RESUME PERBAIKAN BUKU QA

* KELOMPOK DARA MELISA

Dari hasil review kelompok ini, banyak penulisan yang masih belum sesuai. Berikut hasil perbaikan dari kelompok ini, yaitu:

1. Bab 6
2. Pembahasan Positive Testing

Memasukkan nilai dengan 99999 pada kolom tersebut dapat diterima/*valid* oleh sistem dikarenakan kolom tersebut bertipe Number.

Perbaikan:

Number 🡪 *Number*

1. Bab 7
2. Pembahasan Modern Web Architecture Explained

Pengujian web arsitektur terbagi atas dua uji, yaitu secara *Front-end* dan *Back-end*. Pengujian secara *Front-end* meliputi uji aplikasi dari bagian *user* yang melakukan interaksi secara langsung.

Perbaikan:

*Front-end* 🡪 *front-end*

*Back-end* 🡪 *back-end*

Pada gambar 1.7 dapat diketahui bahwa *user* adalah pada bagian *client*, dan *Server Side* adalah bagian yang tidak diketahui oleh *user* atau biasa disebut dengan *Back-end* aplikasi. Pada bagian *Server Side, user* tidak akan mengetahui proses logika dari aplikasi, *database*, dan hal sensitif lainnya.

Perbaikan:

*Server Side* 🡪 *server side*

*Back-end* 🡪 *back-end*

1. Pembahasan Bug Triage

Dari gambar 7.2 dapat diketahui bahwa *Back-end* adalah sebuah hal yang dilakukan untuk membuat *Front-end* sesuai dengan yang diinginkan. Seperti pada gambar 7.2, *developer* ingin menampilkan singa mengaung, tentunya pada segi *back-end* atau belakang panggung menunjukkan bahwa singa tersebut diikat dan hanya menampilkan kepalanya saja.

Perbaikan:

*Back-end* 🡪 *back-end*

*Front-end* 🡪 *front-end*

1. How to Debug Front-End or Back-End Issues?

Dari gambar 7.10 dapat diketahui bahwa proses validasi terhadap kolom *login* masing kurang bagus, dikarenakan harus melakukan proses secara *front-end* dan *back-end*. Alangkah lebih bagus menggunakan proses validasi seperti menggunakan Ajax atau pengecekan javascript terlebih dahulu sebelum melakukan proses *login*.

Perbaikan:

javascript 🡪 javascript (tidak ada perbaikan karena javascript tidak termasuk bahasa asing).

1. Bab 8
2. Pembahasan Sanity Testing

Tujuannya adalah "bukan" untuk memverifikasi secara menyeluruh fungsionalitas baru tetapi untuk menentukan bahwa pengembang telah menerapkan beberapa rasionalitas saat memproduksi perangkat lunak.

Perbaikan:

Tujuannya adalah untuk menentukan bahwa pengembang telah menerapkan beberapa rasionalitas saat memproduksi perangkat lunak.

1. Bab 9
2. Pembahasan Peran dan Tanggung Jawab Quality Assurance
3. Melakukan penyusunan terkait perencanaan Prosedur Operasi Standar (SOP) proses produksi terhadap produk dan layanan.

Perbaikan:

Prosedur Operasi Standar 🡪 Standar Operasional Prosedur

1. Pembahasan Education Requirements for Quality Assurances
2. Memiliki pola pikir pengujian, pengujian keamanan, otomasi hingga UAT (User Acceptance Testing)

Perbaikan:

User Acceptance Testing 🡪 *User Acceptance Testing*

Dalam menjalankan proses atau tugasnya, seorang QA membutuhkan dukungan peralatan atau tools yang kompatibel sesuai dengan pekerjaannya.

Perbaikan:

tools 🡪 *tools*

Misalnya *Software Quality Assurance* atau *QA Engineer*, maka harus bisa menguasai SDLC (*Software Development Life Cycle*), bahasa pemrograman seperti (Java, JavaScript, Python, C++ dll), *testing tool*, *manual testing* hingga *automation testing*.

Perbaikan:

Java, JavaScript, Python, C++ 🡪 Java, JavaScript, Python, C++ (tidak ada perbaikan karena tidak termasuk bahasa asing).

1. Pembahasan Quality Assurance Jobs

Banyak *platform* yang dapat digunakan untuk melamar pekerjaan sebagai *quality assurance*, diantaranya LinkedIn, Glassdoor, Indeed, dan Monster.

Perbaikan:

LinkedIn, Glassdoor, Indeed, dan Monster 🡪 LinkedIn, Glassdoor, Indeed, dan Monster (tidak ada perbaikan karena tidak termasuk bahasa asing).