**Nama : Rania Annisa**

**NIM : 1957301036**

**Kelas : TI 4B**

**Modul Software Quality Assurance QA**

**(Pada video 3-15) TESS**

1. **Software Development Life Cycle (SDLC)**

SDLC (Software Development Life Cycle) adalah kerangka kerja atau model manajemen proyek terstruktur yang menguraikan fase-fase yang diperlukan untuk membangun sistem TI, dari awal hingga hasil akhir. Tujuan dari Software Development Life Cycle (SDLC) ini untuk menciptakan proses produksi yang efektif dan berkualitas tinggi agar dapat memenuhi harapan klien sesuai dengan anggaran dan jadwal yang telah ditentukan.

Berikut penjelasan sederhana tentang SDLC, yang dianalogikan dengan pengalaman restoran. Ada pelanggan yang memesan sesuatu yang dinginkan, tapi cara bicara pelanggan dan cara memasak chef memiliki bahasa yang berbeda. Oleh karena itu, dibutuhkan penerjemah bahasa yang disebut sebagai pemilik produk. Dalam kasus restoran ini, dapur adalah pengembangan tim (development tim). Juru masak (chef) sebagai pengembang yang melakukan sesuatu yang mana chef ini yang dapat merasakan terlebih dahulu sesuatu yang mereka masak. Jadi, sebelum makanan dari chef tersebut di hidangkan ke pelanggan, maka hanya chef yang mengetahui rasa dan penampilan dari makanan tersebut yang sesuai dengan permintaan pelanggan. Apabila, pesanan pelanggan sudah sesuai dengan keinginannya dan kualitas cukup baik untuk dilayani, maka akan dihidangkan ke pelanggan. Berikut gambaran dari SDLC sederhana yang dianalogikan sebagai restoran:



1. **Testing Definition**

Untuk memastikan kualitas dari suatu aplikasi, testing harus dilakukan yaitu dengan cara menguji apakah sistem data yang dihasilkan sesuai dengan testing yang telah dilakukan. Tujuan dari testing ini sendiri antara lain, yaitu:

* Memastikan aplikasi berjalan sebagaimana mestinya
* Mendeteksi terjadinya error serta memvalidasi apakah sudah memenuhi keinginan user
* Melakukan pengecekan/pengetesan entitas-entitas, termasuk software,untuk pemenuhan dan konsistensi dengan melakukan evaluasi hasil terhadap kebutuhan yang telah ditetapkan
* Melakukan validasi untuk dapat melihat kebenaran sistem, apakah proses yang telah ditulis dalam spesifikasi adalah apa yang sebenarnya.
* Mendeteksi adanya error, testing untuk mendeteksi kesalahan secara insentif, yaitu menentukan apakah suatu hal tersebut terjadi bilamana tidak seharusnya terjadi.

Terdapat dua tahapan proses perangkat lunak bebas bug dan memenuhi semua persyaratan, yaitu:

1. Verifikasi

Verifikasi adalah  suatu pengecekan ataupun pengetesan entitas- entitas, termasuk aplikasi, untuk pemenuhan serta konsistensi dengan melakukan penilaian hasil terhadap kebutuhan yang telah ditetapkan.

“are we building the product right?” (menurut Boehm)

“apakah kita telah membuat produk dengan benar?”

1. Validasi

Validasi yaitu setelah melaksanakan proses verifikasi berikutnya adalah melihat kebenaran sistem, apakah proses yang ditulis dalam spesifikasi adalah apa yang sesungguhnya diinginkan ataupun dibutuhkan oleh pengguna.

“are we building the right product?” (menurut Boehm)

“apakah kita telah membuat produk yang benar?”

Berikut 5 kegiatan utama yang akan dibahas pada testing definition, yaitu:



1. Bug Hunting 🡪 aktivitas untuk mencari bug dan ketika bug sudah ditemukan akan dilaporkan.
2. Bug reporting 🡪 pemantauan dalam melacak status dan riwayat, yang mana memerlukan tempat pusat dalam proses pemantauan.
3. Prioritization 🡪 tidak semuanya akan diperbaiki, hanya memprioritaskan kekurangannya yang masuk akal saja.
4. Test cases 🡪 menulis, mengeksekusi, dan melaporkan kasus yang akan diuji.
5. Story testing 🡪 untuk melakukan beberapa pengujian terhadap persyaratan, sehingga dapat menemukan bug, melaporkan dan menulis kasus yang diuji.
6. **Software Defect (Bug)**

Kesalahan dalam program komputer yang menyebabkan hasilnya tidak benar atau tidak terduga, atau berperilaku dengan cara yang tidak diinginkan.

