1. What QA do in SCRUM? Explain!

Answer: Pada SCRUM terdapat empat tahapan yang melibatkan anggotanya. Tahapannya yaitu *Sprint Planning*, *Daily SCRUM*, *Sprint Review*, dan *Sprint Retro*. Pada setaip masing-masing tahapan, yang dilakukan QA pastinya berbeda-beda.

* ***Sprint Planning***, tahap ini adalah tahap awal yaitu perencanaan sprint terhadap produk yang akan dibuat yang akan menghasilkan *product backlog*. Pada tahap ini, sebagai QA hal yang akan dilakukan yaitu membaca dokumen requirement yang ada. QA juga perlu melihat *design* atau *mock up* yang dibuat oleh *designer*. Setelah membaca *requirement* dan melihat *design mockup*, QA dapat bertanya kepada PM dan juga *designer* terdapat *requirement* dan *design* yang belum dipahami atau tidak jelas. QA juga dapat memberikan saran terhadap *requirement* atau *design* yang dikira kurang kepada PM serta *designer*.
* ***Daily SCRUM,*** pada tahap ini ialah tahap progress dari pembuatan produk. Setelah tahap perencanaan selesai, QA akan membuat test scenario serta test case sesuai dengan product backlog berdasarkan dari design yang telah dibuat oleh *designer*. Kemudian, setelah product backlog yang sudah selesai dibuat oleh developer maka tugas QA yaitu melakukan pengujian berdasarkan scenario yang sudah dibuat sebelumnya. Pada daily scrum, setiap hari anggota tim melakukan rapat, di sini QA perlu menjabarkan progress yang sudah dilakukan dan melaporkan hambatan yang mungkin ada.
* ***Sprint Review,*** tahap ini QA sebagai anggota tim dapat memberikan umpan balik kepada tim selama sprint berlangsung serta QA juga perlu melaporkan hambatan yang ada.
* ***Sprint Retro,*** tahap ini setiap anggota tim dapat memberikan umpan balik terhadapat kinerja dari masing-masing anggota selama melakukan sprint. Review ini digunakan untuk mengetahui yang tidak berjalan baik dan berjalan baik yang dapat digunakan untuk meningkatkan kinerja tim untuk sprint berikutnya.

1. Test Scenario
2. Explain about severity of bugs!

Answer: *Severity of bugs* adalah tingkat bug yang ditemukan dimana bug tersebut berdampak pada operasi fungsionalitas yang diuji. Bug yang memiliki efek lebih tinggi pada fungsionalitas akan masuk ke katagori yang lebih tinggi dari pada bug yang memiliki efek rendah. Tingkat bug pada umumnya terdapat *minor, medium, mayor* yang memiliki arti masing-masing, yaitu :

* ***Minor***, bug yang tidak berdampak besar pada sistem
* ***Medium***, bug yang menimbulkan beberapa hal tidak diinginkan, namun fungsionalitas sistem masih dapat digunakan.
* ***Major***, bug yang memiliki dampak besar sehingga dapat membuat sistem rusak.

Tingkat bug ini diperlukan karena untuk dapat memberikan prioritas pada bug apa saja yang perlu dikerjakan, apalagi jika bug ditemukan saat dekat-dekat dengan deadline rilis sistem.

1. Find issue and report the bugs!
2. How do you explain about deployment?

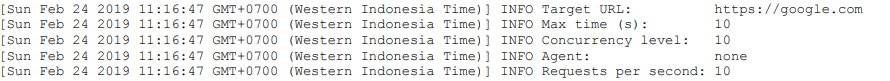
Answer: *Deployment* adalah proses di mana perangkat lunak atau perangkat keras dapat digunakan oleh user yang lainnya pada lingkungan yang berbeda. Prosesnya yaitu instalasi, konfigurasi, penjalanan, pengujian, dan membuat perubahan. Proses *deploy* lebih jelasnya, pada lingkungan yang berbeda perangkat masih tetap dapat digunakan. Contohnya jika pada pembuatan sistem di bidang IT, ada tiga lingkungan yaitu *develop, staging* dan *production*. Pada lingkungan *develop* yaitu sistem hanya dapat diakses oleh *developer* saja. Kemudian *staging*, dapat digunakan oleh QA. Terakhir *production* di mana sistem dapat digunakan oleh *end user* sesungguhnya.

1. Postman

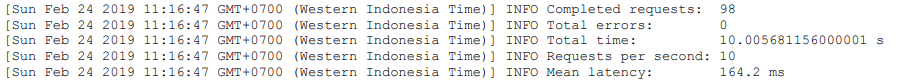
Answer: <https://www.getpostman.com/collections/856793247da2b57b6d5e>

1. Performance Test

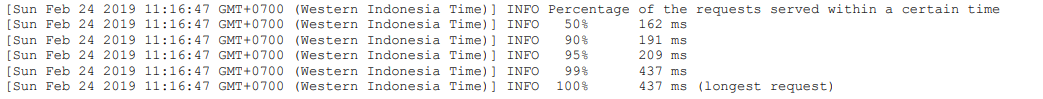
Answer:

1. 

Gambar di atas menampilkan perintah yang diminta/ditulis yaitu terget yang akan di uji yaitu google.com, waktu maksimal tunggu yaitu 10 detik, jumlah usernya yaitu 10, dan permintaan per detiknya yaitu 10. (dari atas ke bawah)



Gambar di atas menampilkan request yang berhasil dibuat yaitu sebanyak 98, dengan tidak ada error, dengan jumlah waktunya 10.000 detik sekian, request per detiknya 10, serta rata-rata jeda waktu yaitu 164.2 ms.



Gambar di atas menampilkan waktu yang didapat dari persentase requests yang dibuat dimana pertama melakukan tes dengan setengah permintaan yang menghasilkan waktu 162 ms dan kemudian hingga 100% yang menghasilkan permintaan terlama. Ini bertujuan untuk mengetahui apakah stabil atau tidak.

1. Membuat tes performa pada domain ***https://www.google.com*** menggunakan loadtest. Perintah di atas dapat dibaca menentukan 100 request setiap detiknya untuk setiap 10 users dalam permintaan yang dibuat dengan waktu maksimal 120 detik untuk menunggu responnya pada domain [www.google.com](http://www.google.com).
2. APACHE Jmater, Tsung, dan Locust.
3. Git