

5. Semua kasus uji integrasi
6. Semua kasus uji kompleks
7. Kasus uji nilai batas
8. Contoh kasus uji yang berhasil
9. Contoh kasus uji Kegagalan

### C. Positive Testing and Negative Testing

Pengujian secara positif adalah untuk menentukan bahwa aplikasi berfungsi seperti yang diharapkan. Jika terjadi kesalahan selama pengujian positif, pengujian dianggap gagal. Adapun pengujian secara negatif adalah memastikan bahwa aplikasi dapat menangani input yang tidak valid atau perilaku pengguna yang tidak terduga.

#### a. Positive testing

Pengujian ini berupa menguji inputan nilai yang *valid*. Kemudian melihat apakah aplikasi berjalan sesuai dengan harapan. Contohnya pada inputan berikut ini:

Enter Only Numbers

99999

#### Positive Testing

Gambar 6.1 Pengujian dengan inputan yang benar

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

#### b. Negative testing

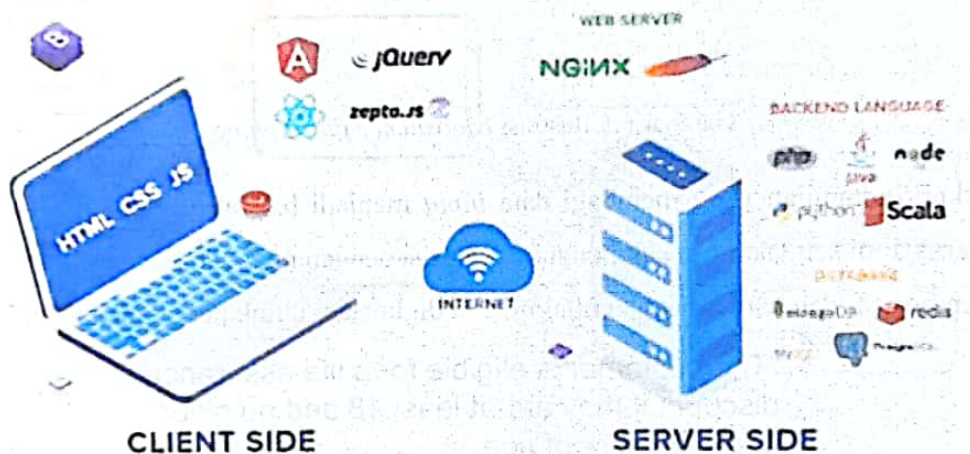
Pada *negative testing*, pengujian yang dilakukan berupa menguji inputan yang salah, dan melihat apakah program akan memberikan notifikasi berupa *invalid* atau tidak. Contohnya jika kolom inputan seperti berikut :

## BAB VII WEB ARCHITECTURE

### A. Modern Web Architecture Explained

Arsitektur pengujian adalah ilmu disiplin melihat aliran pengiriman dan mencari tahu apa, bagaimana, dan kapan pengujian untuk mencapai hasil terbaik. Pengujian ini menghasilkan kerangka kerja aktivitas pengujian yang kemudian dapat dicakup, disiapkan, dan dieksekusi.

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. Contoh:



Gambar 7.1 Ilustrasi interaksi user

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.

Web Architecture bekerja secara *front-end* dan *back-end*. Adapun proses yang terjadi selama *user* melakukan *request* untuk membuka sebuah web terbagi atas 2 proses, yaitu proses pada bagian *user* yang disebut *front-end* dan proses pada bagian yang tidak dapat dilihat oleh *user* disebut *back-end*.

