Group 19

Project 3 Design Paradigm

25 October 2021

Object-Oriented Design

For project three our team decided that we would use the programming language Java because we felt that this language would offer a nice-looking UI through the GUI Library JavaFX. Java is entirely object oriented, so our design paradigm selection was rather straight forward. It follows that we decided to develop our budget tracking program using an object-oriented design. Using an object-oriented design simplified storing transaction data as well as creating user defined goals. Transaction objects hold all the data for any given addition to the budgeting list, so extracting this data is as easy as calling a getter. In order to provide useful data to the UIController, transactions are specified by date, sign(+/-), item, price and category. This information associated with each transaction is then used to provide the user with budgeting information. For example, a user may want to cut down fast-food spending, so having data about which transactions belong to the food category is a necessity. Our team also used these transactions with their category and price to provide the user with a pie chart that displays the user’s expenses per category. Essentially, we needed a way to store and extract user data and using object are perfect for solving this problem. We also used object-oriented design for the user defined goals. Each goal contains the category for the goal, and the price that the user does not want to go over in that category. Since the goals are objects, we can easily check each goal price against the amount of spending in each category, which creates a ratio, ultimately reflecting on the progress bars for each goal. Overall, we chose an object-oriented design due to the nature of a budgeting application. When it comes to monthly or yearly budgeting there is a massive amount of data that needs to be sifted through, and object-oriented programming is the perfect paradigm to sift, collect, and interpret data.