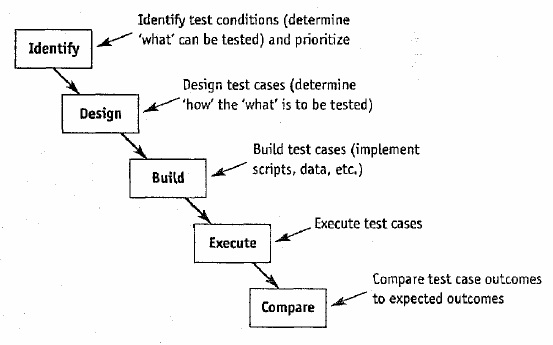
**Siklus Pengujian Perangkat Lunak**

Pengujian perangkat lunak adalah proses atau rangkaian proses yang dirancang untuk memastikan bahwa program computer menjalankan apa yang seharusnya dilakukan dan sebaliknya, memastikan program agar tidak melakukan hal yang tidak diharapkan.

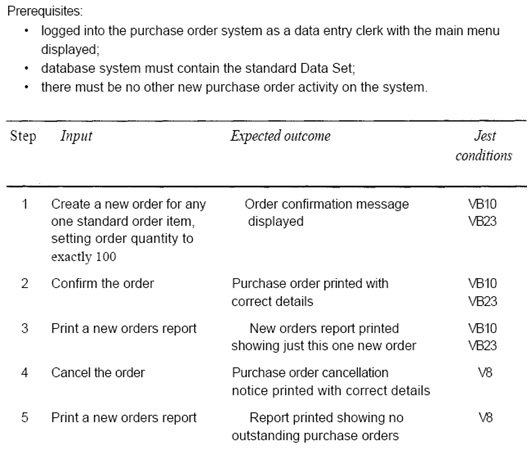
Sebuah perangkat lunak seharusnya dapat diprediksi dan konsisten. Tugas utama dari seorang penguji adalah untuk menemukan bug atau error sebanyak mungkin serta mengetahui bagaimana error atau bug itu dihasilkan. Sehingga dapat dikatakan bahwa pengujian adalah proses mengeksekusi program dengan tujuan untuk menemukan error. Pengujian merupakan proses penting dalam siklus pengembangan software untuk memastikan kualitas dari sebuah software.

Proses pengujian terdiri dari beberapa aktifitas yang biasa disebut sebagai siklus hidup pengujian. Pada beberapa praktek, masing-masing aktifitas dapat dilakukan secara formal atau informal. Gambar di bawa ini menunjukkan aktifitas pengujian berdasarkan Fewster.



**Gambar 1 Aktifitas Pengujian**

1. Mengidentifikasikan kondisi pengujian merupakan aktifitas pengujian yang pertama untuk menentukan apa yang bisa diuji dan memberikan prioritas terhadap kondisi-kondisi pengujian yang ada. Kondisi pengujian adalah suatu hal atau peristiwa yang dapat diverifikasi melalui proses pengujian. Terdapat berbagai jenis kondisi pengujian dari suatu sistem, sesuai dengan kategori dari pengujian yang dilakukan, apakah pengujian fungsional, pengujian keamanan, dst.
2. Aktifitas kedua adalah perancangan test case yang bertujuan untuk menentukan bagaimana pengujian akan dilakukan. Sebuah **test case** adalah sekumpulan dari pengujian yang dilakukan secara sekuensial dan terkait dengan tujuan dari pengujian. Perancangan test case akan menghasilkan sejumlah percobaan yang terdiri dari nilai inputan, hasil yang diharapkan, serta informasi lain yang dibutuhkan untuk menjalankan pengujian seperti kondisi prasyarat pengujian. Berikut ini adalah contoh dari test case pada kasus pembelian (order) barang.



**Gambar 2. Contoh Test Case**

1. Aktifitas selanjutnya adalah membangun test case, yang dapat dibagi menjadi dua kategori; secara manual atau otomatis. Pembuatan test case secara manual akan menghasilkan sebuah prosedur pengujian, yang merupakan langkah menjalankan pengujian dengan informasi test input/test data yang dibutuhkan beserta output yang diharapkan. Sedangkan dalam pengujian otomatis, test case dibuat dalam bentuk test script. **Test Script** adalah data dan/atau instruksi yang berupa syntax formal yang dapat dieksekusi oleh sebuah alat mengujian otomatis. Sebuah test script dapat mengimplementasikan satu atau lebih test case, navigasi, prosedur persiapan dan pembersihan serta verifikasi.
2. Setelah test script/prosedur dibuat dan dilengkapi dengan data input dan output yang dibutuhkan, maka pengujian dapat dilakukan. Pada eksekusi pengujian secara manual, penguji mengikuti prosedur yang telah dibuat dengan kombinasi input dan melakukan pengecekan terhadap output. Hasil pengujian serta catatan observasi lainnya dicatat secara manual. Sedangkan pada pengujian secara otomatis, konfigurasi terhadap alat pengujian harus dilakukan terlebih dahulu, kemudian test script dipilih dan dijalankan secara otomatis. Pencatatan hasil pengujian juga dilakukan secara otomatis, namun formatnya akan berbeda sesuai masing-masing alat pengujian.
3. Aktifitas terakhir dari pengujian adalah melakukan perbandingan terhadap hasil pengujian dengan hasil yang diharapkan. Proses perbandingan dapat dilakukan ketika proses pengujian berlangsung, seperti mengecek pesan yang tampil pada layar, dan ketika pengujian telah selesai dilakukan, dengan membandingkan perubahan data yang ada. Apabila hasil pengujian dan hasil yang diharapkan sama, maka dapat disimpulkan bahwa software lolos proses pengujian. Sebaliknya, jika hasil yang diharapkan tidak sesuai, maka perlu dilakukan perbaikan software dan pengujian akan diulang setelah perbaikan dilakukan.

Sesuai dengan konsep iterasi pada metodologi agile, proses pengujian dilakukan langsung setiap satu product increment selesai dikembangkan. Sehingga untuk satu produk perangkat lunak, akan dilakukan beberapa kali pengujian sesuai dengan jumlah iterasi yang dimiliki. Masing-masing iterasi akan membutuhkan pengujian integrasi untuk menggabungkan feature-feature yang dibangun.

Keterlibatan customer dalam pengembangan agile juga dilakukan pada proses pengujian dengan melakukan acceptance testing. Acceptance testing diformulasikan dari user story atau acceptance criteria yang telah disetujui antara customer dan tim pengembang pada tahap awal iterasi. Untuk memfasilitasi kebutuhan feedback yang cepat, otomasi pengujian merupakan hal dasar yang harus diterapkan untuk mengefisiensikan proses pengujian.

Referensi: Crispin, L., & Gregory, J. (2009). *Agile testing: A practical guide for testers and agile teams*. Pearson Education . Written By : *Meiliana, S.Kom., M.Sc.* Concentration Content Coordinator - Software Engineering | School of Computer Science