### Front-end :

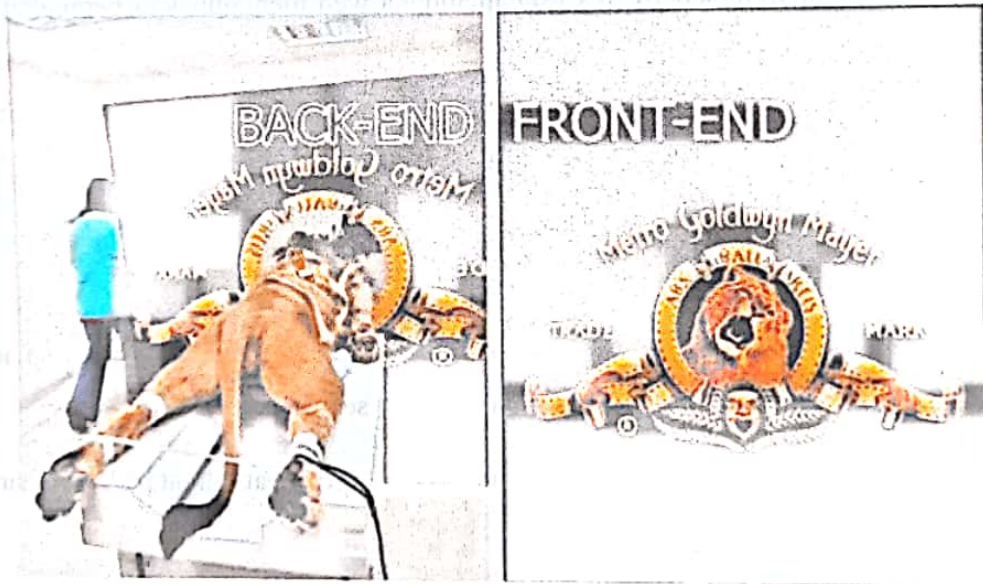
- *Browser* : User mengakses halaman web (HTTP/HTTPS Protocol)

### Back-end :

- *Back-End* : Mengirim kembali sebagai HTML (Bahasa Standar untuk web *browser*)
- *Browser* : Menerjemahkan halaman web
- *Browser* : Download seluruh *asset* yang dibutuhkan seperti *script*, *font*, gambar  
Menghubungi pihak ketiga dari *back-end* seperti (metode pembayaran, *analytics*,  
*maps*)
- *Browser* : Menampilkan halaman web ke user

## B. Back-End Testing

Pengujian *back-end* adalah pengujian yang dilakukan dari segi logika atau segi yang tidak dapat dilihat oleh user. Pengujian dari segi *back-end* biasanya melibatkan pengujian seperti *Web Service Testing*, *Database Testing*, dan *Server Side Testing*. Adapun analogi dari *back-end* dan *front-end* sebagaimana gambar 7.2.

