

ABSTRAK

Dependency Injection (DI) merupakan metode penting dalam pengembangan perangkat lunak berorientasi objek yang memungkinkan pemisahan antara pembuatan objek dan pengaturan dependensinya. *DI* meningkatkan *modularitas*, kemudahan pengujian, dan pemeliharaan aplikasi. Penelitian ini bertujuan untuk mengevaluasi performa perangkat lunak dengan penerapan *DI* dibandingkan dengan *pendekatan konvensional*.

Dalam penelitian ini, dua versi perangkat lunak serupa dikembangkan: satu dengan penerapan *DI* dan *konvensional*. *Pengujian kinerja* dan pemeliharaan dilakukan terhadap kedua versi tersebut. Metodologi yang digunakan mencakup *pendekatan kuantitatif* dan *eksperimental*, dengan pengukuran kinerja, *waktu respons*, dan efisiensi sebagai indikator utama.

Hasil penelitian menunjukkan bahwa perangkat lunak dengan *DI* memiliki kinerja yang lebih baik dalam hal modularitas, kemudahan pengujian, dan pemeliharaan. Temuan ini diharapkan dapat memberikan panduan praktis bagi pengembang perangkat lunak dalam memilih pendekatan yang sesuai untuk meningkatkan kehandalan dan skalabilitas aplikasi.

Kata kunci: *Dependency Injection (DI)*, Pengembangan perangkat lunak berorientasi objek, Kinerja perangkat lunak, Pengujian perangkat lunak, Pemeliharaan aplikasi

ABSTRACT

Dependency Injection (DI) is a crucial method in object-oriented software development that enables the separation between object creation and dependency management. DI enhances modularity, ease of testing, and application maintenance. This study aims to evaluate the performance of software with DI implementation compared to the conventional approach.

In this research, two similar software versions were developed: one with DI implementation and the other with the conventional approach. Performance and maintenance testing were conducted on both versions. The methodology used includes quantitative and experimental approaches, with performance measurement, response time, and efficiency as the main indicators.

The results show that software with DI implementation performs better in terms of modularity, ease of testing, and maintenance. These findings are expected to provide practical guidance for software developers in choosing the appropriate approach to improve the reliability and scalability of their applications.

Keywords: Dependency Injection (DI), Object-oriented software development, Software performance, Software testing, Application maintenance