# Bagian A – Pengetahuan Dasar QA

1. Jelaskan perbedaan antara bug, defect, dan error.

Jawaban:

* + Error adalah **kesalahan yang dibuat oleh manusia** selama proses pengembangan, desain, atau penggunaan perangkat lunak. Contohnya syntax error (kesalahan penulisan) yang disebabkan oleh developer; suatu API memerlukan parameter namun dari FE tidak mengirimkan parameter yang dibutuhkan sehingga menyebabkan data tidak tampil.
  + Defect adalah **ketidaksesuaian** antara hasil yang diharapkan dan hasil aktual dari perangkat lunak. Contohnya tombol login tidak berfungsi ketika diklik; login dengan data yang tidak valid namun berhasil login.
  + Bugs adalah defect yang ditemukan setelah **perangkat lunak mulai diuji** atau digunakan. Contohnya harga yang perlu dibayar ketika melakukan checkout barang A dan B tidak menampilkan jumlah harga dari barang A dan B melainkan hanya menampilkan harga barang A saja.

1. Apa yang dimaksud dengan regression testing dan kapan biasanya dilakukan?

Jawaban:

Regression testing adalah jenis pengujian perangkat lunak yang dilakukan untuk memastikan bahwa perubahan pada kode tidak menyebabkan kerusakan atau gangguan pada fungsi yang sudah ada sebelumnya. Biasanya dilakukan setelah perbaikan bug (bug fixing), penambahan fitur baru, perubahan konfigurasi system, saat menjelang rilis versi baru dan secara berkala di setiap sprint dalam model agile.

1. Sebutkan lifecycle dari bug/defect dari ditemukan hingga ditutup.

Jawaban:

* + New : ketika bug baru ditemukan dan dicatat di bug tracking, belum ada peninjauan dari developer
  + Assigned : bug sudah diperiksa oleh QA Lead atau Project Manager, lalu diberikan ke developer untuk diperbaiki
  + In progress : developer mulai menginvestigasi dan memperbaiki bug tersebut
  + Fixed : developer telah melakukan perubahan pada kode dan menyatakan bug telah diperbaiki
  + Retest : bug dikembalikan ke tim QA untuk dilakukan pengujian ulang (retest) guna memastikan perbaikan benar-benar berhasil
  + Reopen : saat QA melakukan retest namun bug masih muncul atau belum sesuai expected result
  + Closed : QA menutup bug karena sudah berhasil diperbaiki dan tidak ada masalah lanjutan
  + Rejected : bug yang telah ditinjau namun dianggap tidak valid atau tidak perlu diperbaiki

1. Apa itu test case dan test scenario? Jelaskan perbedaannya dan berikan contohnya.

Jawaban:

* + Test scenario adalah gambaran umum apa saja yang akan diuji tanpa menjelaskan langkah-langkah nya secara rinci dengan memberikan konteks mengenai alur atau proses yang perlu diuji.
  + Test case adalah rincian dari test scenario yang sudah dibuat dan berisi informasi terkait komponen apa saja yang akan diuji, langkah pengujian, kondisi awal, hasil yang diharapkan, data pengujian dan status pengujian.
  + Contohnya ketika akan melakukan pengujian fitur login:
    - Test scenario

Judul: verifikasi proses login pada aplikasi menggunakan kredensial yang valid.

* + - Test case

Judul: login dengan email dan password yang valid

Data Uji:

email: [vtest21@gmail.com](mailto:vtest21@gmail.com)

password: 12345678A

Langkah-langkah:

1. Buka halaman login
2. Masukkan email yang valid
3. Masukkan password yang valid
4. Klik tombol “Login”

Pre-condition: sudah memiliki kredensial yang valid, halaman login dapat diakses, pengguna belum dalam kondisi login

Expected result: login berhasil dan pengguna diarahkan ke halaman dashboard

Actual result: login berhasil dan pengguna diarahkan ke halaman dashboard

Status: Pass

1. Apa yang dimaksud dengan severity dan priority dalam konteks bug? Berikan masing-masing contohnya.

Jawaban:

* Priority dalam konteks bug menunjukan urgensi perbaikan bug. Biasanya dibagi menjadi 3 level, yaitu:
  + High: bug yang harus segera diperbaiki karena berdampak besar pada pengguna atau proses bisnis. Contohnya user tidak bisa melakukan login.
  + Medium: bug yang cukup penting namun dapat menunggu setelah prioritas diatasnya selesai terlebih dahulu. Contohnya di halaman “Riwayat Transaksi”, urutan data tidak sesuai default (tidak dari yang terbaru).
  + Low: bug bersifat minor yang tidak mengganggu fungsionalitas utama sehingga dapat diperbaiki dengan adanya pengembangan versi terbaru. Contohnhya di halaman dashboard, ikon user tidak sejajar dengan teks nama
* Severity dalam konteks bug menunjukan tingkat keparahan bug berdasarkan tingkat kerusakan atau kegagalan sistem yang ditimbulkan oleh bug tersebut. Level keparahan sebuah bug biasanya dibagi menjadi:
  + Critical (Blokir): ketika tidak dapat melanjutkan ke proses selanjutnya atau menimbulkan kerusakan yang parah, contohnya setelah klik tombol register, aplikasi blank; transaksi berhasil, tetapi saldo user tidak berkurang
  + High (Mayor): fungsi penting tidak berjalan sesuai ekspektasi, tapi sistem masih bisa berjalan sebagian, contohnya transaksi gagal tanpa pesan error
  + Medium (Minor): ada error, tapi tidak memengaruhi fungsi utama, contohnya format tanggal di filter salah (misalnya tampil dengan format "MM/DD/YYYY" padahal seharusnya "DD/MM/YYYY"); notifikasi muncul dua kali untuk satu aksi.
  + Low (Trivial): masalah sangat kecil seperti typo atau kesalahan UI yang tidak memengaruhi penggunaan, contohnya warna tombol tidak sesuai figma; icon terlalu besar pada mobile view