

Bab 8

Pengujian Tahap Program

Kompleksitas pelaksanaan tahap pemrograman kepada ketelitian pada tahap perancangan. Spesifikasi rancangan yang terukur dan terdefinisi dengan baik (well defined) sangat menyederhanakan pekerjaan pemrograman.

Pengujian selama tahap pemrograman dapat dilakukan secara statis maupun dinamis. Pada sebagian dari tahap ini kode program bias tidak dieksekusi, jadi dilaksanakan menggunakan tool uji lain.

8.1. Deliverable Tahap Pemrograman

Tahap program mencakup tiga bagian. Pertama, spesifikasi program ditulis dari spesifikasi rancangan. Kedua, pemrogram mengubah spesifikasi program menjadi instruksi-instruksi yang dapat dijalankan di mesin (computer). Ketiga, pemrogram memeriksa apakah instruksi-instruksi tersebut telah bekerja sesuai spesifikasi program.

Tahap pemrograman dalam pembangunan sistem menghasilkan volume “deliverable” yang besar. Maka, adalah sangat penting mengetahui deliverable, resiko-resikonya, dan bagian-bagian dari deliverable yang perlu diuji. Secara umum deliverable tahap pemrograman terdiri atas :

- Spesifikasi program
- Dokumentasi program
- Listing program computer
- Executable program
- Flowchart program
- Instruksi operator
- Hasil uji pemrograman

8.2. Deliverable Tahap Pemrograman

Karena besarnya ukuran deliverable tahap program, maka adalah sangat penting mengkonsentrasikan cakupan utama selama tahap pengujian. Penguji memilih cakupan dari kepentingan utama dan kemudian memilih tool yang sesuai dengan kepentingannya.

Cakupan uji yang perlu dipikirkan selama tahap pengujian adalah sbb :

Kendali integritas data terimplementasi

Kendali khusus perlu diimplementasikan dengan tujuan mencapai ketelitian integritas yang diinginkan.

Aturan otorisasi terimplementasi

Aturan otorisasi perlu diimplementasi dengan jalan membuatnya sulit untuk mengelak dari aturan otorisasi tersebut. Sebagai contoh, jika batas otorisasi telah ditentukan, maka orang tidak bias mengelak dari batas tersebut.

Kendali otorisasi terimplementasi

Kendali integritas file diimplementasi dengan tujuan untuk meminimalkan kebolehjadian (probability) dari hilangnya integritas file. Kendali semestinya mencegah hilangnya integritas dan mendeteksi hilangnya integritas yang terjadi.

Implementasi Audit Trail

Audit trail perlu diimplementasi dengan tujuan menyediakan cara retrieve informasi audit trail.

Menulis Rencana Kontingensi

Rencana kontingensi adalah set prosedur detail yang menjelaskan tahap-tahap yang harus dijalankan pada saat terjadinya masalah. Rencana hendaknya menjelaskan tahap persiapan sehingga data dan sumber daya lain yang sesuai tersedia pada saat di mana rencana kontingensi perlu diaktifkan.

Merancang sistem untuk mencapai tingkat layanan

Implementasi prosedur keamanan

Keamanan adalah kombinasi dari kesadaran dan latihan pekerja ditambah teknik dan perangkat keamanan yang diimplementasikan. Prosedur yang menjamin dua bagian ini diimplementasikan dan kerja bersama perlu dikembangkan selama tahap program.

Program menuruti metodologi

Prosedur-prosedur hendaknya diimplementasi sehingga terjamin sesuai standar, kebijaksanaan, prosedur dan metoda yang berkembang. Jika ketidaksesuaian ditemukan, maka ukuran yang cocok diambil untuk mendapatkan variansi dari metoda atau untuk mengubah sistem atau merancang sehingga kesesuaian tercapai.

Program untuk memenuhi rancangan (correctness)

Kondisi yang berubah menyebabkan personil proyek pengolahan data mengabaikan tujuan proyek selama tahap program. Tim uji harus memonitor implementasi dari tujuan.

Program memenuhi rancangan (case of use)

Implementasi spec sistem mungkin mengabaikan beberapa aspek kemudahan penggunaan dari rancangan. Pemrograman adalah terjemahan dari spesifikasi rancangan yang mungkin gagal untuk mencapai kemudahan penggunaan tersebut. Pemrogram harus mencapai kemudahan penggunaan, di samping spesifikasi fungsional yang lain.

Program dapat dirawat (maintainable)

Metoda rancangan program dan pengkodean mempunyai signifikansi yang lebih besar pada maintainabilitas daripada mengerjakan spesifikasi rancangan itu sendiri. Aturan dari kode yang dapat dirawat harus secara terpisah ditentukan oleh standar departemental, dan secara terpisah oleh spesifikasi sistem. Sebagai tambahan, pemrogram hendaknya menggunakan judgement dan pengalaman dalam membangun kode yang dapat dirawat.

Program menurut rancangan (portable)

Portabilitas program bergantung pada bahasa yang dipilih, dan bagaimana bahasa itu digunakan. Spesifikasi hendaknya menunjukkan hal-hal yang dikerjakan (“do”) dan hal-hal yang tidak dikerjakan (“don’t”) oleh pemrograman untuk portabilitas.

Program menurut rancangan (coupling)

Spesifikasi rancangan hendaknya menunjukkan parameter passing ke dan dari sistem aplikasi lain.

Membangun prosedur operasi

Prosedur hendaknya dibuat selama tahap pemrograman untuk mengoperasikan sistem aplikasi. Prosedur operasi hendaknya konsisten dengan kebutuhan operasional sistem aplikasi.