Berikut ini adalah beberapa jenis bug yang paling sering terjadi selama proses pengembangan software.

1. Functional error

Functional error merupakan sebuah kategori luas yang mencakup masalah yang terkait fungsionalitas sebuah program. Bug jenis ini bervariasi, mulai dari tombol yang tidak dapat di-klik hingga masalah pada kegunaan aplikasi itu sendiri.

1. Performance defects

Performance defects adalah kategori bug yang terkait dengan kecepatan, stabilitas, response time, dan penggunaan sumber daya dari sebuah software.

1. Usability defects

Usability defects adalah jenis bug yang menyebabkan pengguna tidak dapat memanfaatkan sebuah software secara maksimal. Jenis bug ini biasanya menyebabkan software sulit atau tidak nyaman untuk digunakan. Selain masalah pada kode software, usability defects juga dapat disebabkan oleh desain UI yang terlalu rumit sehingga pengguna kesulitan menemukan fungsi yang mereka cari.

1. Compatibility error

Compatibility error merujuk pada masalah software yang tidak dapat berjalan sebagaimana mestinya dalam situasi tertentu. Pada umumnya, compatibility error akan muncul ketika kamu menjalankan aplikasi lama pada sistem operasi yang lebih baru.

1. Security error

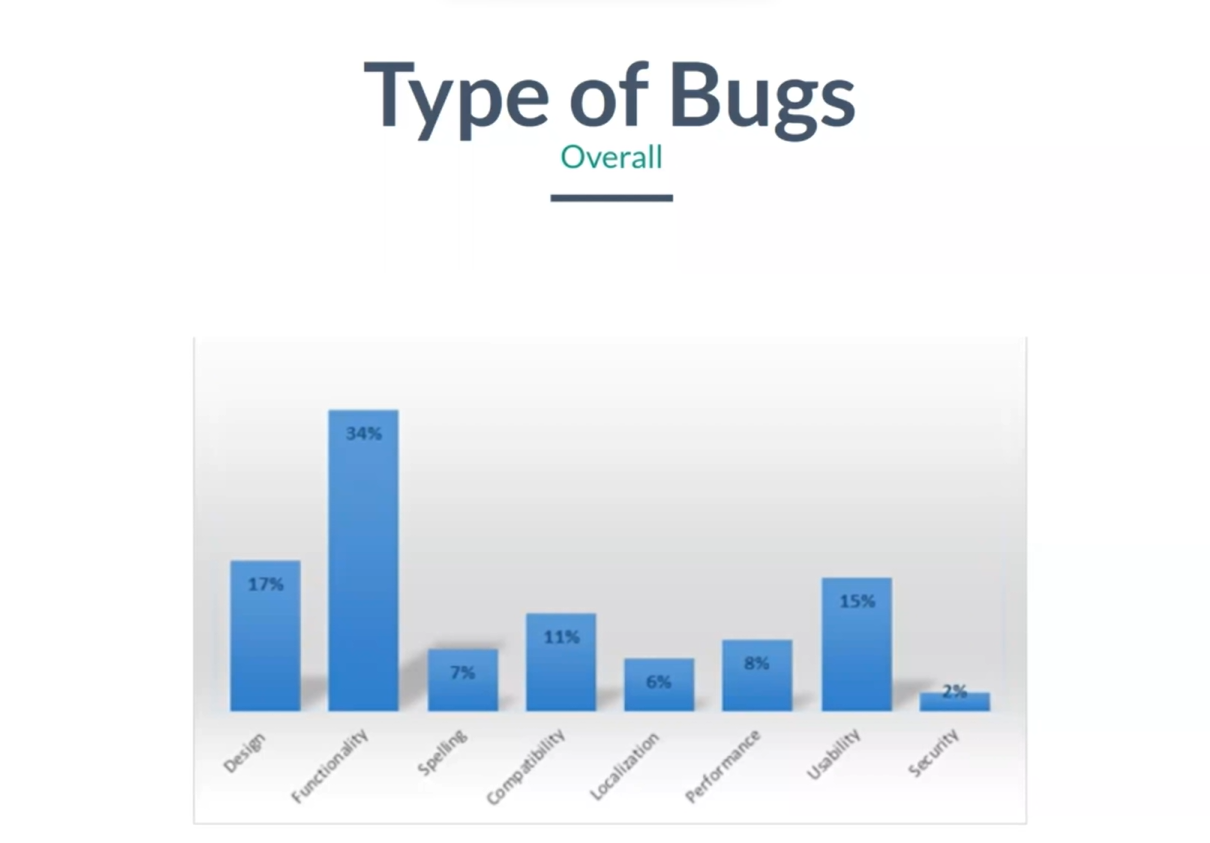
Security error merupakan jenis bug yang cukup berbahaya karena terkait langsung dengan sistem keamanan sebuah software. Sebuah bug dalam sistem keamanan software, besar atau kecil, akan membuka celah untuk cracking dan mencuri data atau merusak software tersebut.