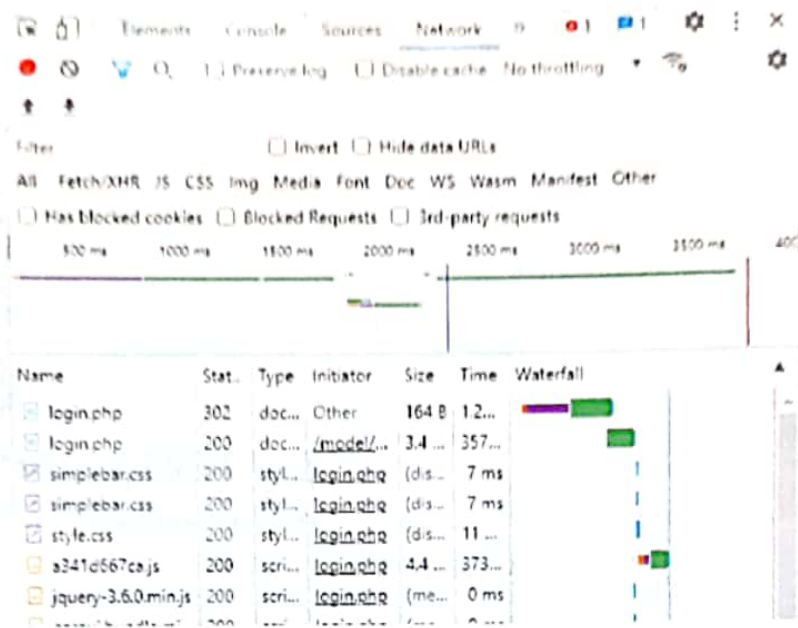


Gambar 7.2 Ilustrasi *back-end* dan *front-end*

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



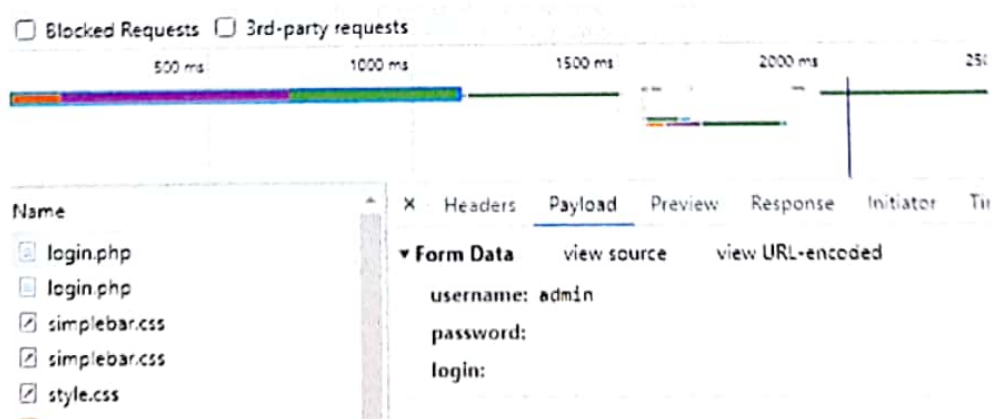
6. Kemudian klik tombol *login* dengan *password* yang kosong, maka hasil pada *network* sebagai pada gambar 7.10.



Gambar 7.10 Hasil login tanpa password

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*.

7. Pada bagian file login dengan status 302 kita akan dapat melihat status *payload* dengan informasi dari *username* dan *password* yang diinput



Gambar 7.11 Payload form data

- *Build* adalah versi perangkat lunak yang diserahkan oleh tim pengembangan ke tim pengujian untuk tujuan pengujian, sedangkan
- *Rilis* adalah perangkat lunak yang tim uji berikan kepada pelanggan untuk dicoba

#### D. What is Sanity Testing?

*Sanity testing* adalah sejenis pengujian perangkat lunak yang dilakukan setelah menerima pembuatan perangkat lunak, dengan sedikit perubahan dalam kode, atau fungsionalitas, untuk memastikan bahwa *bug* telah diperbaiki dan tidak ada masalah lebih lanjut yang ditimbulkan karena perubahan ini. Tujuannya adalah untuk menentukan bahwa fungsionalitas yang diusulkan bekerja seperti yang diharapkan. Jika *sanity testing* gagal, *build* akan ditolak untuk menghemat waktu dan biaya yang terlibat dalam pengujian yang lebih ketat.

Tujuannya adalah "~~bukan~~" untuk memverifikasi secara menyeluruh fungsionalitas baru tetapi untuk menentukan bahwa pengembang telah menerapkan beberapa rasionalitas saat memproduksi perangkat lunak. Contoh *sanity testing*:

Dalam proyek *e-commerce*, modul utama adalah halaman *login*, halaman beranda, halaman profil pengguna, pendaftaran pengguna, dll. Terdapat kecacatan proses pada halaman *login* ketika kata sandi menerima kurang dari empat karakter dan persyaratan menyebutkan bahwa kata sandi ini bidang tidak boleh di bawah delapan karakter. Oleh karena itu, cacat dilaporkan oleh tim pengujian ke tim pengembangan untuk menyelesaikannya. Kemudian tim pengembangan memperbaiki cacat yang dilaporkan dan mengirimkannya ke tim pengujian untuk dibersihkan. Kemudian tim penguji memeriksa apakah perubahan yang dilakukan berfungsi dengan baik atau tidak. Itu juga ditentukan apakah itu berdampak pada fungsi terkait lainnya. Sekarang ada fungsi untuk memperbarui kata sandi di halaman profil pengguna.

Keuntungan dari *sanity testing*:

1. *Sanity testing* membantu dengan cepat mengidentifikasi cacat pada fungsionalitas inti.
2. Dapat dilakukan dalam waktu yang lebih singkat karena tidak ada dokumentasi yang diperlukan untuk *sanity testing*.
3. Jika cacat ditemukan selama *sanity testing*, proyek akan ditolak untuk menghemat waktu pelaksanaan uji regresi.
4. Teknik pengujian ini tidak begitu mahal jika dibandingkan dengan jenis pengujian lainnya.

