Nama : Fadhlurrahman (171511008)

Muhamad Wahyu Maulana Akbar (171511019)

Behavioral Patterns

Design Pattern yang cocok digunakan dalam aplikasi **ATM** adalah **Behavioral Pattern** karena focus pada komunikasi antar objeknya. Penggunaan design pattern yang lebih tepat dalam system **ATM** adalah **Command Design Pattern** yang merupakan turunan dari **Behavioral Pattern**. **Command Design Pattern** merupakan Design yang lebih mengutamakan fleksibilitas dalam memanggil suatu service. Mengapa ATM sangat cocok untuk Design Pattern ini?

1. Karena pada **Command Design Pattern** mengutamakan bagaimana sebuah command mempengaruhi dalam pemanggilan fungsi lain dari class/objek yang berbeda.
2. Karena **Command Design Pattern** juga lebih dikenal sebagai Design untuk **Action and Transaction**. Dan dalam **ATM** merupakan aplikasi yang berfokus untuk melakukan suatu transaksi.
3. Mudah untuk menambah command baru(contoh pemilihan jenis pembayaran lain yg merupakan command yg berbeda), tanpa merubah class yg ada.
4. Goals dari **Command Design Pattern** adalah **Flexibility and Robustness.** Sesuai dengan kebutuhan **ATM** dimana class class pada **ATM** harus dapat memanggil objek dari class lain secara fleksibel karena fungsi transaksi yang dapat terus ditingkatkan/ditambah sehingga mudah menghubungkan fungsi transaksi baru ke yg fungsi lain. Lalu **Robustness** sangat penting untuk kekuatan dari **ATM**nya sendiri agar tidak rentan.
5. Penggunaan **Command Design Pattern** juga bertujuan untuk menghindari pemanggilan method secara langsung, sehingga sangat cocok untuk peningkatan **Robustness** dari aplikasi **ATM** sehingga tak mudah terbobol/pemanggilan fungsi tanpa perantaranya dahulu(mis : Tidak bias memanggil getTotalBalance secara langsung pada class Account tanpa melalui void getTotalBalance pada class BankDatabase terlebih dahulu)