1. Syntax error

Syntax error adalah jenis bug yang terjadi pada source code sebuah program. Jenis bug ini akan menghambat proses pengembangan software dan menyebabkan software tidak dapat di-compile dengan baik.

1. Logic error

Logic error adalah jenis bug lain yang terjadi pada source code sebuah program. Jenis bug ini biasanya disebabkan oleh penggunaan kode yang tidak sesuai sehingga menghasilkan output yang salah.



Pada grafik ini, functionality merupakan nomor satu dalam sebuah popularitas, yang mana 34 persen cacat.

1. **Tipe-tipe pada pengujian**
2. UI Testing

Kerusakan/ketidaksempurnaan pada UI/design dapat dikategorikan dalam 5 kategori, yaitu:

1. Layout

Kerusakan pada layout dapat terlihat pada ketidaksejajaran margin, ketidaksempurnaan model kotak IE6(Internet Explorer 6), overlapping, kerusakan gambar, jarak lebar dan tinggi garis, ruang yang tidak diingankan diantara daftar item, jarak vertikal yang memiliki kecacatan.

1. Font

Kerusakan pada font dapat terlihat dari tipe font yang berbeda, family font yang digunakan berbeda, tidak adanya keseragaman dalam ukuran font yang digunakan.

1. Color

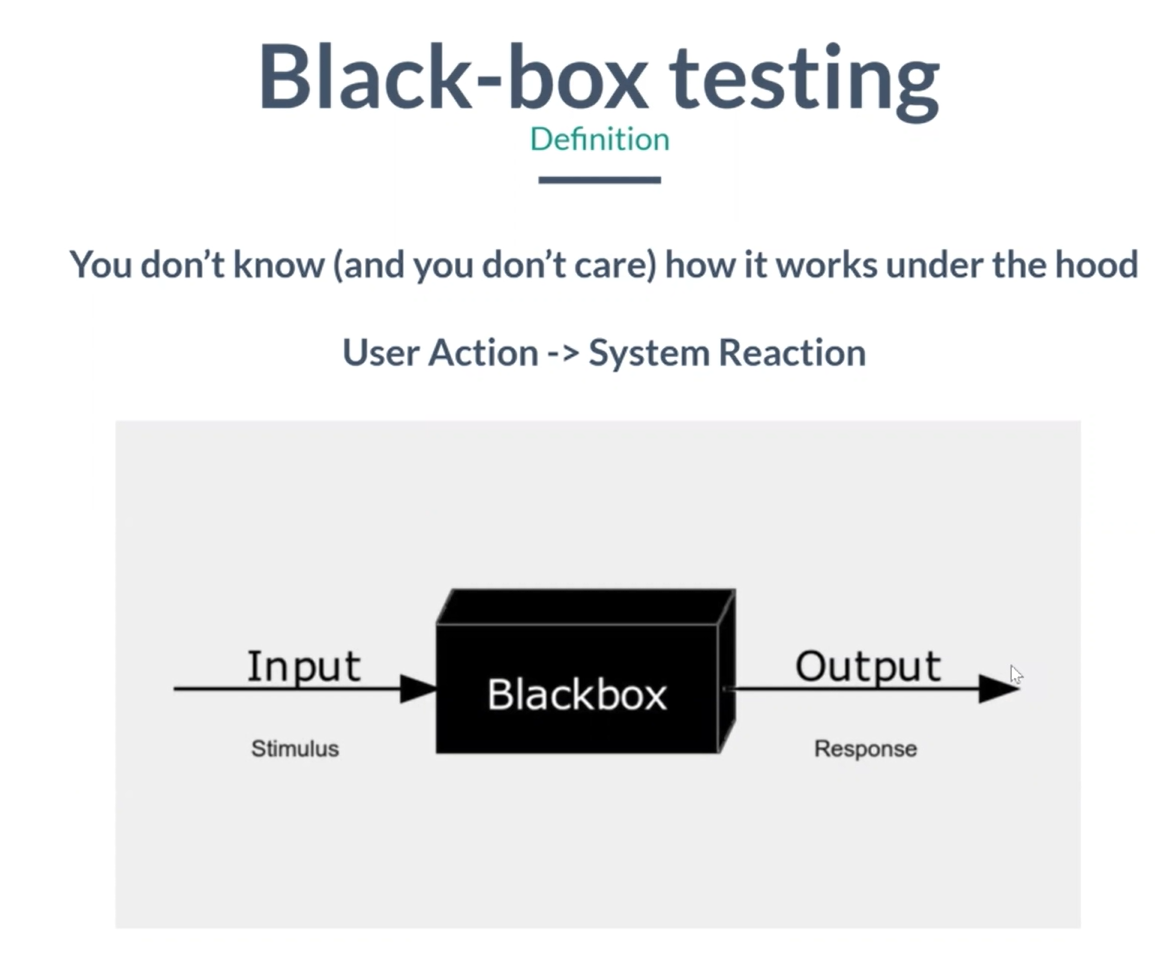
Kerusakan pada warna dapat terlihat dari tidak adanya keseragaman warna yang digunakan dan warna tombol seharusnya berubah saat dinonaktifkan tetapi tidak berfungsi.

1. Content

Kerusakan pada content dapat terlihat dari perbedaan besar kecilnya suatu ukuran font, tidak adanya keseragaman kata, kerusakan gambar dan kesalahan dalam ejaan kata.

1. Functional Testing

