

**PENGUJIAN PERANGKAT LUNAK
BLACK BOX TESTING**



**OLEH :
KELOMPOK 4 – TI 3B**

Aldyansyah Putra Arifiyanto – 1841720127

Muhammad Hifzhan Silmi – 1841720159

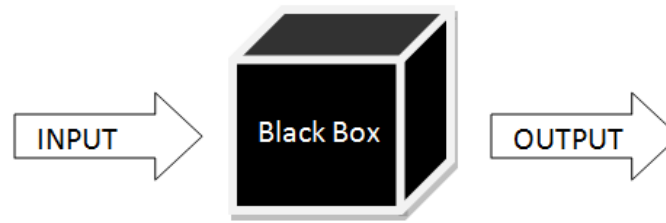
Mardhiyah Millania – 1841720081

Wahyu Setyapamungkas – 1841720009

Gilang wahyu hidayat - 1841720172

**PROGRAM STUDI D-IV TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
SEPTEMBER 2020**

1. Definisi dan cara kerja dari Black Box



Black Box Testing Disebut juga pengujian behavioral/Pengujian Partisi/Pengujian Interface merupakan metode pengujian Perangkat Lunak yang digunakan untuk menguji perangkat lunak tanpa mengetahui struktur internal kode atau Program.

Dalam pengujian ini, tester menyadari apa yang harus dilakukan oleh program tetapi tidak memiliki pengetahuan tentang bagaimana melakukannya.

2. Pada tingkatan apa pendekatan pengujian perangkat lunak dapat dilakukan? (ex: unit,integration,functional)

Metode uji blackbox dapat diterapkan pada semua tingkat pengujian perangkat lunak: unit, integrasi, fungsional, sistem dan penerimaan. Ini biasanya terdiri dari kebanyakan jika tidak semua pengujian pada tingkat yang lebih tinggi, tetapi juga bisa mendominasi unit testing juga. Pengujian Perangkat Lunak dengan Black Box Testing memiliki teknik antara lain:

1. Equivalence Class Partitioning (Pembagian Kelas yang sama)
2. Boundary Value Analysis (Analisa Penilaian Terbatas)
3. Comparison Testing (Pengujian Perbandingan)

Pendekatan pengujian blackbox bisa dilakukan dari fungsional perangkat lunak, non fungsional perangkat lunak serta regression.

1. Pengujian fungsional perangkat lunak

Pengujian fungsional adalah jenis pengujian yang memastikan bahwa produk perangkat lunak bekerja sesuai dengan persyaratan. Dalam pengujian ini, kami menguji semua fungsionalitas produk. Penguji dapat memberikan input yang sesuai dan memeriksa hasilnya. Kemudian dia dapat membandingkan hasil aktual dengan hasil yang diharapkan

Misalnya, asumsikan perangkat lunak seperti sistem manajemen pusat medis. Sistem ini memiliki sejumlah fungsi seperti menambah, memperbarui, menghapus rincian pasien, menambah, memperbarui menghapus rincian dokter, laporan umum, mencetak tagihan, dll. Memeriksa apakah masing-masing fungsi berfungsi seperti yang diharapkan adalah pengujian fungsional.

2. Pengujian non fungsional perangkat lunak

Pengujian nonfungsional adalah jenis pengujian yang memeriksa aspek nonfungsional dari sistem. Ini berfokus pada berbagai aspek seperti kinerja, kegunaan, keandalan dll. Dengan kata lain, ini menguji parameter yang tidak diuji dalam pengujian fungsional.

Misalnya memeriksa bagaimana sistem bekerja untuk beban kerja yang diantisipasi, memeriksa sistem untuk beban kerja yang ekstrem, memeriksa bagaimana perangkat lunak menangani beban kerja yang diharapkan untuk suatu periode beban kerja mereka dan memeriksa bagaimana sistem bekerja pada beban mendadak

3. Pengujian Regresi

Pengujian Regresi dilakukan setelah perbaikan kode, peningkatan atau pemeliharaan sistem lainnya untuk memeriksa kode baru tidak memengaruhi kode yang ada.

Misalnya ada salah satu fitur yang diperbaiki kodingannya ataupun ditingkatkan fiturnya, jadi blackbox akan memeriksa apakah fitur yang lain tetap bekerja dengan fungsinya atau ada kode baru yang mempengaruhi kerja fitur lain

3. Mengapa harus dilakukan pengujian dengan menggunakan pendekatan tersebut?

Tujuan dari pengujian black box adalah memastikan setiap bagian sudah sesuai dengan alur proses yang ditentukan, serta memastikan semua kesalahan masukan yang digunakan oleh user dapat ditangani oleh sistem. Pengujian secara black box untuk memastikan fungsi-fungsi yang ada pada aplikasi berjalan dengan semestinya. Pengujian ini lebih ke *User Interface* dari suatu aplikasi, agar lebih mudah dioperasikan.

