

Budget Friendly Final Report

Saron Abebe, Ben Prentice, and Allison Spitzenberger

I. Introduction + Problem

Saving money is not easy and savings can be spent quickly if no financial measures are taken. Our approach to this problem was to create an app that provides a simplistic approach to a user that needs assistance with managing his or her spending wisely. Using an app can help newly financially independent people build healthy habits.

The purpose of the app is to assist the user with budgeting, no matter the level of budgeting experience the user has. The app will have a simplistic design and layout, as many popular budgeting apps lack this quality. Some apps send too many notifications and do not guide users through the app functionality. We are offering something more intuitive for users that would like to have more control of their spending habits.

The user will be able to log in or create an account to have access to several pages: a home page, an income page, an expense page, and a settings page where the user can log out of their account. Each of these pages separates the tasks that the user can do, allowing the user to not feel overwhelmed by searching for a particular part of the app that they need to use. The use of a navigation bar at the bottom of the screen makes the app control easy and simple to view.

This app is targeted for young adults who are learning to live independently. We want the app to be a learning tool for the user to understand how money works. The main challenge is that there are a lot of apps that are focused on managing money, each with its own pros and cons. However, we noticed the lack of budgeting apps on the market with Android, noticing a huge room for improvement with our product.

The other main challenge is that app development is new to all of the members of the group. Completing this project was a huge learning curve, with struggles along the way.

II. Related Work

Before deciding on how we would like to proceed with our application, we looked into other apps on the market and observed the pros and cons of the features and functionality. Some of the apps we looked at include:

Mint

This application was free to use and allowed email and text alerts as well as customizable settings to fit the user. The downfall was the amount of functionality on a page that made the user feel both overwhelmed and confused. This application required a huge learning curve and time to understand the budgeting power that the app has to offer.

YNAB

This application costs around \$12 a month to use and allows you to easily sync your financial information with a bank as well as being supportive on multiple devices. This application also allows you to sort information into categories for easy viewability. The cons of this application include not showing an overall view of your financial situation, such as letting the user know if they are spending and saving wisely, as well as lacking reporting features.

While these applications have good budgeting functionality and features, none consider the user first in terms of using the application. These apps lack a form of guidance to allow a new signup to learn how to navigate the application as well as focusing on simplicity to avoid overwhelming the user.

While our app may not connect to a bank account due to the limitations of our knowledge with app development, our app will split the functionalities on four different pages after the user logs in to their account. The income will be separate from the expenses so the user can easily understand their spending to income ratio.

III. Factors Considered + Methods of Approach

Our issue with budgeting is that every user has his/her own conditions. They could be married and have an additional income for support or they could be in debt. We wanted to create an application that would suit everyone, no matter the value of their income or expenses that they may be tackling.

We wanted a secure login feature for the app, so the user's information remains confidential. We used Google Firebase for our database. Firebase allows us to track users who create an account with the option to delete, disable, or create accounts through this link. On top of user information, we can view their data inputs through the real-time database section in Firebase. Firebase simplified the creation of login, signup and logout through Android Studio.

