

## Abstrak

Sebuah perangkat lunak yang dibangun berbasis objek, akan merepresentasikan dunia nyata ke dalam sebuah sistem dengan menitikberatkan pada objek-objek apa saja yang terlibat pada suatu sistem aplikasi yang akan dibangun. *Design pattern* merupakan salah satu metode pemrograman *object oriented* yang diterapkan pada awal kita mendesain sebuah sistem. Tujuan menerapkan *design pattern* pada sistem yang akan dibangun adalah untuk mendapatkan sistem perangkat lunak yang handal, yang dapat dilihat dari segi fungsionalitas, perawatan, dan pengembangan selanjutnya.

*Strategy pattern* merupakan salah satu dari *design pattern*. *Strategy pattern* muncul karena adanya method yang tidak bisa diturunkan langsung dari kelas induk ke kelas turunannya sehingga perlu dibuat interface. Interface dibuat untuk mendeklarasikan method pada kelas induk agar dapat diimplementasikan pada kelas turunannya.

Pada tugas akhir ini, dibuat sebuah perangkat lunak dengan mengimplementasikan *strategy pattern*. Untuk mengetahui kelebihan dan kekurangan sebuah perangkat lunak yang dibangun, maka dibutuhkan sebuah alat ukur yang sering disebut sebagai *object oriented metrics* atau *OO metrics*.

Hasil yang diperoleh dengan mengimplementasikan *strategy pattern* adalah dapat meningkatkan tingkat *flexibility* dan *reusability*. Selain itu, juga memenuhi karakteristik dari *well formed design class*.

**Kata kunci:** pemrograman *object oriented*, *design pattern*, *strategy pattern*, *object-oriented metrics*