Gambar 3-7. Eksekusi *build* dengan cara manual

|  |
| --- |
| Penjelasan:  *Developer* Y mengeksekusi proses *build* tanpa menggunakan *automated build tool*. *Developer* Y mengeksekusi proses *build* setelah pengujian unit dan integrasi pada integrasi modul telah menampilkan hasil yang benar. Semua eksekusi proses *build* dilakukan secara manual. Pada proses *build*, Setiap eksekusi proses bergantung pada proses sebelumnya. |

Gambar 3-8. Pemberian notifikasi kesalahan secara manual oleh *integrator*

|  |
| --- |
| Penjelasan:  *Developer* Y mengintegrasikan modul pada mesin integrasi tanpa bantuan *automated* *continuous* *integration* *tool*. *Developer* Y perlu melengkapi modul A dan modul C dari *developer* X dan Z. Setelah modul tersebut lengkap dan benar, *developer* Y akan menguji hasil integrasi modul A dan C. Ketika ada kesalahan pada satu atau lebih hasil pengujian, maka *developer* Y perlu menginformasikan kesalahan tersebut kepada *developer* yang lain untuk segera diperbaiki. |

Gambar 3.9**.**  Pengarsipan paket aplikasi secara manual oleh *integrator*

|  |
| --- |
| Penjelasan:  *Developer* Y mengarsipkan paket aplikasi pada mesin integrasi tanpa bantuan *automated* *continuous* *integration* *tool*. Ketika *integrator* telah melakukan pengujian integrasi dan memastikan semua pengujian telah menampilkan hasil yang benar, *integrator* mengeksekusi proses *build* secara manual. *Integrator* akan mengarsipkan paket aplikasi tersebut secara manual pada mesin integrasi untuk mendapatkan *history* dari paket aplikasi. |

Gambar 3-10. Pembuatan *report* kemajuan proses pembangunan perangkat lunak oleh *integrator*

|  |
| --- |
| Penjelasan:  *Developer* Y akan menginformasikan kemajuan proses pembangunan perangkat lunak tanpa menggunakan bantuan *automated* *continuous* *integration* *tool. Developer* Y akan membuat *report* kemajuan perangkat lunak secara manual. Setelah itu, *developer* Y akan memberikan *report* tersebut kepada setiap *developer*. |

Gambar 3-19. Eksekusi *build* pada mesin lokal

|  |
| --- |
| Penjelasan:  *Developer* X membuat modul A beserta kode pengujian unit dan integrasi modul A di *repository* lokal. *Developer* X akan mengeksekusi *build* *script* pada mesin lokal menggunakan bantuan *automated build tool*. Setelah *developer* X membuat modul A beserta kode pengujian unit dan integrasi. *Developer* X akan memanggil *target* pada *build* *script* melalui *automated build tool*. Kemudian *automated build* akan mengeksekusi proses *build* tersebut secara otomatis. Umumnya *build script* yang digunakan oleh setiap *developer* adalah sama. Penyamaan *target* dilakukan tim untuk menyamakan alur kerja dari setiap *developer*. |

Gambar 3-20. Eksekusi *build* pada mesin integrasi

|  |
| --- |
| Penjelasan:  Keseluruhan kode program pada modul A, B dan C beserta kode pengujian unit dan integrasi dari setiap modul yang ada di setiap *repository* tersebut disimpan dalam *repository* pusat. Mesin integrasi akan memanggil *target* pada *build script* melalui *automated build tool*. *Automated* *build* akan mengeksekusi proses tersebut secara otomatis. Sebaiknya, *build script* pada mesin lokal dibuat terpisah dengan *build script* pada mesin integrasi, kerena adanya kemungkinan anggota tim menggunakan IDE yang berbeda. |

**Gambar 3-21.** *Deploy* paket aplikasi ke *environment customer* dengan bantuan *tool* dari *automated build*

|  |
| --- |
| Penjelasan:  Keseluruhan kode program pada modul A, B dan C beserta kode pengujian unit dan integrasi dari setiap modul yang ada di setiap *repository* tersebut disimpan dalam *repository* pusat. Mesin integrasi akan memanggil *target* pada *build script* melalui *automated build tool*. *Automated* *build* akan mengeksekusi proses tersebut secara otomatis. Paket aplikasi dari eksekusi *build script* akan di-*deploy* ke *customer* *environment*. |

Gambar 3-22. Penjadwalan eksekusi *build script* pada mesin integrasi

|  |
| --- |
| Penjelasan:  *Automated CI tool* mengeksekusi *integration* *build* dan *release build* pada mesin integrasi berdasarkan penjadwalan. Penjadwalan dapat dilakukan setiap malam atau setiap terjadi perubahan pada *repository* pusat. |

Gambar 3-23. Notifikasi kesalahan secara otomatis dari mesin integrasi

|  |
| --- |
| Penjelasan:  Keseluruhan modul yang telah dilengkapi oleh setiap *developer* pada setiap *repository* lokal akan disimpan di dalam *repository* pusat oleh anggota tim. *Automated* *CI tool* akan mengeksekusi *integration* *build*. Ketika *integration* *build* yang dieksekusi oleh *automated* CI *tool* gagal, maka hasil notifikasi kesalahan dari *integration build* akan dikirimkan *automated CI tool* ke anggota tim. |

Gambar 3-24. Pengarsipan paket aplikasi oleh mesin integrasi secara otomatis

|  |
| --- |
| Penjelasan:  *Automated CI* *tool* akan memanggil *target* pada *build script* melalui *automated* *build* *tool*. Kemudian *automated build tool* akan mengeksekusi proses *build* secara otomatis. Paket aplikasi yang dihasilkan dari eksekusi *build script* akan diarsipkan oleh *automated build tool* dan disimpan dalam mesin integrasi. |

Gambar 3-25. *Report* kemajuan proses pembangunan perangkat lunak secara otomatis

|  |
| --- |
| Penjelasan:  *Atomated CI tool* menghasilkan *report* kemajuan proses pembangunan perangkat lunak. *Report* tersebut akan disimpan pada mesin integrasi. Setiap *developer* dapat memantau kemajuan proses pembangunan lunak tersebut. |