

**TUGAS KAPITA SELEKTA**  
**PEMAHAMAN 7 PRINSIP TESTING, FUNCTIONAL TESTING,**  
**NON FUNCTIONAL TESTING, STRUCTURAL TESTING DAN**  
**TESTING RELATED TO CHANGE**

Disusun untuk Memenuhi Tugas Kapita Seleкта



Disusun oleh :

**GUNA DHARMA (1204071)**

**D4 Teknik Informatika 3C**

**SEKOLAH VOKASI PROGRAM D4 TEKNIK INFORMATIKA**  
**UNIVERSITAS LOGISTIK DAN BISNIS INTERNASIONAL**  
**TAHUN AJARAN 2022/2023**  
**BANDUNG**

#### A. 7 Prinsip Testing :

1. **Testing Shows Presence of Defects** => Pengujian ini Menunjukkan Adanya sebuah kecacatan, tetapi tidak dapat membuktikan ketika tidak adanya sebuah kecacatan. Pengujian harus dirancang untuk menemukan sebanyak mungkin cacat.
2. **Exhaustive Testing is Impossible** => Pada Mengujian ini semuanya (semua kombinasi input dan prasyarat) tidak dapat dilakukan kecuali untuk kasus-kasus yang sepele. Alihalih pengujian yang menyeluruh, analisis risiko dan prioritas harus digunakan untuk memfokuskan upaya pengujian.
3. **Early Testing** => Dalam Software Development Life Cycle (SDLC) kegiatan pengujian harus dimulai sedini mungkin dan harus difokuskan pada tujuan yang telah ditetapkan.
4. **Defect Clustering** => Sejumlah kecil modul berisi sebagian besar cacat ditemukan selama pengujian pra-rilis atau menunjukkan kegagalan yang paling operasional. Hal ini dapat memudahkan tester dalam mengidentifikasi.
5. **Pesticide Paradox** => Jika jenis yang sama dari tes diulang lagi dan lagi, akhirnya set yang sama dari kasus uji tidak akan lagi dapat menemukan bug baru. Untuk mengatasi hal ini “Pesticide Paradox”, itu benar-benar sangat penting untuk meninjau kasus uji secara teratur dan tes baru dan berbeda harus ditulis untuk latihan bagian yang berbeda dari perangkat lunak atau sistem untuk berpotensi menemukan lebih banyak cacat.
6. **Testing is context different** => Pengujian pada dasarnya tergantung konteks. Berbagai jenis situs yang diuji berbeda. Misalnya, Testing software yang berkaitan erat dengan keamanan tentu berbeda dari sebuah situs e-commerce.
7. **Absence-of-errors-fallacy** => Jika sistem yang dibangun tidak dapat digunakan dan tidak memenuhi kebutuhan dan harapan pengguna kemudian ditemukan dan memperbaiki cacat/bugs. hal ini tidak akan berguna sama sekali.

#### B. Functional Testing

Functional testing lebih berfokus pada hasil eksekusi dari proses setiap fitur di aplikasi sudah bekerja dengan baik berdasarkan kebutuhan klien secara menyeluruh. Functional testing dilakukan sebelum melakukan non-functional testing. Pemodelan dengan UML, ataupun penjelasan fitur-fitur dalam bentuk problem statements, adalah

termasuk dalam Functional Requirement.

### **C. NonFunctional Testing**

Non-functional testing merupakan pengujian mengenai cara kerja sistem, Nonfunctional testing lebih kepada meningkatkan kualitas sebuah aplikasi seperti bergantung pada Waktu, Akurasi, Stabilitas, Kebenaran dan Durability di berbagai macam keadaan.

### **D. Structural Testing**

Structural Testing adalah menentukan sejauh mana suatu struktur yang telah dilakukan dalam pengujian, penentuan ini dinyatakan dalam persentase dari item yang dinilai. Jika coverage tidak 100%, maka pengujian lebih lanjut mungkin dirancang untuk menguji itemitem yang terlewatkan untuk meningkatkan coverage.

### **E. Testing Related To Change**

1. **Confirmation testing or retesting (Pengujian konfirmasi atau pengujian ulang)**, setelah cacat terdeteksi dan diperbaiki, perangkat lunak harus diuji ulang untuk mengonfirmasi bahwa cacat asli telah berhasil dihilangkan.
2. **Regression testing (Pengujian regresi)**, adalah pengujian berulang dari program yang sudah diuji, setelah modifikasi, untuk menemukan cacat yang diperkenalkan atau ditemukan sebagai hasil dari perubahan.