Brian Engel

Project 2

In this project I have created some wireframes for the borrower flip phone application. They are attached at the end of the paper. There is also a rational of the design choices for the data and how the data can be adapted for a design in a cloud-based system for the lenders.

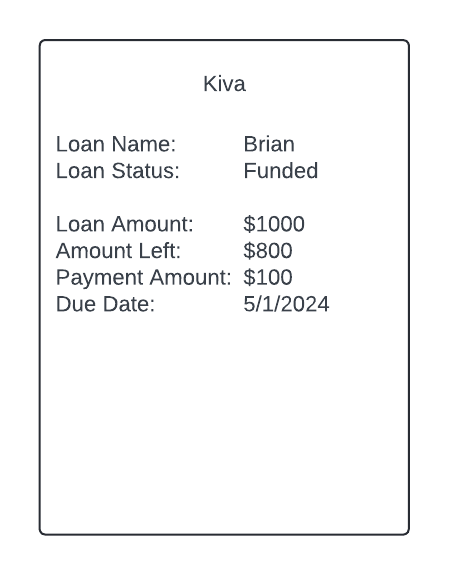
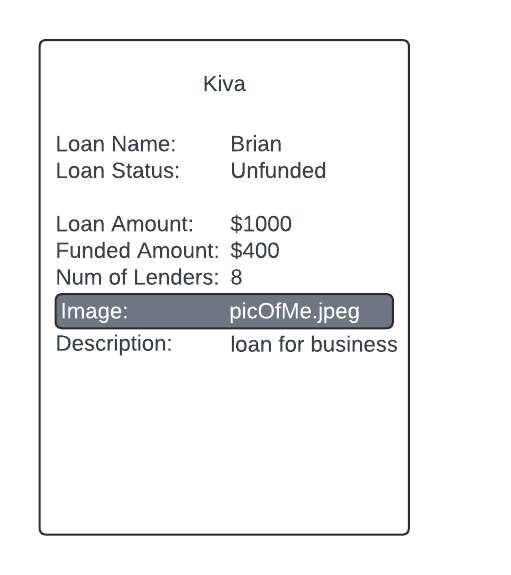
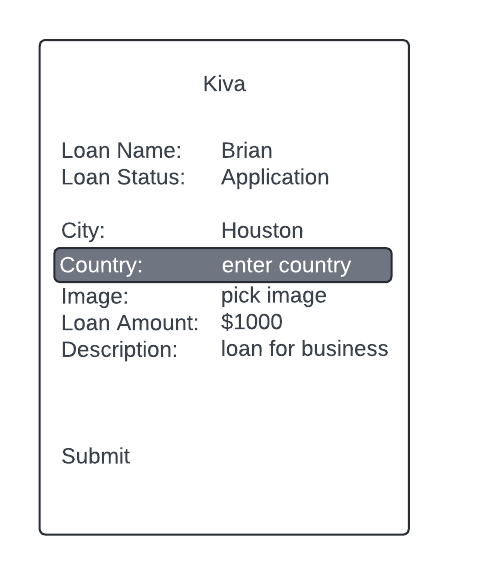
The first screen in the flip phone application is a screen to input the information for the loan into the database. There will be a highlighted row to let you know where you are in the application. All the rows will be highlightable except for the Loan Status row since that stays as is while on this screen. Loan Name will export to the LOAN\_NAME column in the database, City will export to TOWN\_NAME, Country will export to COUNTRY\_NAME, Image will export to IMAGE\_ID, Loan Amount will export to LOAN\_AMOUNT, and Description will export to DESCRIPTION. The last line will only be highlightable once all the fields have been input and when pressed will send the information to the database and continue to the next screen. Everything on this screen and the other screens will only display in portrait orientation since most flip phones don’t allow landscape.

The second screen will appear once the user has completed the first page and before the loan is completely funded. This screen is mainly for seeing how well the fund raising is going and allows the user to tweak the fields that will matter most in that aspect. We start off by just showing the name of the loan (LOAN\_NAME) and the status of unfunded, followed by the loan goal (LOAN\_AMOUNT), the amount funded so far (FUNDED\_AMOUNT), and the number of lenders (NUM\_LENDERS\_TOTAL) that have donated to the cause at this point. These are all read only for the data. Next are a couple of lines that the user will be able to select by highlighting them and pressing enter. This will allow the user to change the image (IMAGE\_ID) and description (DESCRIPTION), as these are the fields that will matter the most for trying to get funding. Once the loan has the correct amount of funding, it will automatically switch to the last screen of the application.

The final screen will display once the loan has been fully funded, and the payback process has started. This screen is just a readable screen and doesn’t have any interactions available. The top of the screen stays consistent with showing the name of the loan (LOAN\_NAME) and the status of funded. After this the total loan amount (LOAN\_AMOUNT) the amount left, the next payment amount, and the next payment due date are shown. While none of the last 3 rows has a column in the database, they might be in another database we can pull from, or we can figure it out using lender term and date the loan was distributed.

This design gives the borrower exactly what they asked for in the user story – the ability to request loans, stay informed, and track loan payback process. If they had asked for the ability to make payments on the loan with the application, we could have added lines for entering credit card information, but maybe that’s not how they are paying so it wasn’t a requirement. It also assumes the lowest level of flip phone technology so it should work on any flip phone and not exclude any of the borrowers that have a flip phone. I have taken Kiva into account in the design as well by not submitting information as a bunch of API calls and waiting until all the information is input and taking care of all of it at the same time with the submit button. There is also a guideline to not misrepresent the data. While I do need to manipulate the data to get things like monthly payments and amount left for repayment, it is not a misrepresentation of the data. Misrepresenting the data would consist more of picking specific pieces of data from the whole and representing them in a way that leads to a false conclusion. I think the submit button is an innovative solution to help reduce the load on the server with API calls. The research that I ended up using the most was the Kiva Code of Conduct (Kiva). It states what the intent of application should follow instead of hard and fast rules and lets you know the spirit of how to create your application, namely be respectful to all involved and use the minimum amount of resources possible.

Designing an interface for the lenders will require different choices since they will be using a modern smartphone. We do not need to limit the interface to just text and can use images, charts and graphs to visually represent the data. The data that we need to display is mostly the same though. We wouldn’t need an application page, but all the data from the second page is exactly what the first requirement in the persona asks for – explore borrower requests. The next requirement for the lenders would be to use the database to search through the lenders so they can see what projects are most popular and other data. The last requirement the lenders have is to follow borrower loan repayments, and that can be done with the data on the last page of the borrower application. We can definitely be more creative in this application, and use charts of payments over time, users and donation amounts, and other various visuals to help make data make more sense. To align with Kiva’s business vision and mission we need to make sure we follow their code of conduct and make the application efficient and respectful. Both applications will need to communicate with the database API, but we can cache the information and only pull it again if it expires or if the data changes. Lenders might have different reasons for wanting to help and a filter option would be useful to find the person that they deem needs their money most. Some may want to donate to a specific country, some may want to donate to a specific type of business, and some may not care at all and donate to anyone, but we should make it as easy as possible no matter the motivation.



Kiva. (n.d.). Code of Conduct. Kiva. Retrieved from <https://www.kiva.org/build/code-of-conduct>