

Refactoring File

EECS2311Z 23W Team 7

Major Refactoring#1: Reduce DB Query for Some UI Classes

Before Refactoring:

There used be a query for a category in cloud database, whenever user click on generate graph button or make a change to the table (add item/update item/delete item). That reduce the respond time of all the features mentioned, which is not a good sign for user experience.

Solution:

Made a LedgerList class. Which works like a cache. It only queries when Navigator page is invoked, or user click on generate graph button.

After Refactoring:

Performance for features related to graphs are significantly improved. Code redundancy is also reduced.

Major Refactoring#2: Exception Handling

Before Refactoring:

There were some exceptions are thrown from DBUtil class to UI classes. Some of them may cause termination of the application.

Solution:

Made an ErrorPage class (implements ActionListener). This class is used to catch exceptions and pop up whenever an exception is caught. Also display the premade error message. User can click on "Expand" button to see what exactly went wrong.

After Refactoring:

All possible exceptions have been tested and handled by this class. None of the exceptions terminated the application as expected.

Major Refactoring#3: Bugs Fix for UI

Before Refactoring:

- 1) Add new item buttons for all UI pages disappear after click;
- 2) Input form pages stay open after clicking submit button;
- 3) Typos;

Solution:

Debug corresponding classes.

After Refactoring:

- 1) Add new item buttons now all stays after click;
- 2) Input form pages now all dispose after clicking submit button;
- 3) No more typos found on UI;