Willis Hand

Sridhar Rangan

Jose Solorzano

**BudgetBuddy Final Paper**

**Introduction**

This semester in Mobile Application II, we worked on building an android app which would benefit a community or group of people in some way. After forming the group, we decided to individually form idea concepts. We wanted the ideas to have some substance, so we made sure that each idea was clear about its functions and goals.

**Brainstorming**

During the brainstorming phase, we each came up with ideas and discussed each of them in detail. One of the first ideas we had was a tutoring application named Study Buddy. The purpose of the app was to help UMass Lowell students who were introverted find assistance in their classes if they were struggling. Another app idea suggested was a TV list app which provided a way for users to keep track of the shows that they were watching and discuss episodes. Similarly, a movie goer app was discussed which allowed users the same functionality as the TV list app but with movies.

After all ideas were suggested we decided it would be best to compare the potential apps and decide what features we liked and disliked about each of them. One of things that we liked about all three of the concepts was that each of the apps was user centered and focused on helping a specific group of people. Based of this and the need to build an app that would benefit the community we wanted to turn our attention to who our target audience was going to be.

**Target Audience**

To decide on a target audience, each of the suggested apps were looked at and based on those we figured that the age of the users should range from 18 years of age to 28 years of age. With the audience in mind, we wanted the app to be able to teach or demonstrate any valuable life skill for the user. The target audience range we developed also had our ages included. This allowed for us to be more relatable to the user and we could determine what we wanted by putting ourselves in the position of our users.

**Application Philosophy**

After discovering our target market, we wanted to get on the same page as to regard to app development. One of the first things we discussed was the app itself and how complex it was going to be. We figured that it would be best to keep the app as simple as possible. We kept the app simple in two regards one of which was for the building of app. The second was to have BudgetBuddy be simple and straightforward for the user to use.

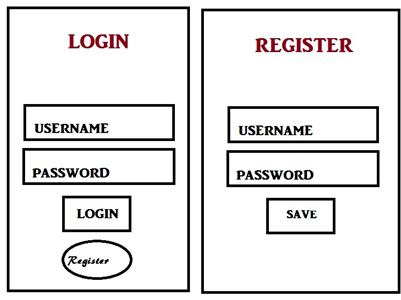
Another part of our app philosophy was to have everything kept local. As we will discuss in the login and authentication system, we believed that having the app run solely locally would be simpler. This would help to avoid any security issue as well as again help to simplify the application.

**Budgeting App Design**

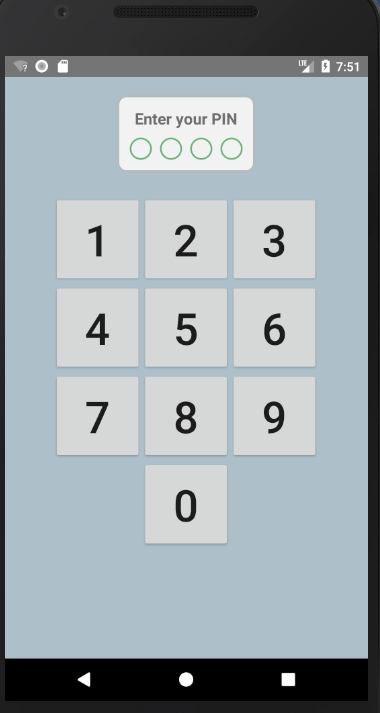
Based on our discussion, we came to a consensus that budgeting would be a valuable skill to learn. When designing the app, we wanted to focus on being able to teach the user necessary tools to be able properly budget. For the app, we decided to split the app in to three components in order to minimize any conflict in terms of files. The plan was to finish the individual pieces and then work on merging the components together. After the merge, we were to work on any bug fixes (perform testing and walking through the app) as well as user interface changes to make the app more visually appealing to use.

**Login and Authentication System (Willis)**

For the first part of this project as noted we divided the app into three pieces. The first piece was the login system. We wanted to build a login system due to the nature of the data that was going to be inputted by the user. One major concern that came up was how can we best protect personal financial data. In past, some of us had success implementing login systems using Google’s Firebase. We knew how to implement the login system via the online console provided by Google and set up and monitor accounts. However, due to the sensitivity of the data, we wanted to have a system that could minimize any security flaws. From the firebase authentication system, we transitioned into having a local authentication system. This coupled with our app philosophy allowed us to build a simpler and local system.

**Original mockup of login system**

Initially the login system was built so that the user would have to create a username and password. The credentials would be stored on the device and for each time the user returned to the app they would be prompted to enter the set password. As the app progressed, Willis suggested that a pin-based system would make the login process simpler and more effective security wise. This made sense as in the banking systems pins are used for security purposes. As such this current iteration of the app has a four-digit pin system which has the option of being reset along with the data. Since all of the data is local, the app becomes easier to use and there is less risk of any security breach.

