

ABSTRAK

Continuous integration adalah praktik pembangunan perangkat lunak yang dilakukan secara tim dengan membagi modul perangkat lunak dan mengintegrasikan pekerjaan anggota tim secara rutin. Aktivitas pembangunan perangkat lunak dengan *continuous integration* dilakukan oleh sebuah tim secara manual. Kegiatan yang dilakukan tim pada pembangunan perangkat lunak mencakup penyimpanan versi kode program, pengujian kode program, eksekusi *build* dan pengintegrasian modul. Perangkat lunak yang dibangun adalah aplikasi rekam medis yang disebut *medrecapp* berbasis *Java desktop*.

Pada pembangunan aplikasi *medrecapp*, penyimpanan versi kode program yang dilakukan oleh setiap *developer* dilakukan dengan menduplikasi modul sebelum mengubah modul tersebut. Pada pengujian kode program, para *developer* menguji kode program dengan cara manual dan berulang kali. Setelah kode program diuji, tim tersebut memerlukan seorang *integrator* untuk mengintegrasikan modul dari para *developer* dan membuat paket aplikasi yang berisi *file* siap pakai. Sebelum *integrator* membuat paket aplikasi, *integrator* perlu menguji integrasi modul tersebut. Pengintegrasian modul yang dilakukan oleh *integrator* memerlukan *effort* yang besar karena ketika terjadi kesalahan pada satu atau lebih hasil pengujian sebelum pembuatan paket aplikasi, *integrator* perlu membuat informasi kesalahan dan menginformasikannya kepada setiap *developer*. Selain itu, *integrator* perlu mengarsipkan paket aplikasi hasil eksekusi *build* untuk mendapatkan *history* dari paket aplikasi di mesin integrasi dan membuat *report* kemajuan proses pembangunan aplikasi *medrecapp* untuk dijadikan sebagai *milestone*.

Praktik manual tersebut memerlukan *effort* yang besar dan rawan terhadap kesalahan. Oleh karena itu, kegiatan manual dan berulang pada praktik *continuous integration* perlu diotomasi. Untuk mengotomasi proses pada praktik *continuous integration* tersebut diperlukan *tool* dari praktik *automated continuous integration*. Tim yang mengimplementasikan praktik *automated continuous integration* perlu menggunakan *tool* dari tiga praktik lain yaitu, *version control system*, *automated testing* dan *automated build*.

Dengan menggunakan *version control system tool*, tim tidak lagi menduplikasi modul untuk menyimpan versi dari modul yang sudah diubah. Selain itu, dengan menggunakan *automated testing tool*, pengujian kode program dapat dilakukan secara berulang kali tanpa

mengeluarkan *effort* yang besar. Rangkaian proses pada pembuatan paket aplikasi yang berisi *file* siap pakai dapat dilakukan *integrator* dengan menggunakan *automated build tool* sehingga *integrator* hanya melakukan satu kali eksekusi proses. Dengan menggunakan *automated continuous integration tool*, tim dapat menjadwalkan pengintegrasian modul dan pembuatan paket aplikasi di mesin integrasi sehingga ketergantungan tim terhadap seorang *integrator* dapat dihilangkan. Selain itu, *tool* tersebut dapat memberikan notifikasi kesalahan, mengarsipkan paket aplikasi dan membuat report kemajuan proses pembangunan aplikasi medrecapp secara otomatis.

Kata kunci: penyimpanan versi kode, *compile*, *testing*, *packaging*, *build*, *Automated Continuous Integration* (CI)