



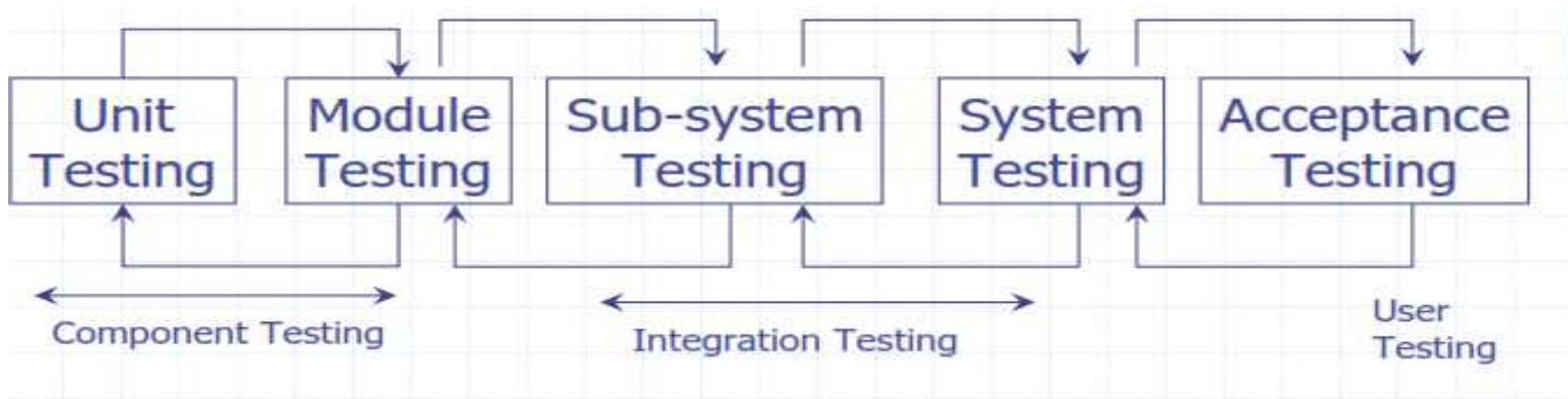
PENGUJIAN / TESTING

Definisi

- ▶ Proses eksekusi suatu program dengan maksud menemukan kesalahan.
- ▶ Sebuah ujicoba kasus yang baik adalah yang memiliki probabilitas yang tinggi dalam menemukan kesalahan-kesalahan yang belum terungkap.
- ▶ Ujicoba yang berhasil adalah yang mengungkap kesalahan yang belum ditemukan.



Proses Testing



Lanj..

- ▶ **System Testing**
Pengujian terhadap integrasi sub-system, yaitu keterhubungan antar sub-system.
- ▶ **Acceptance Testing**
Pengujian terakhir sebelum sistem dipakai oleh user.
Melibatkan pengujian dengan data dari pengguna sistem.
- ▶ **Component Testing**
Pengujian komponen-komponen program
Biasanya dilakukan oleh component developer (kecuali utk system kritis)
- ▶ **Integration Testing**
Pengujian kelompok komponen-komponen yang terintegrasi untuk membentuk sub-system ataupun system
Dilakukan oleh tim penguji yang independent
Pengujian berdasarkan spesifikasi sistem



Rencana Pengujian

- ▶ Proses testing
- ▶ Deskripsi fase-fase utama dalam pengujian
- ▶ Pelacakan Kebutuhan
- ▶ Semua kebutuhan user diuji secara individu
- ▶ Item yg diuji
- ▶ Menspesifikasi komponen sistem yang diuji
- ▶ Jadwal Testing Prosedur Pencatatan Hasil dan Prosedur
- ▶ Kebutuhan akan Hardware dan Software Kendala-kendala
- ▶ Mis: kekurangan staff, alat, waktu dll.



Teknik-Teknik Pengujian

- ▶ *White box testing*
- ▶ **Black-box testing**



White box testing

- ▶ *White box testing* didasarkan pada pemeriksaan detail prosedural.
- ▶ Alur logikal suatu software diujicoba dengan menyediakan kasus ujicoba yang melakukan sekumpulan kondisi dan/atau perulangan tertentu.
- ▶ Status dari program dapat diperiksa pada beberapa titik yang bervariasi untuk menentukan apakah status yang diharapkan atau ditegaskan sesuai dengan status sesungguhnya.



Black-box testing

- ▶ Ujicoba yang dilakukan pada interface software.
- ▶ Walaupun didesain untuk menemukan kesalahan, ujicoba *blackbox* digunakan untuk mendemonstrasikan fungsi software yang dioperasikan; apakah input diterima dengan benar, dan ouput yang dihasilkan benar; apakah integritas informasi eksternal terpelihara.
- ▶ Ujicoba *blackbox* memeriksa beberapa aspek sistem, tetapi memeriksa sedikit mengenai struktur logikal internal software.



Pengujian (testing) sistem informasi

- ▶ Testing saat Input Data

Tindakan untuk menguji edit dan kontrol dalam pemasukan data, misalnya : validasi, cek digit.

- ▶ Testing saat Pemrosesan

Bertujuan untuk meyakinkan bahwa program telah bekerja seperti yang diharapkan.

- ▶ Testing saat Output.

Testing saat Output berguna untuk meyakinkan bahwa laporan yang dihasilkan telah dibuat dengan format yang benar dan mempunyai informasi yang valid.



