

Tugas Section 8 – Slide 40

1.

1. QE Pada Fase Recruitments Gathring

QE membantu perusahaan untuk menghemat anggaran yang ada dengan berpartisipasi memberikan ide tentang fitur yang akan digunakan di masa mendatang. Selain itu, QE juga dapat mencuri start untuk menjalankan test case, membuat skenario test yang akan digunakan nantinya.

2. QE Pada Fase Design

Sama seperti fase sebelumnya, pada fase desain, QE dapat dilibatkan dalam membantu mengidentifikasi desain yang nantinya akan menjadi pemblokir atau masalah perangkat lunak.

3. QE Pada Fase Development

Pada fase ini, QE juga mempertimbangkan dampak dan pengalaman bagi pengguna di masa mendatang, mungkin dengan memberikan saran tentang animasi atau posisi tombol yang nantinya ada di perangkat lunak.

4. QE Pada Fase Testing

Fase ini merupakan fase terpenting bagi QE dimana QE akan membuat test case dan mengeksekusinya untuk menjaga kualitas yang akan dikembangkan. Selain itu, QE akan membuat laporan bug jika menemukan bug dalam proses pengujian. Saat proses dijalankan QE juga mengetahui kebutuhan sistem terbaik yang akan digunakan oleh pengguna akhir nantinya.

5. QE Pada Fase deployment

Pada fase ini biasanya dilakukan smoke testing untuk memastikan fungsi-fungsi penting yang ada pada software agar dapat berjalan dengan baik tanpa ada masalah atau showstopper.

6. QE Pada Fase maintenance

Pada fase ini akan mengulangi apa yang telah dilakukan pada fase pengujian. Hal ini juga berlaku pada tahap bug fixing, QE juga harus melakukan verifikasi terhadap perbaikan bug yang telah dilakukan oleh tim programmer sebelumnya. Pengujian pengujian lainnya juga akan dilakukan untuk memastikan fungsi-fungsi yang ada dapat berjalan dengan baik. Kemudian pada fase ini QE bahkan memeriksa bug yang dilaporkan oleh pengguna perangkat lunak.

2.

1. Testing is an “activity” not a phase

Tester harus melakukan testing secara berkala sedangkan tantangannya adalah ketika di akhir proses pengembangan QE harus menguji software secara keseluruhan dan ini akan memakan waktu yang cukup lama

2. “Prevent bugs” rather than finding bugs

Biasanya karena persyaratan yang tidak lengkap atau rinci. Tantangan yang harus dilalui QE adalah membuat persyaratan terperinci dan mengidentifikasi dari mana bug itu berasal. Hal ini juga dapat dilakukan dengan berhipotesis dengan tim programmer sehingga komunikasi juga merupakan hal yang penting untuk dilakukan

3. Don't be a checker, “be a tester”

Sebagai QE pemula, banyak yang masih takut untuk berkomunikasi dan memberikan umpan balik kepada tim programmer. Tantangan-tantangan ini dapat dilatih dengan menulis ulasan singkat tentang perangkat lunak yang diuji dan diberikan kepada tim programmer sebagai saran untuk perbaikan fitur.

4. Don't try to break the system, instead help “build the best” possible system.

Jika tahap sebelumnya memberikan review dengan kaca mata penguji, kali ini QE harus bisa memberikan masukan dari sudut pandang pengguna. Hal ini terkadang sulit dilakukan, terutama untuk penguji lama karena terkadang hasil input atau ulasan yang diberikan kepada programmer dapat menjadi bias (penguji perbaikan, pengguna: untuk mempermudah pekerjaan).

5. The whole “team is responsible” for quality, not just the tester

Tantangan yang akan dihadapi terkadang berupa tekanan yang didapat dari tim, biasanya setelah selesai build, yang hanya menguji tim QE, sehingga proses pengujian menjadi tidak efektif, terutama untuk tim yang memiliki sedikit QE.