

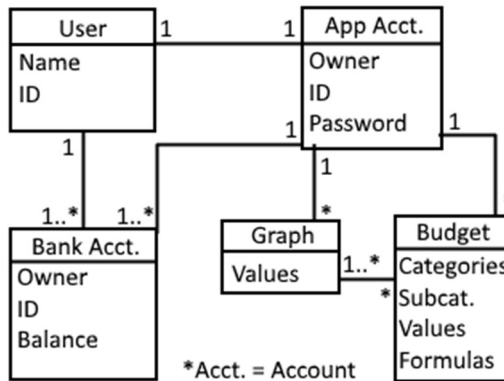
- 1) The CashStash financial management application provides functionality akin to such larger-scale applications as Quicken, similarly allowing users to view and manage their personal finances.

The application will include such tools as budgeting, checking and savings account management, and graphical representation of expenses.

To use the application, users will have to set up a personal profile in which to input and store their personal and sensitive information. This account may have security options—such as password protection—enabled, if the user so chooses. Users will be able to link their profile/account on the application with their bank account or accounts in order for the application to have access to data regarding the user's bank account. They will also be able to unlink their account at a later time and will be able to add additional linked accounts if necessary.

For budgeting purposes, the user will be able to create individual budgets (E.G. separate documents), on which they will be able to add: categories and subcategories; dollar values assigned to these categories; and formulas for calculating category, subcategory, and overall totals. Using these budget documents, users will be able to compare their financial needs with their available finances in their linked bank accounts, enabling them to adjust their spending accordingly or ensure that they limit their spending to within their budget.

The application will be able to display data from the user's budgets and linked bank accounts in the form of graphs (one example being a pie chart). These graphs allow users to more easily understand and consider their current finances and more easily make decisions about their spending.



Constraints:

User ID—must be unique; may be shared with App Account ID, as App Accounts also require unique ID and have 1-to-1 relationship with Users. **context User inv: self.ID->isUnique()**

Bank Account Balance—must be nonnegative. **context Bank Account inv: self.Balance >= 0**

Application Account Password—must fit security requirements (E.G., have at least 8 characters and contain at least one numeral). **context App Account inv: self.Password->length() >= 8 and self.Password->containsNumeral == true**

2)

