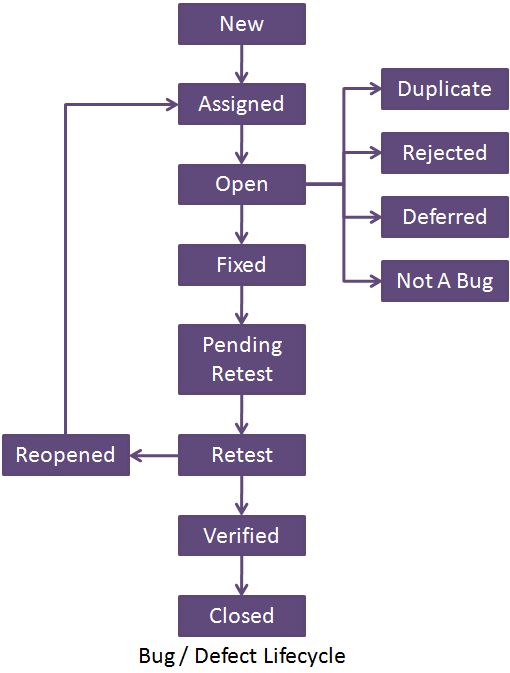
1. Metode agile sendiri yaitu memecah produk menjadi beberapa siklus dan lantas dengan cepat memberikan fungsi pada produk. Metodologi ini merupakan pendekatan pengembangan yang lebih realistis dan cepat.

Metode waterfall adalah salah satu jenis model pengembangan aplikasi dan termasuk ke dalam *classic life cycle*yang mana menekankan pada fase yang berurutan dan sistematis. Untuk model pengembangannya, dapat dianalogikan seperti air terjun, dimana setiap tahap dikerjakan secara berurutan mulai dari atas hingga ke bawah.

Metode waterfall dan agile QA berperan dibagian development, testing dan juga interaksi dengan tim, client dan PO.

1. Bug membuat program mengeluarkan output yang salah atau tidak sesuai atau bahkan tidak diinginkan oleh pengguna. Penyebabnya bisa karena kesalahan program yang tidak sengaja dibuat oleh programmer itu sendiri saat coding aplikasi melinhkupi kesalahan logis, kesalahan penulisan sintak, dan debugging.



* **New :** Ketika bug pertama kali ditemukan dan dicatat oleh tester untuk diberikan kepada pengembang.
* **Assigned :** Setelah tester sudah mencatat bug apa saja yang telah terjadi. Ketua tim dari tester langsung memberikan pencatatan hasil bug kepada tim pengembang. Lalu ketua tim pengembang akan membagi tugas kepada anggotanya untuk mengatasi bug tersebut.
* **Open :**Pada tahap ini, pengembang telah melakukan analisis terhadap bug tersebut dan bekerja untuk memperbaiki kesalahan yang telah terjadi.
* **Fixed :** Ketika pengembang telah mengubah *code* yang diperlukan dan dapat diverifikasi untuk dapat lolos ke bagian pengujian. Sehingga, status *bug* dapat berubah menjadi “*Fixed*”.
* **Pending Retest :** Setelah memperbaiki bug, pengembang telah memberi beberapa kode tertentu untuk diuji ulang ke penguji. Pada tahap diperlukan cukup waktu hingga sampai di penguji akhir. Sehingga statusnya berubah menjadi sedang menunggu tes ulang.
* **Retest :** Pada tahap ini tester melakukan pengujian ulang terhadap kode yang telah diubah yang telah diberikan pengembang kepadanya untuk memeriksa apakah bug tersebut telah diperbaiki atau tidak.
* **Verified :** Penguji tes bug melakukan uji coba setelah diperbaiki oleh pengembang. Jika bug tidak hadir kembali dalam perangkat lunak. Penguji berhak menyetujui bahwa bug tersebut telah diselesaikan dan mengubah statusnya menjadi telah “diverifikasi”.
* **Reopen :** Jika bug masih ada bahkan setelah bug diperbaiki oleh pengembang, pengubah akan mengubah status menjadi “reopened“. Bug itu melewati siklus hidup sekali lagi.
* **Closed :** Setelah bug diperbaiki dan diuji oleh penguji, apabila penguji merasa bahwa bug sudah tidak ada lagi dalam perangkat lunak, dia mengubah status bug menjadi “tertutup”. Keadaan ini menunjukan bahwa bug itu sudah diperbaiki, teruji dan disetujui.
* **Duplicate :** Jika bug diulang dua kali atau dua bug menyebutkan konsep bug yang sama, maka satu status bug berubah menjadi “duplikat”.
* **Rejected :** Jika pengembang merasa bug itu bukan merupakan suatu masalah, dia akan menolak bug yang dimaksudkan. Maka keadaan bug berubah menjadi “ditolak” dan belum dapat diperbaiki.
* **Deferred :** Bug tersebut dapat berubah menjadi status untuk ditangguhkan, berarti bug tersebut diperkirakan akan diperbaiki pada rilis atau periode berikutnya. Alasan untuk mengubah bug ke keadaan ini memiliki banyak faktor. Diantaranya adalah prioritas bug yang mungkin rendah, kurangnya waktu untuk rilis atau bug mungkin tidak berpengaruh besar pada perangkat lunak.
* **Not a Bug :** Status diberikan sebagai “Not a bug” jika tidak ada perubahan fungsi dari aplikasi. Sebagai contoh adalah jika pelanggan meminta beberapa perubahan pada tampilan dan bidang aplikasi seperti perubahan warna beberapa teks maka itu bukan bug tetapi hanya beberapa perubahan pada tampilan aplikasi.

1. Skill yang diperlukan :

* Mengerti basic software engineering
* Time management
* Komunikasi
* Kemampuan analisis dan berpikir logis
* Teliti
* Kreatif
* Mampu bekerja dalam tim