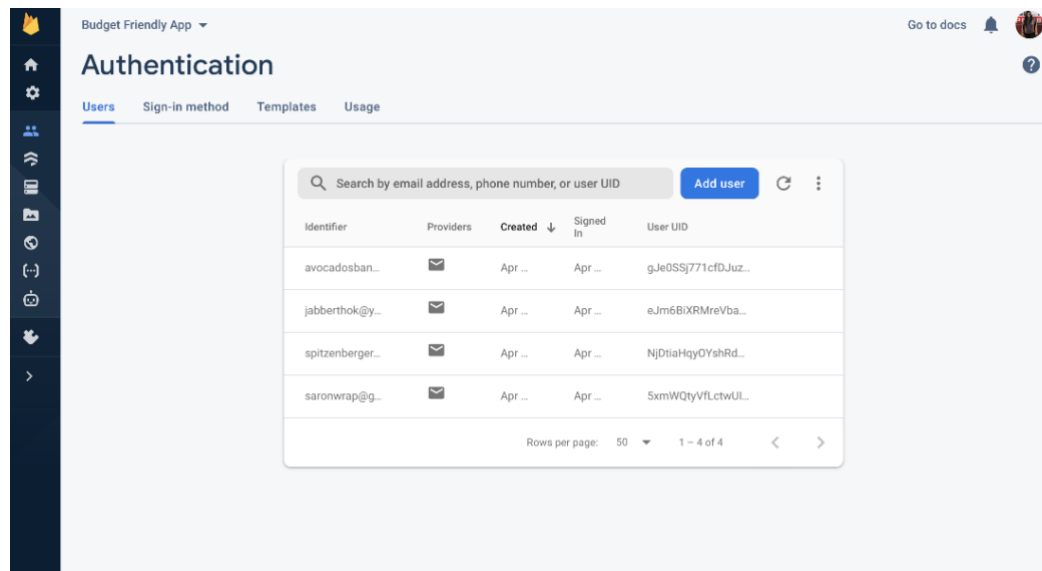


Figure 1: The Google Firebase from the admin viewpoint.

One of the human factor elements that we wanted to focus on was the visual elements. Picking a color palette that contrasted appropriately, as well as picking fonts for easy readability, was key to the project. This is explained more in the features and methods below.

The features that we used for our application include:

- A large navigation bar at the bottom of the screen with icons that correspond to the page.
- Fonts that are sans-serif and easy to read based on style and size.
- Colors that are common and appealing to the eye.
 - These colors work well with people who have a genetic form of color-blindness or difficult seeing.
 - These colors contrast with each other well.
- Large buttons for easy use of the application.