### C. Peran dan Tanggung Jawab Quality Assurance

Pada umumnya tugas dari *Quality Assurance* adalah menjamin kualitas produk dari suatu perusahaan yang akan dijual atau masih dalam proses pengembangan. Selain itu QA juga memiliki beberapa tugas lain, di antaranya adalah sebagai berikut ini:

1. Membuat perencanaan terhadap pengujian dan kasus pengujian secara terperinci serta komprehensi terstruktur.
2. Melakukan tafsir, membangun dan mematuhi standar terhadap jaminan kualitas dari perusahaan yang menjual produk atau jasa.
3. Melakukan analisis terhadap keluhan konsumen dan ketidaksesuaian kualitas, selain itu juga mencari akar masalah serta tindakan penyelesaian yang sesuai dengan visi perusahaan.
4. Melakukan pengembangan standar baru dalam produksi sesuai dengan kebutuhan dan membuat protokol pengujian.
5. Melakukan dokumentasi aktivitas jaminan kualitas dalam bentuk laporan dan audit secara internal dalam perusahaan.
6. Memastikan produk yang dibuat sudah memenuhi standar perusahaan dan kebutuhan konsumen atau para pelanggan.
7. Melakukan dokumentasi perbaikan terhadap produk setelah dilakukan pengujian sebelum nantinya dijual ke pasaran.
8. Melakukan dokumentasi berupa catatan perbaikan yang dijadikan sebagai referensi terhadap produk setelah dilakukan pengujian.
9. Melakukan penyusunan terkait perencanaan Prosedur Operasi Standar (SOP) proses produksi terhadap produk dan layanan. ~~tebal ek (Standar Operasi)~~
10. Bekerja sama, berkolaborasi dengan tim internal agar menemukan solusi dalam pemecahan masalah yang dihadapi saat itu.

### D. Education Requirements for Quality Assurances

Menjadi seseorang dari QA harus memahami skill dan kemampuan dasar untuk bisa bekerja secara profesional dan sesuai dengan kualifikasi kerja di suatu perusahaan, berikut di antara dari kemampuan yang dibutuhkan:

1. Memahami, aktif berkomunikasi dan membaca data dengan sangat baik.
2. Memiliki keahlian dalam manajemen proyek yang optimal.



- *Build* adalah versi perangkat lunak yang diserahkan oleh tim pengembangan ke tim pengujian untuk tujuan pengujian, sedangkan
- Rilis adalah perangkat lunak yang tim uji berikan kepada pelanggan untuk dicoba

#### D. What is Sanity Testing?

*Sanity testing* adalah sejenis pengujian perangkat lunak yang dilakukan setelah menerima pembuatan perangkat lunak, dengan sedikit perubahan dalam kode, atau fungsionalitas, untuk memastikan bahwa *bug* telah diperbaiki dan tidak ada masalah lebih lanjut yang ditimbulkan karena perubahan ini. Tujuannya adalah untuk menentukan bahwa fungsionalitas yang diusulkan bekerja seperti yang diharapkan. Jika *sanity testing* gagal, *build* akan ditolak untuk menghemat waktu dan biaya yang terlibat dalam pengujian yang lebih ketat.

Tujuannya adalah "~~bukan~~" untuk memverifikasi secara menyeluruh fungsionalitas baru ~~tetapi~~ untuk menentukan bahwa pengembang telah menerapkan beberapa rasionalitas saat memproduksi perangkat lunak. Contoh *sanity testing*:

Dalam proyek *e-commerce*, modul utama adalah halaman *login*, halaman beranda, halaman profil pengguna, pendaftaran pengguna, dll. Terdapat kecacatan proses pada halaman *login* ketika kata sandi menerima kurang dari empat karakter dan persyaratan menyebutkan bahwa kata sandi ini bidang tidak boleh di bawah delapan karakter. Oleh karena itu, cacat dilaporkan oleh tim pengujian ke tim pengembangan untuk menyelesaikannya. Kemudian tim pengembangan memperbaiki cacat yang dilaporkan dan mengirimkannya ke tim pengujian untuk dibersihkan. Kemudian tim penguji memeriksa apakah perubahan yang dilakukan berfungsi dengan baik atau tidak. Itu juga ditentukan apakah itu berdampak pada fungsi terkait lainnya. Sekarang ada fungsi untuk memperbarui kata sandi di halaman profil pengguna.

