

NAMA : FENNI LESTARI RAJA GUK GUK

NIM : 190405019

PRODI/FAKULTAS: TEKNIK KIMIA/TEKNIK

TUGAS PRAKTIKUM SECTION 8 (AGILE TESTING MANIFESTO)

1. Pada fase apa saja QE berperan pada proses SDLC? Sebutkan dan jelaskan peran seperti apa yang QE dapat lakukan pada setiap fase tersebut!

Jawab:

SDLC merupakan sebuah proses yang digunakan untuk pengembangan sebuah sistem informasi. SDLC memecah setiap prosedur pengembangan sistem informasi menjadi beberapa fase seperti *Project Selection and Planning*, *Analysis*, *Design*, dan *Testing & Deployment*. Pemecahan setiap prosedur pada SDLC dimaksudkan untuk memudahkan baik *developer* ataupun *user* untuk melakukan *review* atau perbaikan pada fase-fase tertentu. SDLC merupakan sebuah proses yang lebih berfokus kepada bagian fungsionalitas dari sebuah sistem. Beberapa contoh model dari hasil evolusi SDLC adalah *waterfall model*, *iterative and incremental development*, *agile development*, *code and fix*, *spiral model*. SDLC itu sendiri dibagi menjadi empat fase yaitu: (1) *Planning and Selection*, (2) *System Analysis*, (3) *System Design*, dan (4) *System Implementation and Operation*. Model SDLC yang berfokus pada fungsionalitas sebuah sistem daripada kebutuhan penggunaannya menjadi salah satu kesulitan yang dialami suatu organisasi untuk memahami sistem yang digunakan.

Seorang *quality engineer* perlu mempunyai pengetahuan yang komprehensif mengenai proses tersebut. Karena apa yang akan dilakukan, terutama pada saat proses *testing* baik itu manual maupun automated diharapkan akan mempunyai pengaruh yang signifikan terhadap SDLC. QE mengambil peran dalam tahap *System Design* dan *System Implementation*.

Perancangan Sistem (Systems Design), yang mana pada tahap ini, features dan operasi-operasi pada sistem dideskripsikan secara detail. Aktivitas-aktivitas yang dilakukan adalah:

- Menganalisa interaksi obyek dan fungsi pada sistem.
- Menganalisa data dan membuat skema database.
- Merancang user interface.

Implementasi Sistem (Systems Implementation) yaitu mengimplementasikan rancangan dari tahap-tahap sebelumnya dan melakukan uji coba.

Dalam implementasi, QE melakukan aktivitas-aktivitas sebagai berikut:

- Pembuatan database sesuai skema rancangan.
- Pembuatan aplikasi berdasarkan desain sistem.
- Pengujian dan perbaikan aplikasi (debugging).

2. Menurut teman-teman apa tantangan ketika menerapkan agile testing? Berikan penjelasan masing-masing 5 poin agile testing manifesto!

Jawab:

Agile adalah kemampuan untuk membuat dan menanggapi perubahan dalam lingkungan yang tidak pasti. Agile Development Methods merupakan istilah untuk set kerangka kerja dalam pengembangan perangkat lunak yang didasarkan pada nilai-nilai dan prinsip-prinsip yang dinyatakan dalam Manifesto untuk pengembangan Agile. Agile merupakan pendekatan yang relatif baru dalam pengembangan perangkat lunak. Banyak organisasi terkemuka telah mengadopsi Agile di semua atau beberapa proyek mereka.

Factor utama dalam tantangan yang dialami dalam penerapan Agile yaitu budaya perusahaan yang tidak sesuai dengan konsep Agile, kurangnya keterampilan atau pengalaman dalam menerapkan Agile, kurangnya dukungan dari pihak manajemen dan masalah hirarki dalam organisasi.

5 poin agile testing manifesto

1. Testing is an activity not a phase.

>> Di agile, pengujian *bukanlah fase tetapi aktivitas* yang perlu terjadi, bersama dengan pengkodean, dokumentasi, dan yang lainnya.

2. Prevent bugs rather than finding bugs

>> Sebagian besar bug terjadi pada fase persyaratan. Biasanya terlihat seperti ini: Orang membuat asumsi tentang persyaratan mereka dan menerapkan asumsi tersebut

sebelum mengklarifikasinya. Asumsinya baru jelas setelah perangkat lunak diuji dan kesalahan ditemukan. Cara terbaik untuk mencegah kesalahan adalah dengan mengajukan pertanyaan.

3. Building the system OVER breaking the system

>> Tujuan utama dari proses pengembangan yaitu untuk menciptakan produk berkualitas tinggi. Untuk itu, QE perlu memposisikan dirinya sebagai pengguna, mengevaluasi produk dari sudut pandang pengguna, dan mampu menginformasikan pengembang.

4. Testing understanding over checking functionality

>> Penguji yang berpikir bahwa tugas mereka untuk memastikan produk memenuhi spesifikasi sering kali tidak menyukai kelincahan. Namun, mereka hanya secara resmi memeriksa seberapa dekat pengembang mengikuti spesifikasi. Itu tidak benar-benar mengatakan apa pun tentang kualitas produk atau apakah itu memenuhi tujuannya. Jika pekerjaan penguji seperti itu, semua tugas pengujian dapat diotomatisasi dan tidak ada input manusia dalam prosesnya. Pengujian tangkas perlu mengotomatiskan pemeriksaan sederhana untuk memungkinkan penguji melakukan tugas yang tidak dapat dilakukan computer atau pengujian kegunaan.

5. Team responsibility for quality OVER tester responsibility

>> Mempertimbangkan tanggung jawab seluruh tim untuk kualitas adalah prinsip dasar tangkas. Beban kualitas produk yang dibangun tidak boleh hanya ditanggung oleh satu anggota. Misalnya, jika bug baru ditemukan saat produk dirilis, QE tidak harus bertanggung jawab untuk itu. QE mungkin tidak peduli dengan bug dan tidak membebani developer. Namun, setiap anggota tim bertanggung jawab atas kepemilikan bersama atas produk yang sedang dibangun.