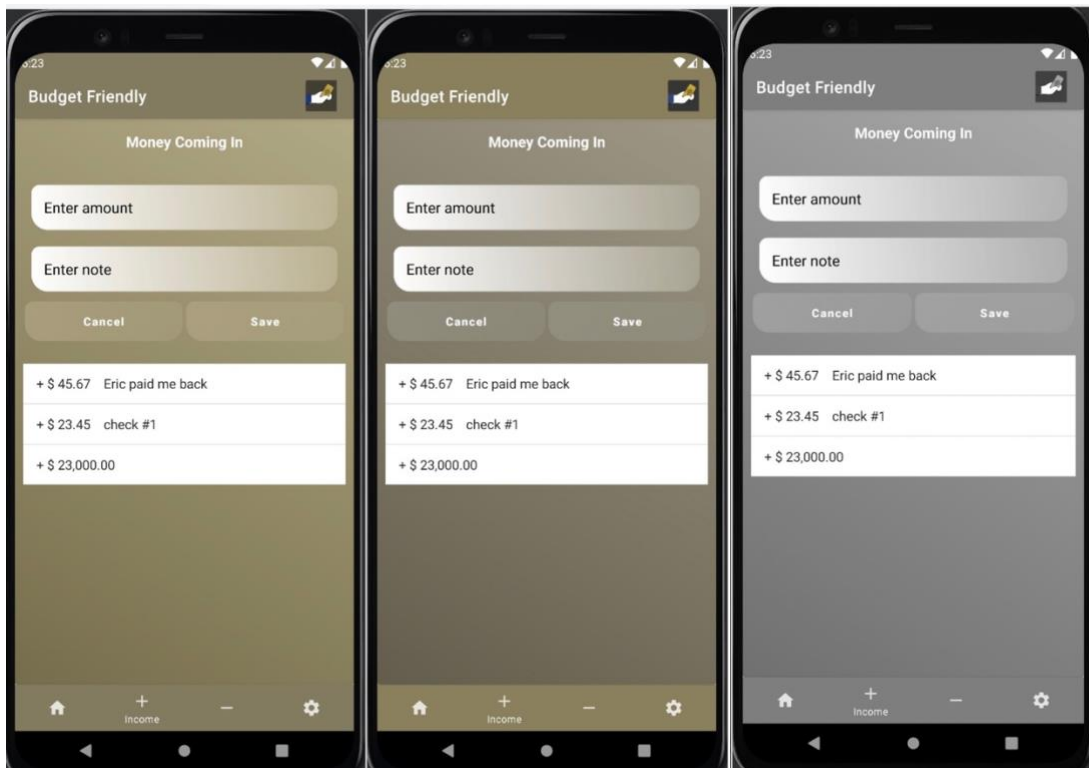


Figure 2: Color blind tests

We created a quick tutorial that shows the user how to use the app. After creating a new account, the user is taken through the steps needed to create a budget.

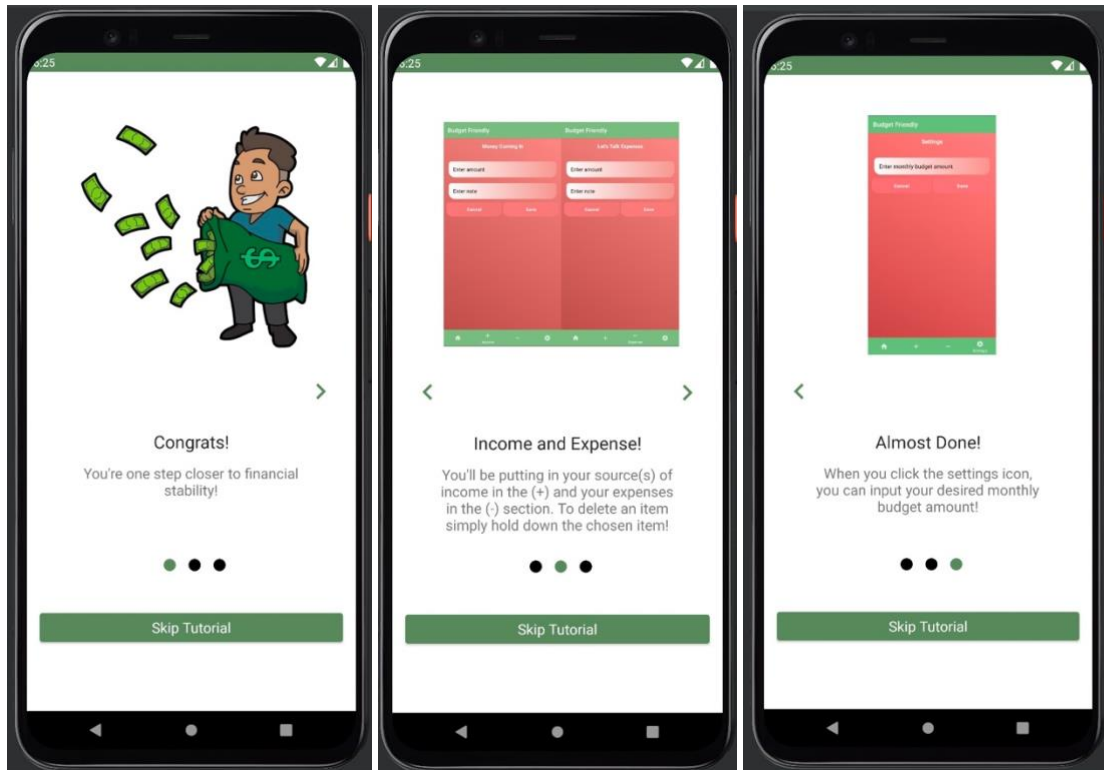


Figure 3: Tutorial

Many methods were taken into consideration when deciding the functionality of the app. First was deciding the structure and layout. We wanted to break up a complex idea, which is managing money, into smaller ideas to avoid overwhelming the user. The user will not need to scroll down a page to find a functionality that they need to use. There are also large buttons that are easy to click and easy to see.

We also used algorithms in our application which include adding and subtracting funds from a monthly budget and a master bank account. The user can add to his monthly income on the income page and subtract from both his monthly budget and his income on the expenses page. We used a data class to pass an object to the fragments so they could communicate with the same account. Users can easily delete income or expense items by pressing on the specific list item. This updates the totals on the home screen.

