



# ***Software Quality Assurance***

**Radical Rakhman Wahid**




21 Mei 2019  
16 Ramadhan 1440

# Yang ada pada presentasi ini :

1. *Overview Software Quality Assurance*
2. Kerangka Teknis untuk *Metric Software*
3. Verifikasi & Validasi dan Pengornasisasian Pengujian Perangkat Lunak

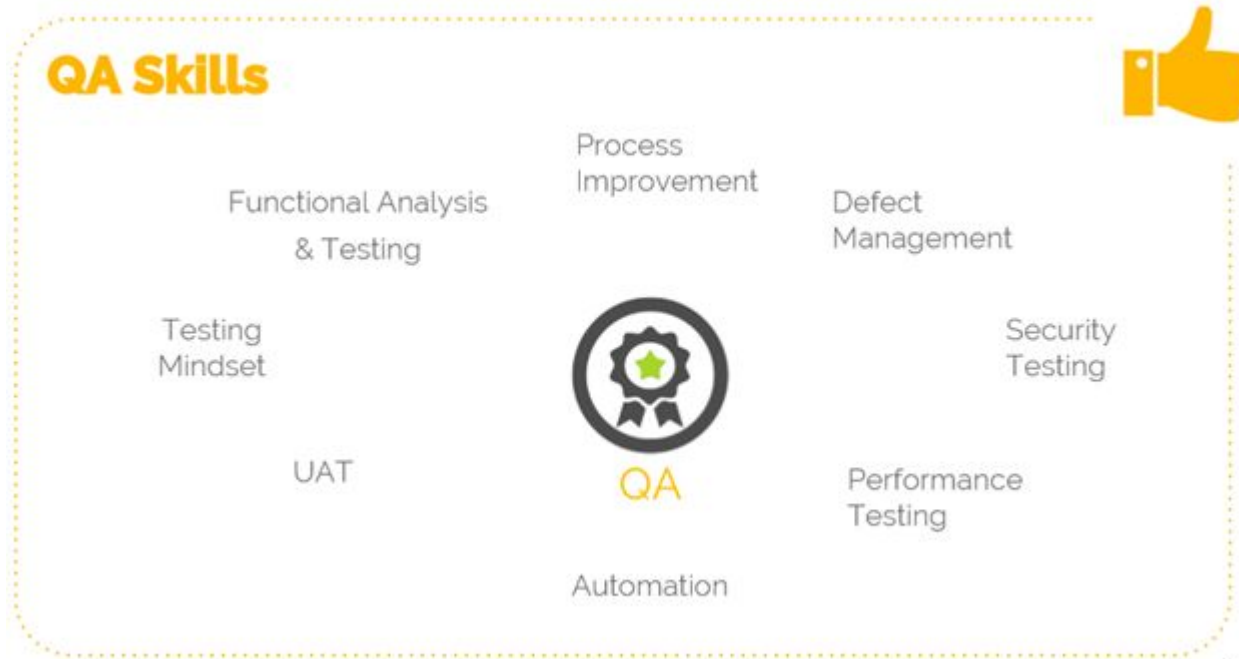


# **1.** ***Overview Software Quality Assurance***

A series of overlapping, semi-transparent geometric shapes in shades of green, blue, and purple, arranged in a diagonal pattern in the top-left corner.

*Software quality assurance* adalah proses sistematis untuk memeriksa apakah sebuah *software* telah dikembangkan sesuai dengan kebutuhan yang telah ditentukan sebelumnya. Proses ini, bisa dilaksanakan oleh seorang *quality assurance tester* atau oleh seorang *quality assurance engineer*.

# Kemampuan dasar untuk melakukan SQA :



# Konsep SQA

1. Pendekatan kualitas manajemen,
2. Teknologi rekayasa perangkat lunak yang efektif (metode dan tools yang digunakan),
3. Tinjauan teknis secara formal yang diaplikasikan melalui proses pengembangan software,
4. Strategi uji coba software yang multitier,
5. Kontrol terhadap dokumentasi software dan perubahannya
6. Prosedur untuk memastikan pemenuhan standar pengembangan software, jika software tersebut diaplikasikan, dan
7. Mekanisme pengukuran dan laporan.

