Abstract

Design pattern appears because of the same problem that usually appears at software design. There are many design patterns created by many programmers lately. Nowadays, design pattern is classified by three different purposes, they are creational, structural, and behavioral.

Composite pattern is one of design pattern that belongs to the structural pattern. One of the problems in the manufacture of the software without a design pattern is a simple interface which can access a collection of objects in a composite structure and has the ability to distinguish between primitive objects and composite objects. Problems like this can be solved by the composite pattern.

In this Final Assignment, a software is created to implement composite pattern in problem solving. To evaluate the strengths and weaknesses in the composite pattern, the calculation results will be compared to object-oriented software metrics that implements the composite pattern with software that does not apply to a composite pattern of similar cases.

Based on the analysis and testing, when seen from the calculation of object-oriented metrics, software with a composite pattern having a higher complexity and requires a greater effort when compared to software maintenance conducted without the composite pattern. However, when seen from its structure, composite pattern software with a structure that can easily to be applied again on different platforms (reusability).

Key words: design pattern, composite pattern, reusability and object-oriented metrics.