Fungsional testing adalah sistem yang diuji terhadap fungsional persyaratan atau spesifikasinya. Jenis pengujian ini juga memverifikasi bahwa setiap fungsi aplikasi telah beroperasi sesuai dengan kebutuhan, yang berfokus pada manual dan automation testing. Fungsional testing ini merupakan salah satu jenis pengujian metode black box testing.



Pengguna memberikan aksi sebagai input dan sistem memberikan reaksi/respon sebagai output. Black box testing hanya akan menjangkau input dan output sistem software tanpa adanya pengetahuan terkait internal program. Metode pengujian black box ini sangat penting agar bisa menemukan gangguan atau bug dalam suatu aplikasi sebelum dirilis secara resmi.

1. Compatibility Testing

Terdapat jenis pengujian lain yang disebut dengan pengujian kompatibilitas. Jenis pengujian perangkat lunak ini untuk memeriksa suatu aplikasi pada sistem operasi perangkat keras, jaringan, serta perangkat seluler (mobile) yang berbeda. Banyak kombinasi berbeda yang dimiliki pada setiap komponen sistem, contohnya komputer memiliki sistem operasi windows, mac dan sebagainya yang masing-masing memiliki fitur yang berbeda. Oleh karena itu, aplikasi yang dibuat pada dasarnya akan berfungsi dengan baik pada salah satu modifikasi tersebut. Salah satu contoh bug pada kompatibilitas ini yaitu pada saat mengerjakan beberapa proyek pengujian, ketika membuka website menggunakan Internet Explorer, hanya homepage halaman kosong berwarna putih yang tampil karena proyek tersebut tidak mendukung menggunakan Internet Explorer. Hal itu menjadi sebuah masalah yang harus diperbaiki secepatnya karena itu permasalahan yang sangat penting.

