

ABSTRAK

Penciptaan dan perkembangan berbagai bentuk aplikasi sangat pesat. Sehingga dibutuhkan paradigma pemrograman yang harus dapat menghasilkan aplikasi yang baik, efisien, dan efektif, yaitu salah satunya memiliki reusability yang tinggi. Salah satu paradigma tersebut adalah Object Oriented Programming. Tetapi ada permasalahan mendasar yang tidak dapat ditangani oleh OOP, sehingga muncul paradigma pemrograman selanjutnya adalah Aspect Oriented Programming. “Aspek mengenkapsulasi technical concern sehingga tidak bercampur dengan business process, permasalahan ini biasa disebut crosscutting concern [5]”. Menurut [10], AOP adalah pelengkap bukan sebagai pengganti OOP. AOP pun belum memetakan permasalahan mendasar dari OOP, tetapi hanya mengurangi kompleksitas struktur program dengan effort yang lebih sedikit. Sehingga memunculkan paradigma pemrograman baru yang menggabungkan antara Dependency Injection (DI), Domain Driven Design (DDD) dan Aspect Oriented Programming (AOP) kedalam suatu model atau metode pemrograman objek yang disebut dengan Composite Oriented Programming [13] yang bertujuan untuk melengkapi pemrograman berbasis objek dan pemrograman berbasis aspek.

Metode ini menggabungkan best practices OOP kedalam sebuah paradigma pemrograman yang berusaha mengakomodir permasalahan mendasar dari OOP seperti pemisahan domain aplikasi, decoupling yang baik, dan behaviour suatu objek tergantung pada konteksnya.

Kata kunci : OOP, AOP, object oriented programming, aspect oriented programming, crosscutting concern, domain driven design, dependency injection , composite oriented programming.