Nama : Qaulan Sakila Usman

Npm : 6101121004

Kelas : B

**TESTING DAN IMPLEMENTASI SISTEM**

Testing atau pengujian adalah proses evaluasi perangkat lunak atau sistem untuk menentukan apakah itu berfungsi dengan benar sesuai dengan persyaratan atau spesifikasi yang ditetapkan.

Berikut adalah beberapa definisi mengenai testing atau pengujian menurut para ahli:

* Dr. Glenford Myers

"Testing atau pengujian adalah proses mengevaluasi suatu sistem atau komponen dengan tujuan menentukan apakah itu memenuhi persyaratan tertentu atau tidak."

* Roger S. Pressman

"Pengujian atau testing adalah proses verifikasi dan validasi untuk menentukan apakah perangkat lunak yang dikembangkan sesuai dengan persyaratan bisnis yang telah ditetapkan dan sesuai dengan harapan pengguna."

* Dr. Magdy Hanna

"Testing atau pengujian adalah proses menguji sistem atau perangkat lunak untuk menemukan kesalahan dan kekurangan dengan maksud untuk memastikan kualitas dan kinerja yang diharapkan."

* Boris Beizer

"Pengujian atau testing adalah proses menjalankan program atau sistem dengan maksud untuk menemukan kesalahan atau cacat yang ada dalam program atau sistem."

Contoh testing adalah ketika seorang pengembang perangkat lunak melakukan pengujian terhadap sebuah aplikasi yang telah dibuat. Misalnya, pengembang ingin memastikan bahwa semua fitur aplikasi berfungsi dengan baik dan tidak ada bug yang dapat mengganggu pengguna. Pengujian dilakukan dengan mengisi formulir, mengklik tombol, memasukkan data, dan melihat hasilnya. Selain itu, pengujian juga dapat dilakukan dengan menggunakan berbagai situasi atau skenario yang mungkin terjadi saat pengguna menggunakan aplikasi. Tujuan dari testing adalah untuk memastikan bahwa aplikasi telah siap digunakan oleh pengguna.

Berikut adalah beberapa faktor kualitas secara umum yang dapat mempengaruhi kualitas sebuah software, yang diambil dari beberapa sumber:

1. Fungsionalitas: Kemampuan perangkat lunak untuk menyediakan fungsi sesuai kebutuhan user dan memuaskan user.

* Kebenaran (Correctness) : Sebuah software akan berkualitas jika software yang dibuat sesuai dengan spesifikasi yang dibutuhkan. Sehingga seorang programmer dalam membuat sebuah software jangan lepas atau keluar dari lingkup spesifikasi yang sudah ditetapkan karena spesifikasi yang sudah ditetapkan merupakan kebutuhan yang diinginkan.
* Reliability : Aplikasi yang dibuat diharapkan jangan sampai mengalami kegagalan pada saat digunakan. Hal ini penting agar fungsi dari aplikasi dibuat dapat diterapkan dan keinginan dari pengguna dapat di akomodasi di dalam software yang dibuat.
* Kegunaan (Usability) : Terkadang pengguna dari software adalah orang yang sudah lama menerapkan sistem secara manual dan pada saat sistem komputerisasi diterapkan maka pengguna tersebut harus belajar untuk memahami software yang dipakai. Oleh karena itu, penting seorang programmer memperhatikan karakter dari pengguna yang akan menggunakan aplikasi yang dibuat. Sehingga dari segi desain harus dibuat lebih mudah untuk digunakan atau bisa dikatakan "Jangan membuat software yang gampang bagi programmer sendiri tapi sulit bagi pengguna yang akan memakainya". Jadi jangan sampai hal itu terjadi.
* Integritas (Integrity) : Sebuah sistem di dalam software terdiri dari data yang penting, oleh karena itu penting pada saat membuat software untuk memperhatikan segi keamanan dari software agar tidak ada penyalahgunaan dari pihak yang tidak berwenang terhadap aplikasi.