1. Localization Testing

Pengujian lokalisasi yaitu memeriksa aplikasi sesuai dengan setting lokal untuk menyesuaikan bahasa dan negara yang sesuai dengan keinginan pengguna. Oleh karena itu, format pada mata uang, tanggal dan waktu akan berubah yang sesuai dengan setting lokal yang dipilih oleh pengguna.

1. Performance Testing

Performance testing yaitu untuk menentukan kecepatan, daya tanggap, dan stabilitas program perangkat lunak. Pengujian ini sebenarnya berlaku untuk bisnis ritel online, yang mana banyak jumlah pelanggan dan pembelian yang dilakukan pada saat bersamaan, maka sangat penting bagi bisnis untuk benar-benar mengatasinya.

1. Usability testing

Usability testing adalah metode yang digunakan untuk mengevaluasi seberapa mudah situs web menggunakan User Experience (UX). Salah satu contoh usability box new away 2018, ketika pada dasarnya semua orang dari negara bagian mendapatkan pesan peringatan hanya karena kesalaham seseorang meng-klik sesuatu yang salah dikomputer. Namun, itu hanya sebuah kutipan, karena masalah penggunannya yang mana memiliki dua tombol yang sama disamping satu sama lain. Oleh karena itu, lebih baik untuk menggunakan warna yang berbeda agar dapat dibedakan antara satu tombol dengan tombol lainnya.

1. Security Testing

Testing technique yaitu untuk menemtukan sistem informasi dalam melindungi data dan memelihara fungsionalitas yang sesuai keinginan. Salah satu contohnya dari Apple, Januari 2019, 14 tahun, Arizona Boy menemukan kerusakan/cacat dalam Apple face time yang pada dasarnya, ketika menelpon seseorang pada perangkat Apple tanpa benar-benar memanggil panggilan. Dalam face time dapat mengakses ke kamera penerima dan mikrofon, yang secara tidak langsung pengguna dapat melihat siapa saja yang dikenal yang ada pada address book.

1. Ad-hoc or Monkey Testing

Ad-hoc testing yaitu salah satu cara untuk menemukan bug dalam pengujian yang mengacu pada beberapa tes yang pada dasarnya terdapat jenis pengujian yang hanya dilakukan pengujian secara acak. Setelah bug ditemukan, maka akan melakukan tangkapan layar yang mana cara dalam menangkap layar tergantung pada sistem yang digunakan. Berikut shortcut dalam tangkapan layar pada beberapa sistem, yaitu:



1. **Cara Melaporkan Bug**

Untuk melaporkan bug dapat menggunakan Jira, berikut 4 langkah utama dalam log bugs dalam Jira:



1. Type

Pilih jenis masalah bug, yang mana tergantung pada proyek yang telah disiapkan di JIRA. Jenis masalah dapat berisi nilai berikut: bug, story, epic, task dan sebagainya untuk melaporkan kerusakan.



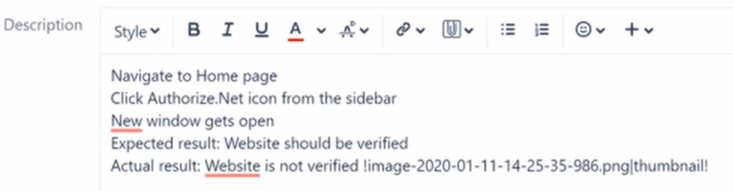
1. Summary

Menulis ringkasan singkat tentang apa dan dimana yang salah. Dapat ditulis di summary yaitu [Authorize.net] Verification is missing.



1. Description

Deskripsi ini adalah bidang yang paling penting ketika membuat laporan bug, yang mana deskripsi ini sedikit lebih deskriptif dari dari ringkasan (summary).



1. Screenshot

Screenshot jendela Authorize.Net untuk lebih terpercaya dan aman, maka dapat ditampilkan screenshot jendela Authorize.Net. Setelah itu, masukkan file image di attachment.