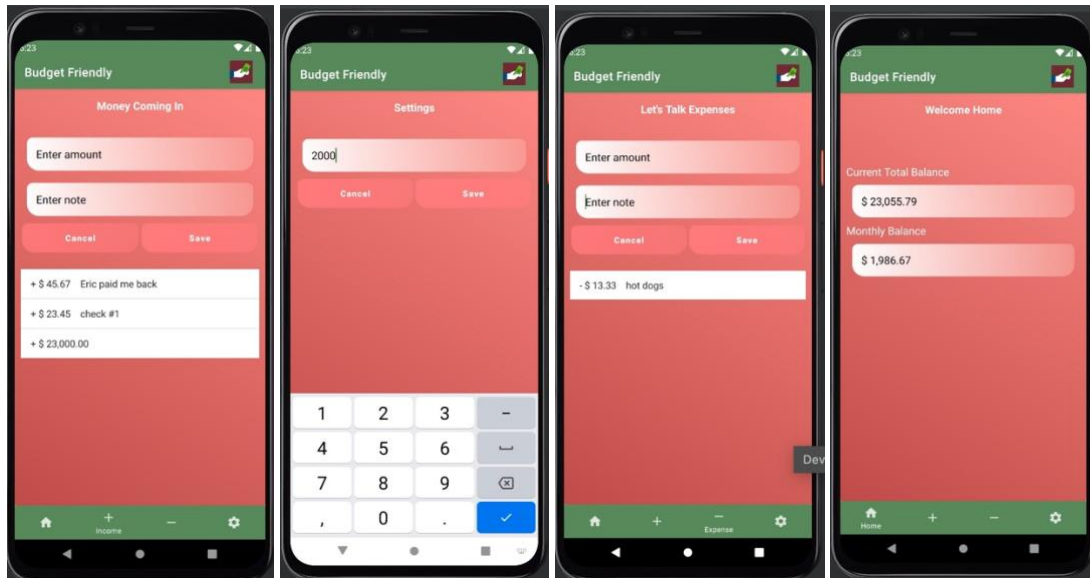


Figure 4: App functionality

IV. Data Collection + Results

Our solution wanted to include a way of keeping track of all the incomes coming in as well as all expenses. *Budget Friendly* will keep track of the expense history in case the user needs to view transaction history from a month ago.

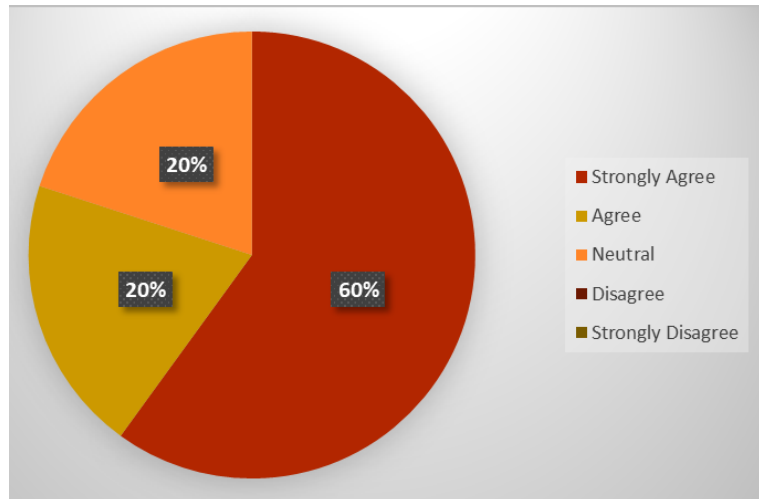
We tested different values by adding and subtracting items in different orders. Testing showed multiple errors which we were able to fix quickly.

The baseline system on Android is lacking in ease of use and is not easy to understand. Our group was confused and frustrated trying to understand how to use these apps.