1. Rekayasa: Berkaitan dengan sifat-sifat teknis dalam pengembangan software, seperti resource yang rendah, kecepatan, multi-platform, multi-bahasa, multi-user, dan tampilan intuitif

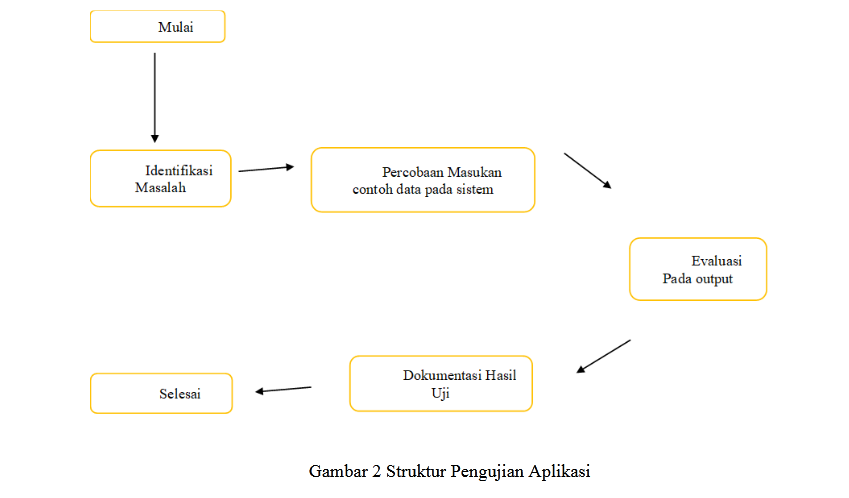
* Efisiensi (Efficiency) : Pada saat proses mengkoding program, seorang programmer berusaha agar kode program yang diterapkan pada aplikasi lebih efisien. Dimana diusahakan bila ada kode program yang sama, jangan dituliskan atau menggunakan procedure. Dan dari segi operasional jangan sampai pengguna merasa kerepotan pada saat menggunakan aplikasi.
* Testability : Sebuah software jika tidak dilakukan pengujian pada saat proses pengembangannya, maka tidak bisa diketahui apakah software sudah benar atau belum. Dan hal ini menjadi tolak ukur bahwa software sudah teruji dan sesuai persyaratannya.
* Dokumentasi : Seorang programmer harus terbiasa mendokumentasikan hasil dari pekerjaannya agar pada saat terjadi sesuatu hal yang tidak diinginkan masih ada histori dari proses koding. Dan juga membiasakan membuat blueprint dari software yang akan dibuat. Sehingga dengan terbiasa mengikuti dokumen yang dibuat, maka software yang dihasilkan pun sama dengan dokumen yang ada.
* Struktur : Seorang programmer harus terbiasa untuk menerapkan pola rekayasa yang baik. Menurut saya pola yang baik terdiri dari tahapan analisa sistem, desain atau perancangan, koding atau implementasi, testing dan pemeliharaan.

