

Nama: Zuhru Shofa Afia

Kelas: B PROA Kominfo

Kelompok: 2

1. Misalkan Anda adalah Quality Assurance Engineer yang bertugas untuk menguji dan memastikan kualitas sebuah aplikasi mobile perbankan yang baru dikembangkan oleh tim pengembangan perusahaan XYZ. Aplikasi ini bertujuan untuk memberikan layanan perbankan digital kepada nasabah, termasuk fitur-fitur seperti cek saldo, transfer dana, dan pembayaran tagihan. Jelaskan langkah-langkah dalam Software Testing Life Cycle (STLC) yang Anda terapkan dalam menguji dan memastikan kualitas aplikasi mobile perbankan ini!

Jawab:

Langkah-langkah yang diterapkan ketika menguji dan memastikan kualitas aplikasi mobile perbankan:

- a. Requirement analysis: pada tahapan ini penguji membuat beberapa persyaratan yang harus terpenuhi dalam aplikasi mobile perbankan seperti
 - a) Fitur cek saldo
Deskripsi: Aplikasi ini akan memungkinkan nasabah untuk memeriksa saldo rekening mereka
Kebutuhan: Aplikasi harus secara akurat menampilkan saldo rekening nasabah dan memberikan catatan transaksi terupdate.
Kriteria penguji: ketika nasabah memeriksa saldo, maka saldo yang ditampilkan harus akurat dari rekening nasabah
 - b) Fitur transfer dana
Deskripsi: Aplikasi ini akan memungkinkan nasabah untuk mengirim dana antar rekening.
Kebutuhan: Aplikasi harus dapat mengkonfirmasi data transaksi, memberikan opsi untuk memilih rekening asal dan tujuan, dan memberikan konfirmasi transaksi.
Kriteria penguji: ketika nasabah mengirim dana maka saldo dari rekening sumber harus berkurang secara akurat dan rekening tujuan akan bertambah secara akurat.
 - c) Fitur otentikasi dan keamanan
Deskripsi: Aplikasi ini harus dapat menjaga informasi nasabah.
Kebutuhan: Aplikasi ini harus menggunakan enkripsi data dan memerlukan otentikasi yang kuat berupa pin atau sidik jari.
Kriteria penguji: aplikasi harus dapat menjaga data nasabah dan memverifikasi bahwa otentikasi berjalan secara tepat.
 - d) Fitur pembayaran tagihan
Deskripsi: Aplikasi harus dapat menampilkan tagihan nasabah dan memungkinkan nasabah melakukan pembayaran
Kebutuhan: Aplikasi perlu memiliki kemampuan untuk mengakomodasi berbagai metode pembayaran untuk tagihan, menyimpan catatan histori pembayaran, dan menghasilkan bukti transaksi pembayaran.
Kriteria penguji: ketika nasabah melakukan pembayaran, transaksi harus diproses secara tepat dan harus dapat menghasilkan bukti transaksi nasabah.

b. Test Planning

Pada tahap perencanaan pengujian, tim QA mengembangkan rencana pengujian yang komprehensif. Ini mencakup identifikasi lingkup pengujian, sumber daya yang diperlukan, jadwal pelaksanaan, dan prioritas pengujian. Tim juga akan menentukan jenis pengujian yang akan dilakukan, seperti pengujian fungsional, pengujian non-fungsional (kinerja, keamanan, dll.), dan pengujian perangkat seluler. Dokumen rencana pengujian akan menjadi panduan selama seluruh proses pengujian.

c. Test development

Dalam tahap ini, tim QA akan mengembangkan skenario pengujian dan rincian kasus uji untuk setiap fitur aplikasi mobile perbankan. Ini mencakup membuat alur kerja (use cases) yang mencakup langkah-langkah spesifik yang akan diuji, data pengujian yang diperlukan, dan hasil yang diharapkan. Selain itu, tim QA akan menyiapkan semua bahan yang dibutuhkan untuk melaksanakan pengujian, seperti script pengujian, data pengujian, dan alat pengujian otomatis jika diperlukan.

