

Hospital Information System (HIS) Refactorings

Group 2

- Refactoring 1: Refactoring VitalSignsDisplayGUI:

The VitalSignsDisplayGUI module underwent a substantial refactoring process aimed at enhancing the user interface and sticking to the Single Responsibility Principle. Initially, the module's design merged input and display functionalities within a single panel, leading to a complex user experience. To fix this, the GUI was refactored into two distinct panels: an "inputPanel" dedicated to gathering user inputs such as the patient ID, and a "displayPanel" solely for presenting the retrieved vital signs information. This separation not only clarified the user interface but also maintained the state of input fields for subsequent queries, thereby simplifying the input process. Complementing this change, the main method was excised from the class, transitioning the responsibility of initiating the application to the "PatientGUI" class. This refactoring choice ensured that the process was controlled by a singular entity, and enhanced the cohesiveness of the system. Furthermore, to enable smooth navigation, a "return" button was designed, allowing users with the ability to toggle between viewing vital signs and the primary patient portal interface. This addition significantly improved the workflow by offering a path back to the patient portal after the completion of tasks related to vital signs.

- Refactoring 2: Refactoring for the Nurse Portal's Patient Addition:

The system design of the Nurse Portal for patient management encountered a significant issue during the process of adding family doctor details with new patient records. The database schema exhibited redundancy with two different tables intended for storing family doctor information—one within a title 'famdoc' table and the other directly embedded within the 'patients' table. This redundancy led to a conflict during data persistence operations, resulting in a functional ambiguity that caused the system to inconsistently update neither column, effectively omitting the family doctor information upon patient record creation.

To address this, a comprehensive refactoring operation was initiated, involving a critical review and subsequent restructuring of over ten classes that interacted with the affected database schema. This included the modification of several methods to unify the patient information entry point and the removal of the extraneous 'famdoc' table. The refactoring was carried out to ensure all dependencies and connections to the redundant table were identified and appropriately rerouted or excised from the system.

The successful unification of the family doctor data into a single, well-defined column within the 'patients' table not only resolved the primary data persistence issue but also simplified the database structure. This enhanced the clarity of the system's data handling strategies within the Nurse Portal. (Note: This modification was made following the bug report)

- Refactoring 3:Refactoring Prescription Management of Patient Portal:

The patient portal's prescription management functionality initially encountered significant user interface issues that affected the overall usability and coherence of the system. Specifically, there were premature triggers of confirmation dialogs and an intuitive navigation flow post-prescription submission, which inadvertently closed the entire patient portal interface.

To address these challenges, a comprehensive refactoring of the prescription management module was undertaken. This involved a careful redesign of the user interface logic to delay the triggering of the prescription confirmation dialog until after a successful form submission. Furthermore, a "Return to Dashboard" button was introduced, ensuring that physicians could seamlessly navigate back to the main patient portal interface upon completing a prescription, without inadvertently exiting the portal.

This refactoring extended to the underlying database interactions, where enhancements were made to ensure prescriptions were stored with clear and direct associations to both patients and physicians. This not only optimized data retrieval and updates but also backed data integrity across the system.

Moreover, the initiative included an overhaul of the error-handling mechanisms to provide more precise and informative feedback to the user, particularly in instances of unsuccessful actions or validation failures. This was critical in enabling a more intuitive and error-tolerant user experience.