**Software Requirements and Design Document**

**For**

**Budget Baller**

Version 1.0

**Authors**:

Ismael Fernandez

Marshall Richardson

Brandon Whyte

Scott Early

# Overview (5 points)

Budget Baller takes a user’s transaction data-- the purchases they make on debit cards, debt accrued on credit cards, etc.-- and logs it in a database. It will then process this data and give users an accurate account of their financial habits with additional statistical analysis; for example, Budget Baller would tell a user how much money they have spent on groceries, eating out, or other purchasing categories, how much of their income they regularly save, and average and net changes in balance on weekly, monthly, and yearly timeframes. The app would then go a step further and help users create budgets and set goals, and keep them accountable to staying in budget and making progress.

# Functional Requirements (10 points)

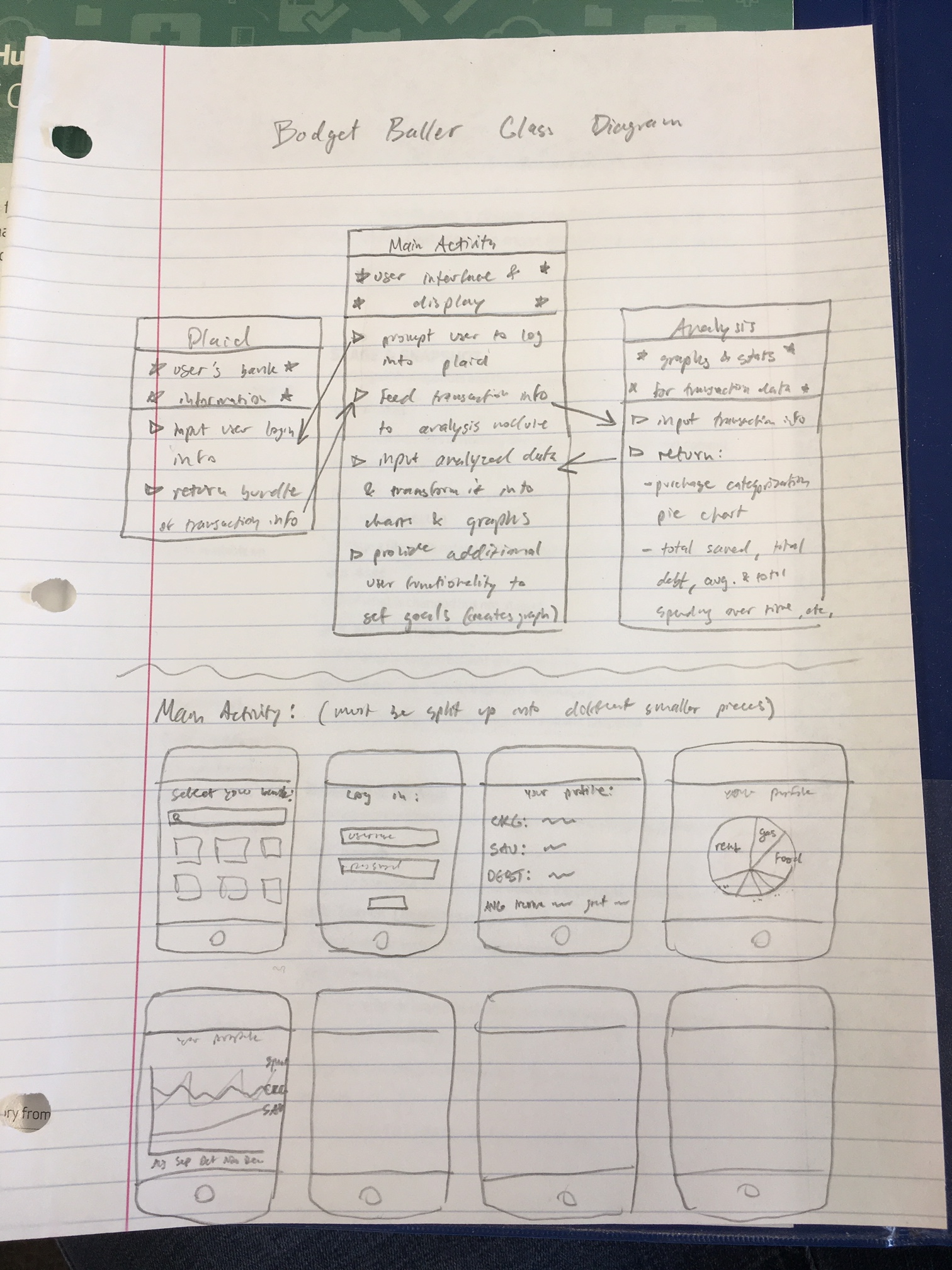
The functional requirements are as follows:

1. upon app launch, the user is connected to the Plaid API and prompted to log into their different bank accounts to retrieve their transaction data (HIGH)
2. the app will provide appropriate log in/log out functionality, allowing a user to stay logged into their accounts or to log out if they wish (MEDIUM)
3. transaction data history is used to provide the user with a pie chart showing spending in different purchase categories (MEDIUM)
4. transaction data history is used to provide the user with useful statistics such as total balance, savings, and debt and total and average income, spending, and saving (HIGH)
5. transaction data history is used to provide the user with a graph plotting changes in balance over time, cataloging spending in different purchase categories, changes in income, savings, and other statistics over a specified timeframe (HIGH)
6. the user will be able to interact with these graphs, manipulating them to see specifically desired information and allowing them to set financial goals (MEDIUM)
7. the app provides functionality for users to keep track of their goals, such as by providing regular reports or notifications that keep them accountable, or display their goals in relation to the graphs (LOW)

