

Budget Sharing Bar Peer Review Modifications

Developer: Judy Li

Submitted February 17, 2023

Reviewer: Allison Austin

Reviewed February 17, 2023

Unmet Requirements

- **BRD requirement:** “Budget values have no upper-bound limit, however, progression towards the budget may not exceed the defined budget resulting in a negative budget readout.” [1]
 - **Solution:** Frontend Input validation will account for this by validating that each bill entry does not cause the budget to be exceeded with the following equation:

$$(Value\ of\ new\ bill + Amount\ Spent\ for\ the\ month) > Total\ Monthly\ Budget$$

Adopted Design Recommendations

- **Suggestion:** “Input validation should be implemented at the frontend (in BudgetBarView), and should include the use case of a user uploading a bill with a value that would exceed the defined budget...” [2]
 - **Solution:** All Input validations, such as valid characters, values and length limits, will be included in the frontend. If an invalid input is entered, the user will get an error message upon submission of the form.
- **Suggestion:** “Alternatively, you can mimic the design for storing user claims in the database. In this case, a single entry in the bills table would represent one bill, and you would have one column for the users that share the bill, and a list of users (stored as a JSON object) in the users column...” [2]
 - **Solution:** I will modify adding, updating, and deleting a bill to follow the same format. Each bill entry in the database will have a list of members rather than multiple entries for each member.
- **Suggestion:** “The return values of the database layer are not accurate/detailed. The database would not return “rows == 1”, it returns the

rows affected if you use `command.ExecuteNonQuery()`. The GroupDAO compares the number of rows ..." [2]

- **Solution:** The sequence diagrams have been updated to show accurate database returns.
- **Suggestion:** "The Bills table should have the group name included as a column. This allows for bills by the same user in separate groups to be distinct at the data access level." [2]
 - **Solution:** I will update the Bills table to include group name to differentiate between bills of different groups
- **Suggestion:** "When pulling up the BudgetBarView, you are running five queries. This can become excessive for a feature that gets navigated to very often. Also, the deletion of bills after 24 hours should be handled by the database server, not the application..." [2]
 - **Solution:** Two methods will not be implemented in favor of scheduled events in the database. Scheduled events will be implemented for deleting bills whose `date_entered` is in the past month or whose `date_deleted` is older than 24 hours. It will delete all outdated bills in the database instead of using queries.

Test Recommendations

- All test recommendations will be followed.

References

- [1] githelsui/WeCasa: @ Team HAGS JP. (2023, Feb 17) GitHub
<https://github.com/githelsui/WeCasa/blob/main/docs/BRD%20v2.0.pdf>

- [2] githelsui/WeCasa: @ Team HAGS JP. (2023, Mar 8) GitHub
<https://github.com/githelsui/WeCasa/commit/1623dee8ea0e74fffb86f5f802dbb20380ce5279>