Keuntungan dari *sanity testing*:

1. *Sanity testing* membantu dengan cepat mengidentifikasi cacat pada fungsionalitas inti.
2. Dapat dilakukan dalam waktu yang lebih singkat karena tidak ada dokumentasi yang diperlukan untuk *sanity testing*.
3. Jika cacat ditemukan selama *sanity testing*, proyek akan ditolak untuk menghemat waktu pelaksanaan uji regresi.
4. Teknik pengujian ini tidak begitu mahal jika dibandingkan dengan jenis pengujian lainnya.

### C. Peran dan Tanggung Jawab Quality Assurance

Pada umumnya tugas dari *Quality Assurance* adalah menjamin kualitas produk dari suatu perusahaan yang akan dijual atau masih dalam proses pengembangan. Selain itu QA juga memiliki beberapa tugas lain, di antaranya adalah sebagai berikut ini:

1. Membuat perencanaan terhadap pengujian dan kasus pengujian secara terperinci serta komprehensi terstruktur.
2. Melakukan tafsir, membangun dan mematuhi standar terhadap jaminan kualitas dari perusahaan yang menjual produk atau jasa.
3. Melakukan analisis terhadap keluhan konsumen dan ketidaksesuaian kualitas, selain itu juga mencari akar masalah serta tindakan penyelesaian yang sesuai dengan visi perusahaan.
4. Melakukan pengembangan standar baru dalam produksi sesuai dengan kebutuhan dan membuat protokol pengujian.
5. Melakukan dokumentasi aktivitas jaminan kualitas dalam bentuk laporan dan audit secara internal dalam perusahaan.
6. Memastikan produk yang dibuat sudah memenuhi standar perusahaan dan kebutuhan konsumen atau para pelanggan.
7. Melakukan dokumentasi perbaikan terhadap produk setelah dilakukan pengujian sebelum nantinya dijual ke pasaran.
8. Melakukan dokumentasi berupa catatan perbaikan yang dijadikan sebagai referensi terhadap produk setelah dilakukan pengujian.
9. Melakukan penyusunan terkait perencanaan Prosedur Operasi Standar (SOP) proses produksi terhadap produk dan layanan. ~~tebalok (Standar Operasi)~~
10. Bekerja sama, berkolaborasi dengan tim internal agar menemukan solusi dalam pemecahan masalah yang dihadapi saat itu.

### D. Education Requirements for Quality Assurances

Menjadi seseorang dari QA harus memahami skill dan kemampuan dasar untuk bisa bekerja secara profesional dan sesuai dengan kualifikasi kerja di suatu perusahaan, berikut di antara dari kemampuan yang dibutuhkan:

1. Memahami, aktif berkomunikasi dan membaca data dengan sangat baik.
2. Memiliki keahlian dalam manajemen proyek yang optimal.



3. Memahami sebuah produk yang bisa mempengaruhi pola pikir, kebiasaan dan kehidupan manusia.
4. Kemampuan berpikir kritis dan kemampuan analisis serta riset.
5. Keahlian dalam manajerial hingga penguasaan bahasa asing yang diperlukan oleh perusahaan.
6. Kemampuan analisa dan pengujian fungsional serta perbaikan dalam proses yang dilakukan.
7. Memiliki pola pikir pengujian, pengujian keamanan, otomasi hingga UAT (User Acceptance Testing).

huruf miring

Dalam menjalankan proses atau tugasnya, seorang QA membutuhkan dukungan peralatan atau tools yang kompatibel sesuai dengan pekerjaannya. Seperti salah satunya instalasi aplikasi GIT, selain harus memahami algoritma dan struktur data bahasa pemrograman yang dipakai dalam perusahaan tempat mereka bekerja. Keahlian untuk menjadi QA biasanya mirip, kurang lebih seperti yang disebutkan dan untuk *hard skill* biasanya disesuaikan dengan posisi yang dilamar. 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*.

miring

## E. Quality Assurance Jobs

Saat ini, kebutuhan perusahaan terkait profesi *quality assurance* semakin meningkat. Oleh karena itu, lowongan kerja *quality assurance* banyak dibuka dan pastinya peluang untuk berpartisipasi pada bidang ini sangatlah besar. Banyak *platform* yang dapat digunakan untuk melamar pekerjaan sebagai *quality assurance*, diantaranya LinkedIn, Glassdoor, Indeed, dan

Monster.

huruf miring