1. Adaptabilitas: Kemampuan software dalam menjalani perubahan dan penyesuaian software dalam lingkungan yang baru.

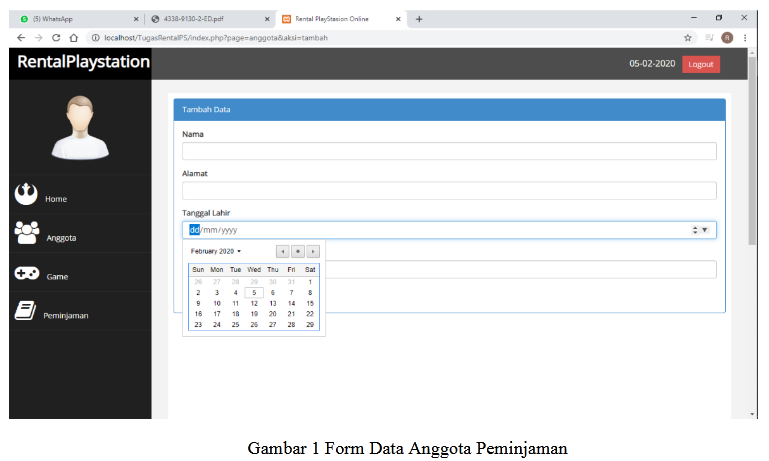
* Fleksibilitas (Flexibility) : Sebuah software diharapkan dapat menyesuaikan dengan kebutuhan pengguna jika dikemudian hari ada pengembangan lanjutan. Oleh karena itu, software harus bisa dimodifikasi untuk disesuaikan dengan kebutuhan pengguna dan itu harus diterapkan pada aplikasi.
* Reusability : Aplikasi yang dibuat diharapkan dapat digunakan untuk pengembangan aplikasi lain. Dalam artian bahwa software dapat dipakai jika dikemudian hari terdapat pengembangan software lain dan tinggal mengintegrasikan dengan software yang baru.
* Maintainabillity : Sebuah software harus lebih mudah untuk dilakukan perbaikan jika terjadi kerusakan pada aplikasi. Hal ini menjadikan software semakin lama semakin teruji dengan adanya perbaikan yang terus menerus dan lebih mudah untuk langsung diperbaiki jika terdapat kerusakan.

**ILUSTRASI PENGUJIAN SISTEM**

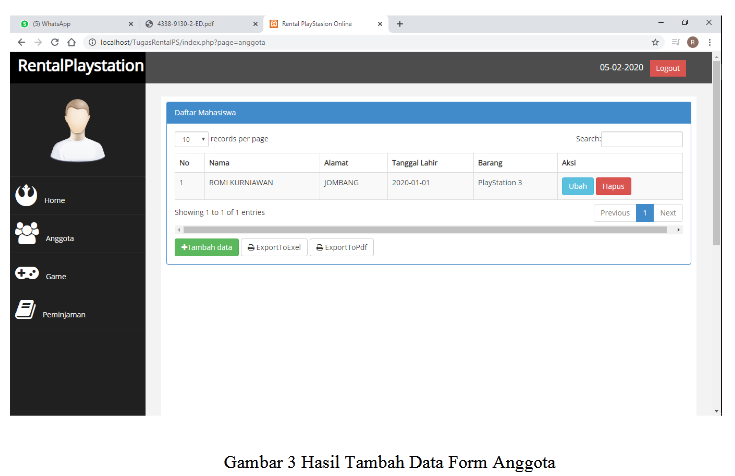
Ilustrasi ini diambil dari jurnal berjudul Pengujian Perangkat Lunak Sistem Informasi Peminjaman PlayStation dengan Teknik Boundary Value Analysis Menggunakan Metode Black Box Testing sebagai berikut.



Gambar 1. Struktur Pengujian Aplikasi

Pada pengujian ini terdapat beberapa tahapan, yang dimana tahapan awal setelah memulai testing sebuah aplikasi yaitu. Pengidentifikasian masalah (proses penentuan kekurangan dari sebuah aplikasi). Percobaan masukan (melakukan memasukan data/ memasukan data). Evaluasi pada hasil Output (penaksiran atau penilaian dari hasil memasukan data). Dokumentasi hasil uji (proses pendataan dari semua rangkaian ).

Gambar 2. Form Data Anggota Peminjaman

Dalam form ini ada beberapa rencana pengujian, Kolom Nama dapat diisi dengan nama Peminjam, Kolom Alamat dapat diisi juga dengan Alamat dari nama peminjam, Kolom Tanggal Lahir juga wajib diisi sesuai dengan Tanggal Lahir yang meminjam, dan Kolom Barang dapat diisi dengan Barang yang ingin dipinjam, Jika terdapat kolom yang tidak diisi atau dikosongkan akan muncul pemberitahuan sistem bahwa terdapat kolom yang belum diisi.

Gambar 3. Hasil Tambah Data Form Anggota

Dalam Form ini adalah hasil dari penambahan data anggota yang diproses oleh user untuk data peminjaman yang terdiri dari, No, Nama, Alamat, Tanggal Lahir, Barang, dan ada Aksi untuk Mengubah data atau Menghapus data user Anggota Peminjaman.