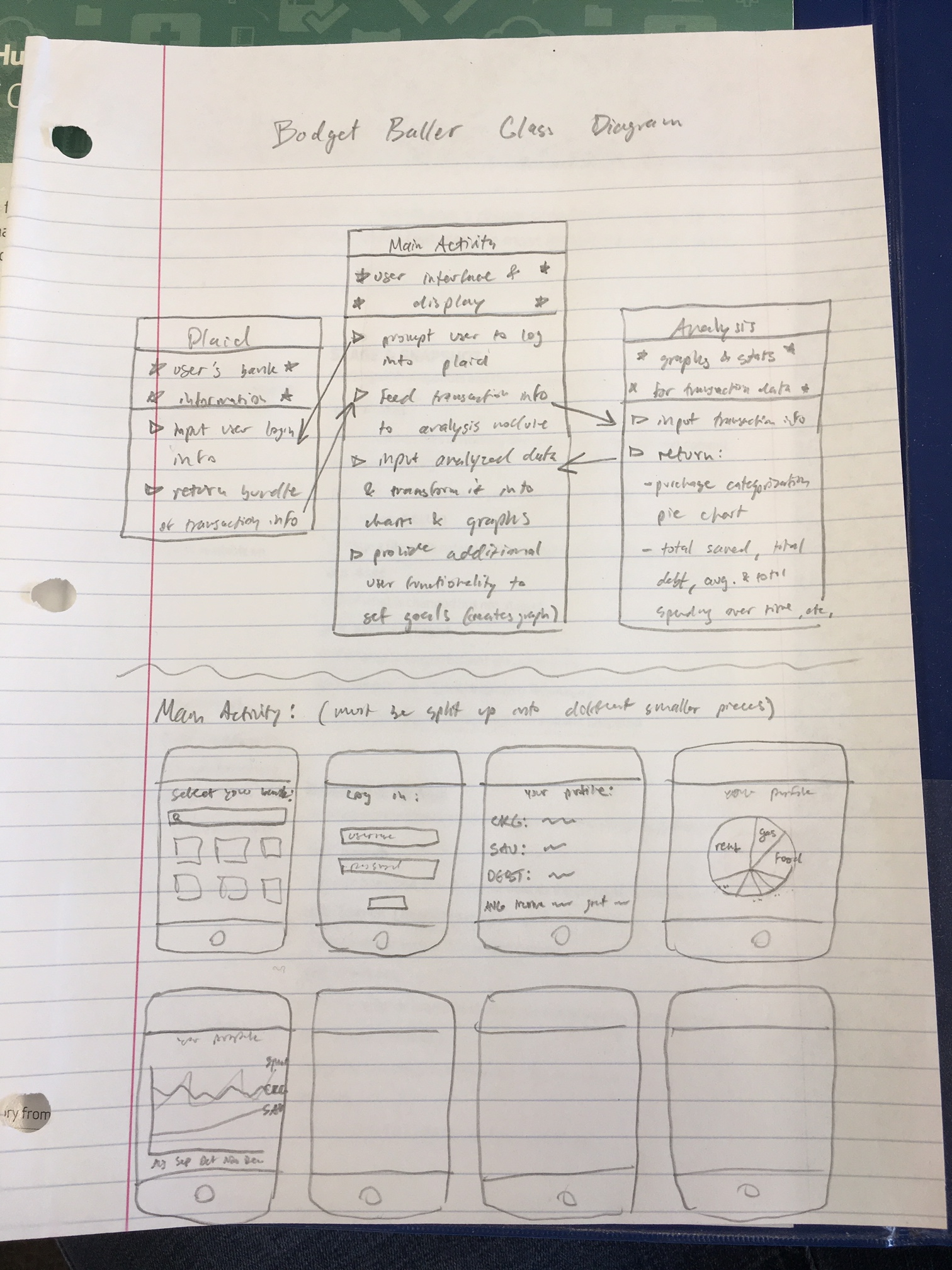
# Non-functional Requirements (10 points)

Security is an issue but Plaid has decent encryption, if we properly manage the permissions we shouldn’t have any problem with security. If we go through with a database implementation of the project we will need to add more security to protect data access from the database. If we go through without the database our main concern will be acquiring the transaction history in a reasonable time, and minimizing the space it requires. The estimate provided by Plaid indicates that the first transaction history request takes between 30 and 240 sec. If it remains running in the background this only needs to be done once. If every operation requires this amount of time we will be forced to include a database intermediary.

# Use Case Diagram (10 points)

Each Activity holds a functional requirement, which is selected within Main Activity. This includes Transaction Entry, Purchase Categorization, Statistical Analysis, Time-Based Graphs, and Budget and Goal settings.

# Class Diagram and/or Sequence Diagrams (15 points)

Class diagram rough draft:

# The main idea is, the user will sign in with the Plaid API, in which our app will incorporate into it’s analysis. Main activity will be a way for the user to access this data, and customize goals based on their needs. Analysis will be used when constructing our function requirements, such as graphs and such.

# Operating Environment (5 points)

N/A

# Assumptions and Dependencies (5 points)

N/A