4. Apa kemampuan yang dibutuhkan oleh seorang tester untuk melakukan pendekatan tersebut?

Dalam pengujian black box, seorang tester harus bisa menyadari apa yang harus dilakukan program tetapi tidak memiliki pengetahuan tentang bagaimana melakukannya. Jadi ibaratnya seorang tester adalah *Hacker*, mereka harus bisa mencari celah pada suatu sistem. Tester harus familiar dengan alat pemindaian otomatis dan metodologi pentest manual. Tester juga harus memiliki kemampuan *mapping* dari sistem yang diuji berdasarkan observasi yang telah dilakukan. Kemampuan seorang tester bisa berdampak terhadap durasi pengujian, jika seorang tester kurang baik maka kerentanan sistem belum bisa ditemukan dan diperbaiki.

5. Pada tahapan pengembangan perangkat lunak apa, pendekatan pengujian tersebut dapat dilakukan?

Metode uji dapat diterapkan pada semua tingkat pengujian perangkat lunak: unit, integrasi, fungsional, sistem dan penerimaan. Ini biasanya terdiri dari kebanyakan jika tidak semua pengujian pada tingkat yang lebih tinggi, tetapi juga bisa mendominasi unit testing juga.

Pengujian pada Black Box berusaha menemukan kesalahan seperti:

- Fungsi-fungsi yang tidak benar atau hilang
- Kesalahan interface
- Kesalahan dalam struktur data atau akses database eksternal
- Kesalahan kinerja
- Inisialisasi dan kesalahan terminasi

6. Tools apa saja yang dapat digunakan oleh tester

Tools Black-box Testing

- a) Equivalence Partitioning
Cara kerja teknik ini adalah dengan melakukan partition atau pembagian menjadi beberapa partisi dari input data.
- b) Boundary Value Analysis
Teknik ini lebih fokus kepada boundary, dimana adakah error dari luar atau sisi dalam software, minimum, maupun maximum nilai dari error yang didapat.
- c) Fuzzing
Fuzz merupakan teknik untuk mencari bug / gangguan dari software dengan menggunakan injeksi data yang terbilang cacat ataupun sesi semi-otomatis.
- d) Cause-Effect Graph
Ini adalah teknik testing dimana menggunakan graphic sebagai pacuannya. Dimana dalam grafik ini menggambarkan relasi diantara efek dan penyebab dari error tersebut.
- e) Orthogonal Array Testong
Dapat digunakan jika input domain yang relatif terbilang kecil ukurannya, tetapi cukup berat untuk digunakan dalam skala besar.
- f) All Pair Testing
Dalam teknik ini, semua pasangan dari test case di desain sedemikian rupa agar dapat di eksekusi semua kemungkinan kombinasi diskrit dari seluruh pasangan berdasar input parameteranya. Tujuannya testing ini adalah memiliki pasangan test case yang mencakup semua pasangan tersebut.

g) State Transition

Testing ini berguna untuk melakukan pengetesan terhadap kondisi dari mesin dan navigasi dari UI dalam bentuk grafik.

7. Apakah ketiga pendekatan pengujian perangkat lunak dapat digunakan secara bersamaan pada proses pengembangan perangkat lunak? Alasannya?

- pengembangan perangkat lunak dapat beragam sesuai paradigma atau model proses. Paradigma yang dipilih untuk pengembangan perangkat lunak adalah paradigma konvensional. Sedangkan model proses yang akan digunakan untuk pengembangan perangkat lunak ada beberapa macam.

Sumber :

1. <https://medium.com/skyshidigital/pengujian-sistem-52940ee98c77>
2. <https://id.strephonsays.com/what-is-the-difference-between-functional-and-nonfunctional-testing>
3. [PPL]+Week+2.pdf
4. <https://www.guru99.com/black-box-testing.html>
5. <https://www.mahadisuta.com/artikel/pengujian-white-box-dan-black-box-wajib-dilakukan-seorang-developer>
6. <https://socs.binus.ac.id/2020/07/02/teknik-dalam-white-box-dan-black-box-testing/>
7. <https://medium.com/skyshidigital/pengujian-sistem-52940ee98c77>
8. <https://www.logique.co.id/blog/2019/07/17/metode-pentest/>
9. <http://tkjpnp.blogspot.com/2013/12/black-box-testing-dan-white-box-testing.html#:~:text=Black%20Box%20pengujian%20adalah%20metode,pemrograman%20pada%20umumnya%20tidak%20diperlukan.>
10. <https://socs.binus.ac.id/2020/07/02/teknik-dalam-white-box-dan-black-box-testing/>