# Biaya penerapan SQA

**1. *Prevention Cost*** : Jumlah yang dikeluarkan untuk memastikan bahwa semua praktik jaminan kualitas diikuti dengan benar.

**2. *Appraisal Cost*** : Jumlah uang yang dikeluarkan untuk merencanakan semua kegiatan pengujian dan kemudian melaksanakannya seperti mengembangkan kasus uji dan kemudian melaksanakannya.

**3. *Internal Failures*** : Biaya yang muncul ketika kasus uji dieksekusi untuk pertama kalinya di tingkat internal dan beberapa di antaranya gagal.

**4. *External Failures*** : Biaya yang terjadi ketika cacat ditemukan oleh pelanggan, bukan *tester*.

The slide features decorative geometric shapes in the corners. On the left, there are overlapping triangles in shades of green, blue, orange, and purple. On the right, there are similar overlapping shapes in shades of green, blue, purple, and orange. The central text is black and bold.

2.

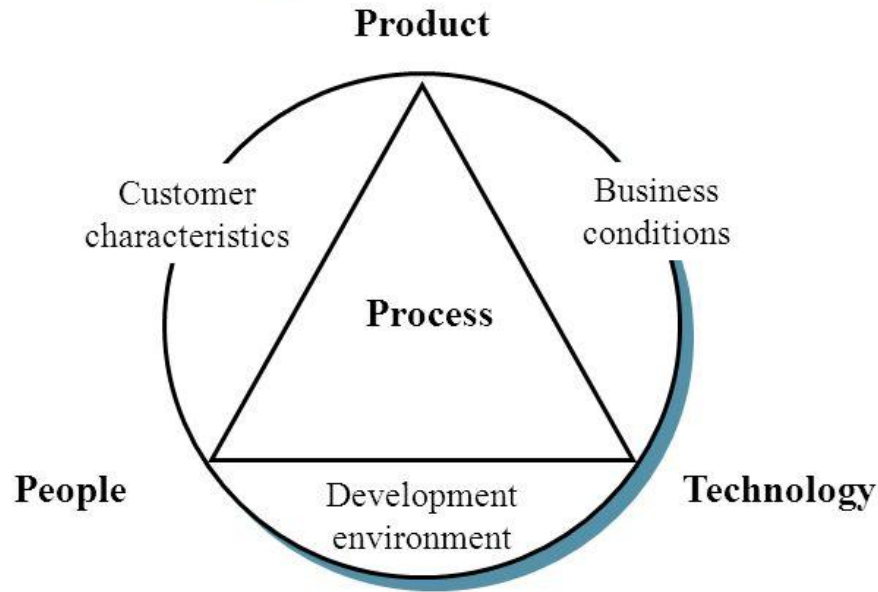
## **Kerangka Teknis untuk *Metric Software***



Metrik perangkat lunak ada untuk mengukur efektivitas proses perangkat lunak dan proyek-proyek yang dilakukan dengan menggunakan proses sebagai kerangka.

Pada domain proses, metrik perangkat lunak membantu mengembangkan seluruh proyek perangkat lunak tersebut dengan jangka waktu tertentu. Pada domain proyek, metrik perangkat lunak membantu dalam perhitungan, kontrol kualitas, perkiraan produktivitas, dan kontrol proyek.

# Metrik Proses Pengembangan Perangkat Lunak



# Kategori metrik perangkat lunak

1. *Size oriented -> Kilo Line of Code*
2. *Function oriented -> Function Point*
3. *Object oriented*

## ***Size oriented***

Diperoleh melalui normalisasi kualitas dan/atau produktivitas tindakan dengan mempertimbangkan ukuran perangkat lunak yang dihasilkan. Seribu baris kode (*Kilo Line of Code/KLoC*) dipilih sebagai nilai normalisasi.

Metrik mencakup :

1. Kesalahan per KLoC
2. Cacat per KLoC
3. Dolar per KLoC
4. Halaman dokumentasi per KLoC

## ***Function oriented***

Metrik berorientasi fungsi menggunakan ukuran fungsi yang ditunjukkan oleh aplikasi sebagai nilai normalisasi.

