

BAB 5

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil implementasi dan pengujian yang telah dilakukan, dapat diperoleh kesimpulan bahwa pemanfaatan *clean code* dan *code refactoring* dapat meningkatkan aspek *maintainability index* pada *back end* Lusa Bot. Akan tetapi, tidak semua *Clean code* yang tersedia dilakukan seluruhnya, hal tersebut dipengaruhi oleh bentuk masalah yang ada pada kode sumber dari aplikasi *Lusa Bot* yang diteliti. Adapun pemanfaatan konsep *clean code* secara umum memberikan petunjuk kepada para pengembang untuk menulis kode sumber yang mudah dibaca agar pengembang lain dapat memahami maksud dari kode yang ditulis tersebut. Meskipun begitu, bentuk pemanfaatan konsep *clean code* dikembalikan lagi, baik kepada tim pengembang ataupun tempat aplikasi dibangun, untuk melakukan standarisasi penulisan kode sumber dari aplikasi yang dibangun. Peningkatan nilai *maintainability* pada aplikasi dapat dilihat pada Tabel 5.1.

Tabel 5.1 Peningkatan *Maintainability* Pada Aplikasi *Lusa Bot*

No	Penilaian	Nilai <i>Pra-Refactoring</i>	Nilai <i>Post-Refactoring</i>
1	<i>Maintainability Index</i>	64.80 (Rata-rata)	68.89 (Rata-rata)
2	<i>Readability</i>	0,8824 (Tim Pengembang) 0,8481 (Luar Tim Pengembang)	0,9168 (Tim Pengembang) 0,8868 (Luar Tim Pengembang)

5.2 Saran

Dalam melakukan pembangunan perangkat lunak, pemanfaatan konsep dari metode clean code adalah dengan mengikuti dokumentasi program dari bahasa pemrograman dan framework yang digunakan. Clean code hanya di jadikan sebagai acuan dalam melakukan penulisan code sehingga penggunaan clean code dapat disesuaikan dengan kebutuhan dari pengembang perangkat lunak.

Adapun saran terhadap penelitian ini ke depannya dapat sebagai berikut:

1. Implementasi clean code juga dilakukan terhadap front end dari Lusa Bot
2. Melakukan testing dengan menggunakan tools PHPUnit.