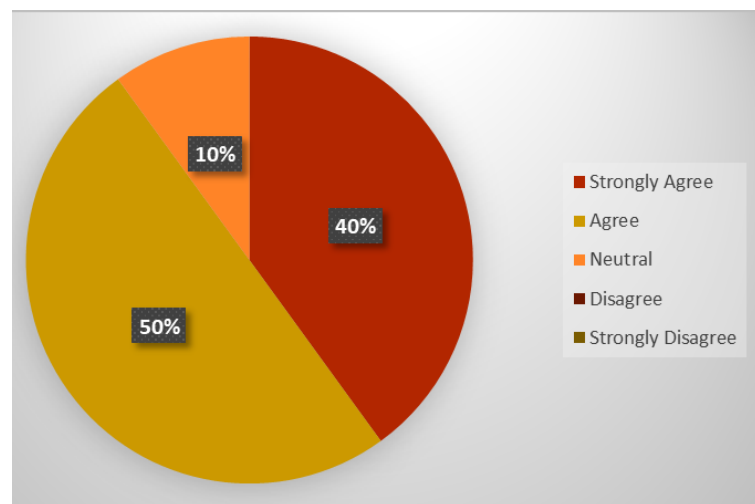
We surpassed our expectations for our front end. We wanted to have multiple sources of income which we achieved as well. *Budget Friendly* is easier to navigate than other apps because we have fewer icons and actions. It also doesn't overwhelm the user because there are no advertisements and we kept inputs to a minimum. We were unable to implement notifications, however we intend to add that functionality in the future.

We had ten people test *Budget Friendly* and then asked them to rate statements based on how well they agreed with them. The results are represented in a pie chart below.

"After logging into the app, I knew how to use it within 30 seconds."



"The tutorial after signing up was helpful."



"There is a lot of functionality within the app."

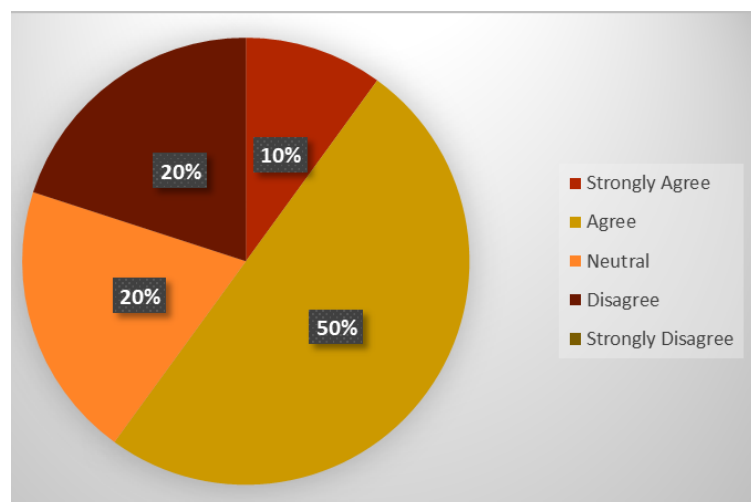


Figure 5: Graphs of results of experimental survey results

Interpreting the results, we can see that most of the feedback from people is positive. We had 2 people disagree on the last statement: "There is a lot of functionality within the app." These results are reasonable as our functionality on the app is limited due to the skills that the members have with app development as well as the limited time frame to complete the project. We had hoped to add more features to our application, but we focused on a working application before the deadline.

Everything in *Budget Friendly* works as it should and there are no errors with any of the calculations within the application from the user standpoint.

In the future, we would like to expand the app to include a sync to the user's bank account(s). This will allow the user to type in less information when on the app, as the history of expenses will be directly linked with the associated card(s) of the account. If the user knows how much is in their account(s), they can use our app to set budgets, such as having a setting to implement a need-want-save setting to control where the money will go. (An example would be 50% for necessities, 30% for personal spending, and 20% for saving.) *Budget Friendly* has the functionality for smart budgeting practices, with the opportunity to provide features to help guide the user towards financial stability in these three categories.