Tabel 1. Field Siap Uji

Berdasarkan kolom tabel deskripsi pengujian dan hasil yang diharapkan, jika mengisi nama pada kolom nama dan mengisi alamat, misal : Jakarta. Dan isikan tanggal lahir, contoh: 19-09-1990. Kemudian isi kolom barang berdasarkan apa yang dipinjam dan tekan tombol simpan. Hasil yang diharapkan, sistem akan mengisi data yang sudah diisi pada form Anggota Pinjaman dan jika akurat statusnya akan Berhasil. Jika tidak akurat berarti data tidak berhasil disimpan.

**KESIMPULAN**

Testing atau pengujian adalah proses evaluasi perangkat lunak atau sistem untuk menentukan apakah itu berfungsi dengan benar sesuai dengan persyaratan atau spesifikasi yang ditetapkan. Contoh testing adalah ketika seorang pengembang perangkat lunak melakukan pengujian terhadap sebuah aplikasi yang telah dibuat. Misalnya, pengembang ingin memastikan bahwa semua fitur aplikasi berfungsi dengan baik dan tidak ada bug yang dapat mengganggu pengguna. Pengujian dilakukan dengan mengisi formulir, mengklik tombol, memasukkan data, dan melihat hasilnya.

beberapa faktor kualitas secara umum yang dapat mempengaruhi kualitas sebuah software, yang diambil dari beberapa sumber:

* Fungsionalitas: Kemampuan perangkat lunak untuk menyediakan fungsi sesuai kebutuhan user dan memuaskan user.
* Rekayasa: Berkaitan dengan sifat-sifat teknis dalam pengembangan software, seperti resource yang rendah, kecepatan, multi-platform, multi-bahasa, multi-user, dan tampilan intuitif
* Adaptabilitas: Kemampuan software dalam menjalani perubahan dan penyesuaian software dalam lingkungan yang baru.

Pengujian aplikasi yang menggunakan metode Black Box hanya berfokus terhadap masukan dan keluaran yang menginformasikan kesesuaian aplikasi yang dikembangkan dengan spesifikasi yang ditetapkan. Hasil pengujian menunjukan bahwa tingkat keberhasilan aplikasi hanya berjalan sekitar 70%. Sistem tetap mampumenghitung nilai dimana keadaan salah satu kolom dikosongkan, Aplikasi ini perlu diperbaiki lagi untuk menginput data Anggota Peminjaman dimana keadaan salah satu kolom dikosongkan.

**DAFTAR PUSTAKA**

Sis.binus.ac.id. 04 April 2019. Kualitas Software Model ISO 9126. Diakses pada 02 Oktober 2023, dari <https://sis.binus.ac.id/2019/04/04/kualitas-software-model-iso-9126/>

Coldeja.com. 05 September 2020. Faktor-Faktor Yang Mempengaruhi Kualitas Program Perangkat Lunak. Diakses pada 02 Oktober 2023, dari <https://www.coldeja.com/2020/09/faktor-kualitas-program-perangkat-lunak.html?m=1>

Banuacode.com. 28 September 2017. Faktor Kualitas Sebuah Software. Diakses pada 02 Oktober 2023, dari <https://www.banuacode.com/2017/09/faktor-kualitas-sebuah-software.html?m=1>

Idmetafora.com. 29 April 2023. Pentingnya Melakukan Testing: Apa Itu Testing? Pengertian, Jenis, Tujuan, Konsep, dan Tahapannya. Diakses pada 02 Oktober 2023, dari <https://idmetafora.com/id/blog/read/3326/Pentingnya-Melakukan-Testing-Apa-Itu-Testing-Pengertian-Jenis-Tujuan-Konsep-dan-Tahapannya.html>

Ahrizal, Dekif., dkk. “Pengujian Perangkat Lunak Sistem Informasi Peminjaman PlayStation dengan Teknik Boundary Value Analysis Menggunakan Metode Black Box Testing”. Jurnal Informatika 05, no. 1 Maret (2020): 73-77