

Refactoring Report

At first, the system has many duplicated codes in different classes, and all data can only save in the memory. There was no GUI controller so it was very difficult to operate the system. Also, there was no exceptions to handle the error messages.

After refactoring

1. Added CareHomeDatabase -> Use SQLite to save and load data.
2. Introduced Staff hierarchy -> Created a parent class and subclasses for each role.
3. GUI controllers -> Each function has its own controller.
4. Added logging system -> Actions are recorded with timestamps and staff IDs.
5. Improved exception handling -> Custom exceptions make error messages clearer.
6. Role-based dashboard -> Dashboard dynamically shows buttons by role.

Benefits:

1. GUI provides better user interface
2. Database ensures data is persistent
3. Easier to read structure
4. Reduced duplicated code

Future Improvements

1. Add Record Administered Medicine list (easier to check all the administered medicine).
2. Add Update/Delete Record Administered Medicine (can modify the record or delete the old one).
3. Add Shift list (easier to see all the staff shifts)
4. Upgrade Assign staff shift (can click on the staff and see the list of the shift and modify directly)
5. Improve error dialogs for more user-friendly feedback
6. Add report generation (export logs or resident data)
7. Upgrade UI design