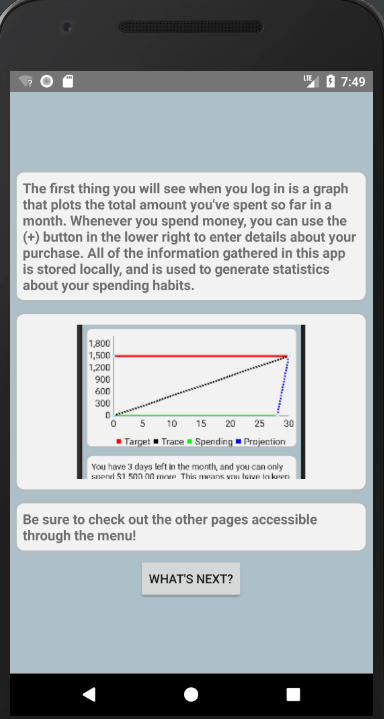
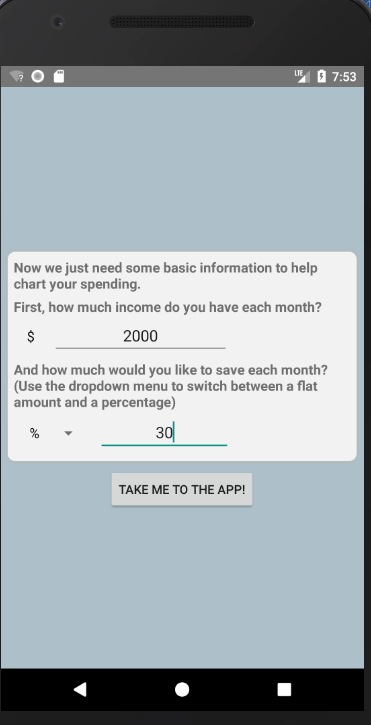
**New Pin Based Login**

**Tutorial and Budgeting (Sridhar)**

The main purpose of the app was to provide a step-by-step tutorial for the user to help them learn the basics of budgeting. When forming the concept for the tutorial we again discussed multiple approaches. One approach was to have walkthrough of the app coupled with instructions to help the user input information such as monthly income, how much they want to save and daily spending. This method would be more appropriate for users who have never budgeted before and are wanting to learn budgeting. Another approach that was considered was having multiple versions of the tutorial. The multiple versions of the tutorial would allow for users of varying expertise to use the app. With this idea, users with moderate to expert experience in budgeting would not feel bogged down with a singular novice-based tutorial.

For the final version of the app, we decided that it would be best to use the first approach in having a singular tutorial prompting user input. To accommodate users who may be familiar with budgeting we set the tutorial to occur only on the first instance of the app being used. This also went with our app philosophy of simplicity and not adding additional clutter of varying tutorials for different users.

In addition to having a tutorial teach the process for how to budget, we also wanted a page that could provide links to resources for additional financial planning. We created a tips section which has some helpful links to pages that have information about stocks and ROTH IRA for retirement. While the main focus of the app remains teaching the user about budgeting we also wanted to point the user in right direction about managing finances.

 **🡪🡪**

**(On left second page of tutorial demonstrating how the app works, on right user inputs for the tutorial)**

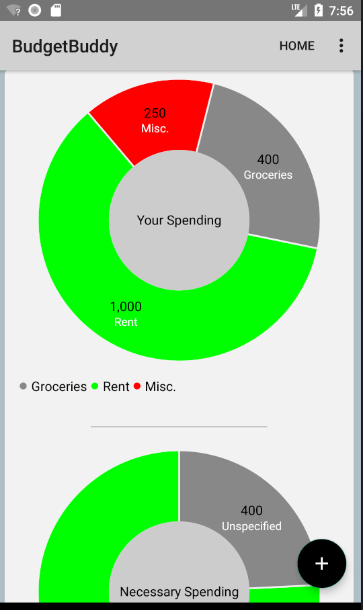
**Inputs and Graphs (Jose)**

Following our app philosophy, we wanted the inputs and graphs to be as simple as possible for the user. In regard to the inputs, in the tutorial we prompt for entering a monthly income and how much the user wants to save. After the tutorial runs, users have options to enter in daily spending amounts for different transactions. In this current iteration of the application we have allowed for users to input amounts as well as provide tags as to what the expense was for such as groceries, gas or rent.

In the tutorial we also talk about understanding which expenses are essential and non-essential. Based of their input, users can decide with a drop-down tab which expenses are necessary and which are non-essential. The tutorial mentions that if the user has too many non-essential expenses, that it may be time to cut back on spending.

For the graphs, all three us had many discussions as to how to best display information in a clear and concise manner. One graph that we settled on was having a standard graph with days on the x-axis and money spent in dollars on the y-axis. From the user inputted monthly income we drew a straight line spanning over the number of days that would serve as maximum amount that the user could spend. Additionally, we drew a similar line across of the limit that the user wanted which was entered during the tutorial. We wanted the graph to be as readable as possible, so we had an updatable line for the budget. Every time the user inputs an expense for the day, the line updates so that the budget stays under the set amount. While the graph looks clear, we made sure to have actual data points pulled from the graph so that the user would not have to estimate how much to spend on a daily basis.

One other graph we wanted to have for the user was a pie-chart. In order to account for all types of users from different backgrounds, we figured that it would be best to provide a universally recognizable chart. We provided two different charts that the users can view. The first chart shows the breakdown of user expenses relative to their spending allotment for the month. The second gives the user insight into the types of expenses and whether they were unspecified, discretionary and necessary. While it doesn’t give a detailed analysis of the information it still gives the user a way to give a cursory glance at their data. From this users can have an idea of how much they have been spending and the breakdown of where their money is being spent.

**Pie chart graphs example**

**Conclusion**

Overall, we really enjoyed working on this project. The process on working on the app was clear and we were able to systematically work through and develop the app. One part we all enjoyed was developing a coherent app philosophy. We felt that with a core belief all of us were on the same page in regard to the development of the app. The breakdown of tasks seemed to work well as all of us were able to complete individual components and then come together for the final product.