Design Patterns in Implementation

Observer Pattern (Qt's Signal/Slot)

Qt's signal/slot mechanism forms the backbone of our communication system:

```
- Component A (Signal) -----> Component B (Slot)
```

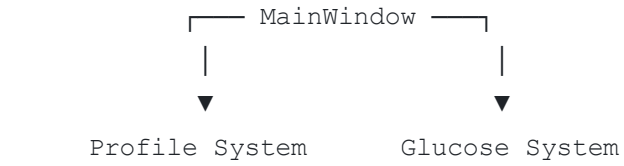
Key examples:

- Timers signal the simulation to update glucose values
- Bolus administration signals glucose level changes
- Profile changes signal updates to insulin calculations

This creates a loosely-coupled system where components react to events without direct dependencies.

Mediator Pattern

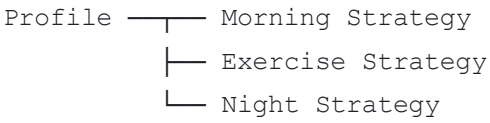
The MainWindow class acts as a mediator between major components:



It coordinates navigation between screens and facilitates communication between otherwise independent components.

Strategy Pattern

The profile system implements different insulin delivery strategies:



Each profile contains different parameters that alter how insulin is calculated and delivered, allowing the system to adapt to different situations without changing its core logic.

State Pattern

The simulation exists in different states (running, paused) that alter its behaviour:

The UI and data generation respond differently based on the current state, providing appropriate feedback and control to the user.

These patterns work together to create a flexible, maintainable system that accurately simulates the t:slim X2 insulin pump while maintaining good separation of concerns and minimizing dependencies between components.