled "Frequency of Notifications:" with a dropdown menu set to "Daily". At the bottom left of the form is a "Next" button. The background of the form is a dark, textured surface with a subtle pattern of small, glowing dots.

Another feature we focused on was the ability to confirm payments. When on the home screen, a user can click on any bubble to either confirm that they have been paid the amount owed by the person, or to send a confirmation request saying that you've paid the debt (see Figure 3). We wanted to set up a two-person interaction between a payment confirmation to ensure both sides of the payment are involved. We used the “Wizard of Oz” prototyping to simulate interaction with our confirmation task. When a confirmation request is sent, we set up a timer that pops up an alert after a few seconds to say that “Person X has confirmed your request”. This is how an actual confirmation request would be implemented with real user profiles. In our storyboard evaluations, evaluators were concerned that the wording was not explicit enough. We changed the wording so that it asked if the debt was paid in full, and the user could either confirm, or choose “not yet”. We felt that this was clear, but did not seem overly negative.

Figure 3: When hovering over bubbles, user gains information on how to complete task.



The above were implementations of our storyboards vertically. The notification centre was also a major feature of our site, and we designed it horizontally so we could see how the layout would be and what organization made the most sense. There is a settings manager, which allows the user to change how their notifications are organized (newest debt first versus oldest) as well as changing how often notifications from you are sent. There is also an option to send a direct message to someone, which is particularly useful if someone isn't responding to the notifications you've sent, or you would just like to add a personal touch to the reminder. There are also standard messages that can be sent, so that the people who don't want to interact directly with someone don't need to come up with a message. The evaluators of the storyboards wanted a clear differentiation between notifications, and personal messages. They wanted to make sure the customized messages might be important and they shouldn't be in the same feed as automatic notifications, so we made that change from our storyboard. We have a separate page for sent notifications and messages, since those aren't as important to see and we didn't want to clutter the main notification page.

Functionality

At this point, there is some functionality that hasn't been implemented, such as the fact that an existing tab can't be modified to increase the amount owed, or decrease if the amount wasn't paid in full. A bug we found was when confirming a red bubble payment on the main screen, there is a five-second delay where the "other" user confirms that you've paid, and we've found that you need to stay on the main screen for the allotted amount of time or the alert won't come up and the bubble won't delete. Another known issue is that for a person to come up in the drop down menu on the Add a Tab screen, the "confirm" button must be clicked after entering the name.

Navigation

To run our application, ensure the appropriate files have been downloaded. Navigate to the signin HTML file and preferably open in Chrome, though there shouldn't be problems with other common browsers. Click sign in to go to the home screen, no username and password necessary at this point.

Use navigation bar at the top of the screen to go to Home, Add a Tab, and Navigation Centre screens.

Add a Tab

- 1) On the navigation bar, click on the "Add Tab" icon.
 - 2) Fill in the information
 - 3) Click next for a new bubble to be added to the screen
- * Try different amounts owed to see different sizes as well as whether you are owed or owing money

Confirm a Payment

- 1) Click on a bubble to confirm
 - 2) If it is a red bubble wait for the other user's confirmation
- * See different messages depending on the type of bubble

Notification Center

- * Make sure to click on all the buttons to see the different options that we think should be available.

Code

Handed in via Dropbox on UMLearn by Julia Stoyko (7763634)

File name: G7_Comp3020_ProjectMilestone3