Umumnya metrik ini menggunakan rumus :

$$\mathbf{FP = Total * [0.65 + 0.01 * jumlah\ (nilai\ Faktor)]}$$

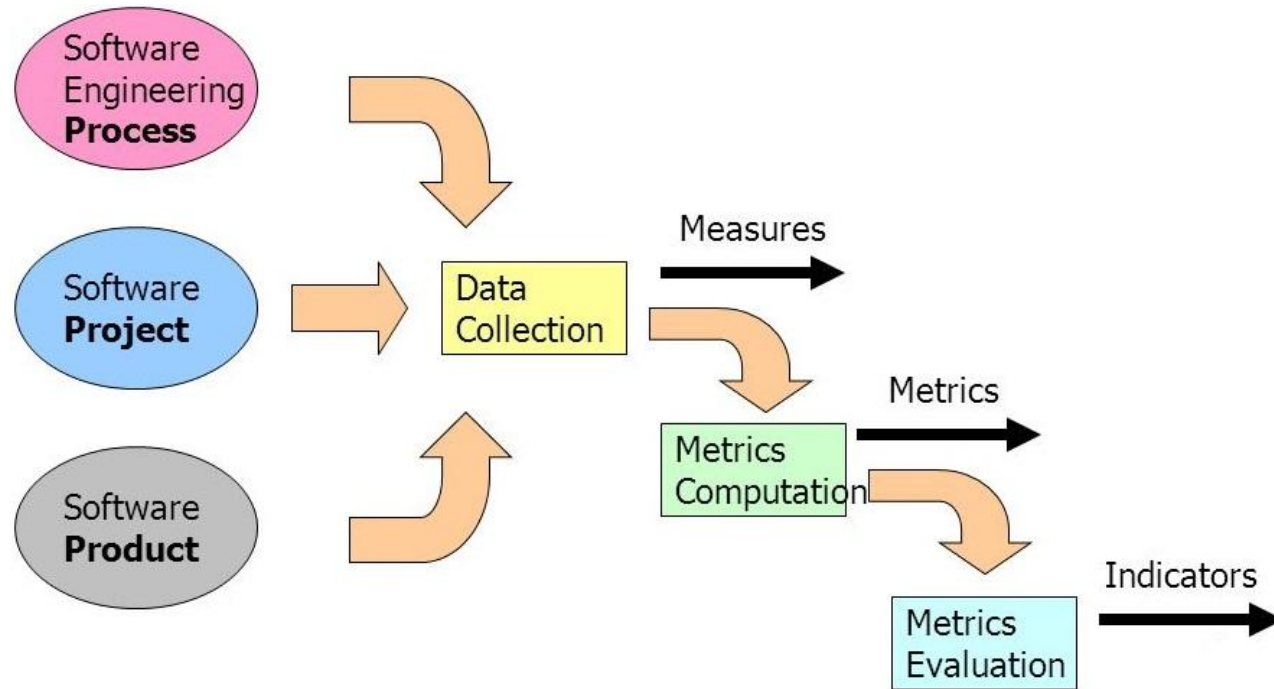
Nilai-nilai *function point* pada proyek-proyek masa lalu dapat digunakan untuk perhitungan, misalnya, rata-rata jumlah baris kode per titik fungsi.


## ***Object oriented***

Ditinjau dari :