Program mencapai kriteria (performance)

Pembuatan program memberikan kesempatan operasional yang pertama kepada user untuk menilai apakah sistem dapat atau tidak memenuhi tingkat kinerja yang diinginkan. Pada poin ini instruksi untuk memenuhi kebutuhan sudah ditentukan dan dapat dievaluasi.

8.3. Deliverable Tahap Pemrograman

Pemrograman adalah fungsi pengolahan data murni. User sedikit terlibat selama tahap ini, kecuali pertanyaan-pertanyaan berkenaan dengan spesifikasi dan atau kebutuhan rancangan. Perhatian yang terus ditujukan kepada perubahan permintaan user (user-requested) selama fase ini. Jika perubahan tidak dapat ditunda, maka itu hendaknya dikerjakan kembali (rework) lewat tahap proses dan prioritas perubahan spesifikasi program yang mungkin diuji.

Ketua proyek pengolahan data hendaknya bertanggung jawab untuk pengujian selama tahap pemrograman. Tujuan utama dari pengujian ini adalah untuk menjamin bahwa spesifikasi rancangan telah diimplementasikan dengan benar. Pengujian program tidak tertuju kepada pencapaian kebutuhan user, tetapi pada struktur yang dibangun yang memenuhi spesifikasi rancangan dan bekerja.

Kebanyakan pengujian dipandu oleh pemrogram. Pengujian pada tahap ini bersifat teknis dan umumnya membutuhkan seorang dengan pengalaman pemrograman. Pengujian hendaknya lengkap terutama pada interkoneksi aplikasi dan uji sistem aplikasi.

8.4. Perangkat Uji Tahap Program yang direkomendasi

Terdapat dua teknik atau perangkat uji, yang secara dekat berkaitan, yaitu Desk Debugging dan Peer Review. Desk Debugging dilakukan oleh pemrogram perorangan, sedangkan Peer Review dilakukan oleh anggota-anggota lain departemen pengolahan data. Dua perangkat ini saling melengkapi.

Desk Debugging

Konsep desk debugging membolehkan pemrogram untuk menilai kelengkapan dan kebenaran program, terutama secara dinamik. Pada proses ini pemrogram berusaha menemukan dan mengoreksi cacat.

Desk debugging dapat dilakukan seekstensif atau seminimal yang diinginkan. Kebanyakan desk debugging dilaksanakan bergantung pada hal-hal sbb :

- Waktu tunggu (wait time) sampai deliverable program selanjutnya diterima
- Jadwal Implementasi
- Sumber daya pengujian
- Efisiensi perangkat uji
- Kebijakan departemen

Desk debugging dapat dibagi menjadi tiga jenis, yaitu :

1. Desk Debugging Sintaksis (Syntactical Desk Debugging)
Pemrogram memeriksa sintaks dokumentasi dan pernyataan (statement) yang sesuai untuk menjamin hal tersebut ditulis menurut aturan.
2. Desk Checking Struktural (Structural Desk Checking)
Masalah structural menghitung sejumlah cacat yang signifikan dalam system aplikasi. Cacat ini juga menutupi cacat fungsional sehingga pendeteksian makin berat.
3. Desk Debugging Fungsional (Functional Desk Debugging)
Fungsi-fungsi adalah kebutuhan sehingga program dapat berjalan.

Peer Review

Peer review adalah review informal tetapi efektif terhadap fungsi program computer. Secara normal review dilakukan oleh kawan (peer) dalam lingkungan yang tidak “mengancam” (monthreatening).

Peer Review meliputi analisis statis terhadap program yang menilai baik struktur maupun fungsi program. Dimungkinkan bahwa peer review menemukan salah sintak tetapi itu lebih disebabkan karena pengamatan pribadi bukan hasil langsung dari “walk through”.

Peer Review dapat dilakukan secara formal maupun informal. Cara formal merupakan langkah terpadu di dalam proses pemrograman.

Tim Review mempunyai anggota-anggota sbb:

- Pemrogram komputer
- Spesialis kendali kerja (job control specialist)
- Karyawan kendali (control clerk)
- Supervisor pemrogram

Peer Review dilaksanakan melalui tahap-tahap berikut ini :

1. Penetapan aturan ground review, meliputi laporan, metoda seleksi ketua tim, lokasi, dan metoda untuk memandu review.
2. Memilih tim review.

3. Melatih anggota tim review.
4. Memilih metoda review. Beberapa metoda untuk memandu review dapat dipakai seperti : flowchart, source code, transaksi contoh, dan spesifikasi program.
5. Memimpin dan memandu (conduct) review
6. Menggambarkan kesimpulan
7. Menyiapkan laporan

8.5. Proses Uji Tahap Program

Kedalaman pengujian bergantung pada kecukupan system pada akhir perancangan. Semakin yakin tim penguji akan kecukupan system, semakin sedikit cakupan yang diuji dalam proses pengujian, jadi akan meminimalkan pengujian.

Terdapat beberapa objektif yang secara terus-menerus harus dipikirkan oleh tim uji selama pengujian, yaitu :

- Apakah system maintainable
- Apakah spesifikasi system telah diimplementasi secara benar/layak
- Apakah program telah memenuhi standard an prosedur pengolahan data sebaik prakteknya
- Apakah terdapat rencana uji yang cukup untuk menilai executable program
- Apakah program telah didokumentasi secara cukup

Proses uji secara rinci mengikuti sembilan belas cakupan uji program. Proses ini meliputi criteria uji, proses uji yang direkomendasi, teknik dan perangkat. Selanjutnya dapat dilihat pada table dari hal 217-233 buku William Perry.

Referensi : Perry, W.E; A Structured Approach to Systems Testing; QED Information Science; 1983