d. Test environment setup

Pada tahap ini, tim QA akan menyiapkan lingkungan pengujian yang sesuai. Ini mencakup memastikan bahwa perangkat keras dan perangkat lunak yang diperlukan tersedia untuk menjalankan pengujian. Dalam konteks pengujian aplikasi mobile, ini juga melibatkan konfigurasi perangkat seluler virtual atau fisik untuk mencocokkan dengan berbagai platform dan perangkat yang digunakan oleh pengguna.

e. Test execution phase

Ini adalah tahap di mana pengujian sebenarnya dilakukan. Tim QA akan menjalankan kasus uji yang telah dikembangkan sebelumnya pada perangkat seluler, mengikuti alur kerja yang telah ditentukan, dan mencatat hasil dari setiap pengujian. Setiap bug atau masalah yang ditemukan selama pengujian akan dilaporkan dengan detail untuk diperbaiki oleh tim pengembangan.

f. Test cycle closure

Setelah semua pengujian selesai, tim QA akan melakukan evaluasi terhadap hasil pengujian. Mereka akan memeriksa apakah semua kasus uji telah dijalankan dengan sukses dan apakah semua masalah yang ditemukan telah diperbaiki. Selain itu, mereka akan menyusun laporan pengujian yang mencakup hasil, statistik pengujian, masalah yang ditemukan, dan rekomendasi untuk tindakan lebih lanjut. Tim pengembangan dapat menggunakan laporan ini sebagai panduan untuk perbaikan dan pembaruan aplikasi.

Langkah-langkah ini penting untuk memastikan bahwa aplikasi berfungsi dengan baik, aman, dan sesuai dengan kebutuhan pengguna.

2. Misalkan Anda adalah Quality Assurance Engineer. Anda diberikan tugas untuk melakukan tes sebuah program matematika sederhana yang akan menghitung akar pangkat dua dari suatu angka. Buatlah Behavior Driven Development (BDD) untuk menguji dan mengembangkan fungsi atau program ini.

Jawaban :

Dalam Behavior Driven Development (BDD), disini menggunakan bahasa alami untuk merinci perilaku yang diharapkan dari program matematika sederhana yang menghitung akar pangkat dua dari suatu angka.

- Feature: Menghitung Akar Pangkat Dua

Scenario: Menghitung akar pangkat dua dari angka positif

Given saya memasukkan angka 4

When saya menghitung akar pangkat dua

Then hasilnya harus menjadi 2

- Scenario: Menghitung akar pangkat dua dari angka negatif

Given saya memasukkan angka -9

When saya menghitung akar pangkat dua

Then program harus memberikan pesan kesalahan "Angka harus positif"

- Scenario: Menghitung akar pangkat dua dari angka desimal

Given saya memasukkan angka 2.25

When saya menghitung akar pangkat dua

Then hasilnya harus menjadi 1.5

- Scenario: Menghitung akar pangkat dua dari angka nol

Given saya memasukkan angka 0

When saya menghitung akar pangkat dua

Then hasilnya harus menjadi 0

Penjelasan Skenario:

1. Skenario pertama menguji perhitungan akar pangkat dua dari angka positif, di mana inputnya adalah 4, dan hasil yang diharapkan adalah 2.
2. Skenario kedua menguji input angka negatif (-9) dan memastikan bahwa program memberikan pesan kesalahan yang benar.

3. Skenario ketiga menguji input angka desimal (2.25) dan memverifikasi hasil perhitungan yang benar (1.5).
4. Skenario keempat menguji input angka nol (0) dan memastikan bahwa hasilnya adalah 0.

Skenario-skenario ini mencakup berbagai situasi yang mungkin terjadi saat menghitung akar pangkat dua dari suatu angka dan akan membantu dalam menguji dan mengembangkan fungsi atau program matematika sederhana ini dengan lebih terstruktur.