Pengontrolan saat input data, yaitu :

- ▶ **Character checks (pengecekan karakter)**
Suatu tindakan untuk melihat apakah suatu field itu dapat menerima karakter tertentu saja atau tidak.
 - ▶ **Numeric Value Checks (Pengecekan Nilai Numerik)**
Ini merupakan character checks yang dikhususkan pada karakter bilangan.
 - ▶ **Check Digit (Digit cek)**
Sejumlah angka mempunyai check digit, maka sistem akan menolak sembarang masukan yang banyaknya angka (digit) tidak akurat.
 - ▶ **Limit Tests (Pengujian Batas)**
Pada suatu field, kadang-kadang nilai yang harus diisikan terbatas jangkauan nilainya.
-



Lanj..

- ▶ **Reasonableness Tests (Pengujian Kelogisan)**

Seperti pada limit test, tetapi pembatasannya pada hal yang logis (beralasan).

- ▶ **Internal Compatibility (Kompatibilitas Internal)**

Suatu data yang sudah dimasukkan, sebaiknya kompatibel dengan data yang lain dalam satu aplikasi tertentu.

- ▶ **Cross Checks with data in other applications**

Suatu data akan diperiksa secara silang dengan aplikasi yang lain, sehingga jika terjadi kesalahan maka pesan kesalahan akan diberikan. Ini bermaksud untuk memeriksa apakah fungsinya telah berjalan dengan benar.

- ▶ **Duplicate Transactions (Transaksi Ganda)**

Suatu system sebaiknya dibuat dapat menolak transaksi yang ganda.



Lanj..

- ▶ **Table Look Ups**

jika suatu kode tertentu dimasukkan dalam suatu field, maka system akan mengakses table yang tepat dan memberikan informasi yang benar.

- ▶ **Existence of Required Data (Keberadaan Data yg Dibutuh)**

Jika suatu data yang dibutuhkan tidak ada, maka system harus memberikan pesan bahwa data yang dibutuhkan tidak ada. Jadi harus ada kejelasan tentang data.

- ▶ **Confirmation Screens (Layar Konfirmasi)**

Suatu tampilan yang akan memberikan konfirmasi bahwa data yang dimasukkan adalah data yang benar.

- ▶ **Field lengths and Overflow checks (Cek panjang field dan overflow)**

Panjang field dapat diberikan dengan ukuran tertentu. Begitu juga dengan banyaknya data yang dapat dimasukkan dalam suatu field.



TESTING SAAT PEMROSESAN DATA

- ▶ **Delete vs Reverse (Hapus lawan Mundur)**

Kita harus tahu bahwa data yang telah dimasukkan, nantinya akan dapat dihapus atau ditelusuri mundur (historisnya). Ketika suatu transaksi telah diupdate (dimutakhirkan), transaksi itu mungkin perlu dihapus atau ditelusuri ke belakang (mundur)

- ▶ **Automatically Triggered Processing (Pemrosesan Terpicu Secara Otomatis)**

Jika suatu system mempunyai sifat automatically triggered processing, maka testing yang dilakukan harusnya dapat meyakinkan bahwa kalkulasi atau pemrosesan telah dilakukan dengan benar. Dalam hal ini testing dilakukan untuk meyakinkan bahwa parameter yang tepat telah digunakan dan output dari pemrosesan telah akurat.



Lanj..

- ▶ **Updating (Pemutakhiran)**

Testing dijalankan untuk meyakinkan bahwa sistem telah di-update secara benar dengan data yang telah dimasukkan.

- ▶ **Audit Trails (Jejak Pemeriksaan)**

Seperti pada nomor 3, maka testing dilakukan untuk meyakinkan bahwa log system dan jejak untuk pemeriksaan telah bekerja dengan baik.

- ▶ **Table Values (Nilai Tabel)**

Testing ini dijalankan pada prosedur untuk peng-updatean parameter system dan tabel kode. Pelaksanaan dari update transaksi harus bekerja dengan tepat, termasuk dalam hal ini adalah pengeditan pemasukan data.



Lanj..

- ▶ **Arithmetic Calculations (Kalkulasi Aritmetika)**

Tes ini ditujukan kepada semua kalkulasi aritmetika yang telah dilakukan sudah benar sesuai rumus yang diinginkan. Untuk meyakinkan report telah benar, maka jika perlu ada cek silang dengan kalkulasi yang telah dilakukan.

- ▶ **Database Management System Testing**

Struktur database juga harus diuji untuk meyakinkan bahwa disain telah dibuat dengan benar.



TESTING OUTPUT

- ▶ Ringkasan laporan dapat diperiksa sebagai hal untuk meyakinkan bahwa format dan isi dari laporan sesuai yang dibutuhkan.
- ▶ Yakinlah bahwa dalam laporan hal-hal yang bersifat aritmetika telah berjalan dengan benar.
- ▶ Sebaiknya seluruh laporan harus disajikan dan lihatlah ada kesalahan atau tidak dengan laporan sebelumnya.
- ▶ Jika ada laporan yang bersifat khusus, maka testing juga harus dapat menjawab bahwa laporan khusus ini ditujukan untuk meyakinkan bahwa data yang diekstrak dari suatu tempat harus cocok dan lengkap dengan kriteria khusus yang diharapkan.



TUGAS 2

- ▶ Cari paper tentang pembuatan sistem informasi apa saja.
Topik : bebas
- ▶ Analisis paper tsb berdasarkan testing atau pengujian yg dilakukan.
- ▶ Buat dalam bentuk cetakan (cover, paper dan hasil analisis) dan powerpoint



Terima Kasih