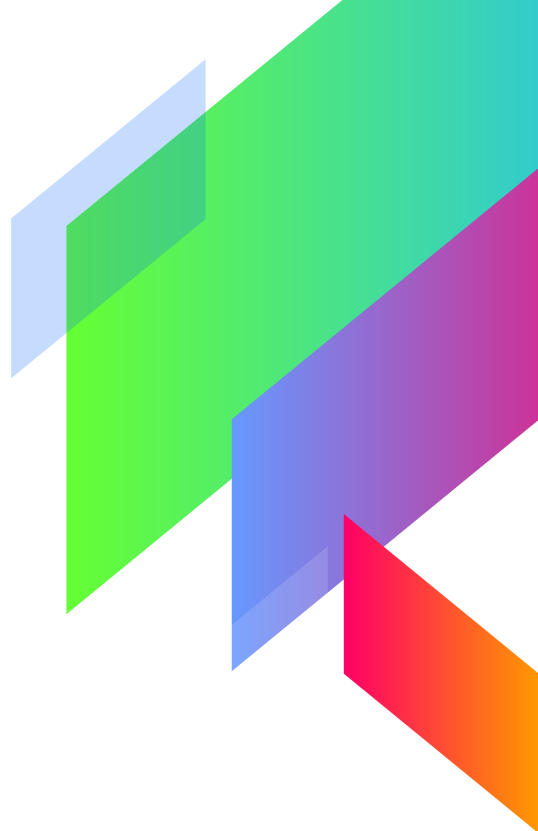
1. Jumlah kasus uji
2. Jumlah kelas kunci
3. Jumlah skrip skenario
4. Jumlah rata-rata kelas dukungan/kelas utama
5. Jumlah subsistem

# Proses Dasar Metrik Perangkat Lunak





### **3. Verifikasi & Validasi dan Pengornasian Pengujian Perangkat Lunak**





## 3.1 Verifikasi & Validasi

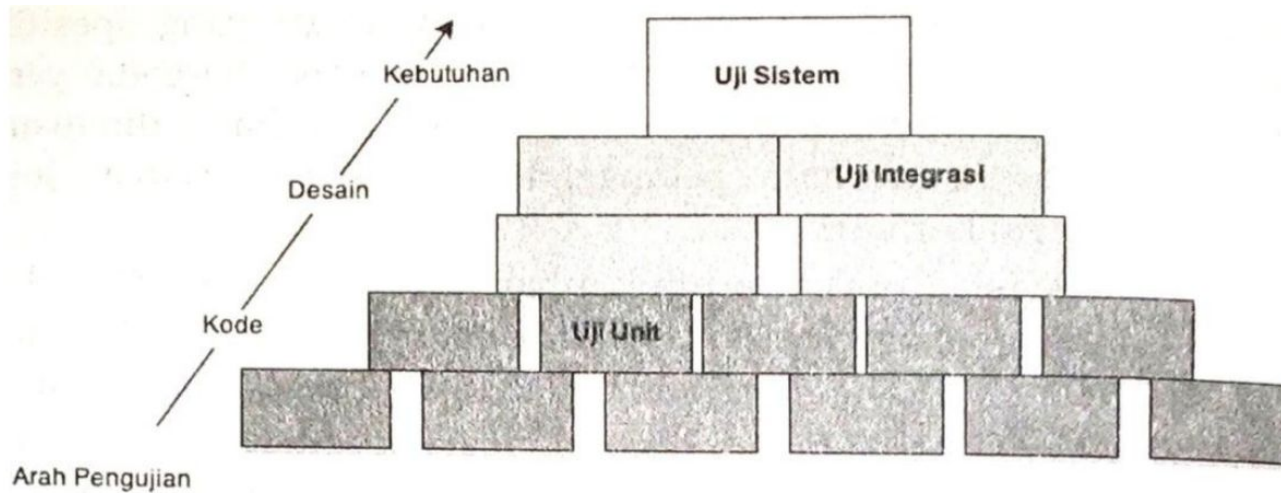
Pengujian perangkat lunak adalah sebuah elemen topik yang memiliki cakupan luas dan sering dikaitkan dengan verifikasi dan validasi.

- Verifikasi mengacu pada sekumpulan aktivitas yang menjamin bahwa perangkat lunak mengimplementasikan dengan benar sebuah fungsi yang spesifik.
- Validasi mengacu pada sekumpulan aktivitas berbeda yang menjamin bahwa perangkat lunak yang dibangun dapat ditelusuri sesuai dengan kebutuhan pelanggan.

# Lingkup dari Verifikasi & Validasi :

1. Peninjauan teknis formal
2. Audit kualitas dan konfigurasi
3. *Monitoring*
4. Simulasi
5. Studi kelayakan
6. Peninjauan dokumentasi
7. Peninjauan basis data
8. Analisis algoritma
9. Pengujian proses pengembangan perangkat lunak
10. Pengujian kualifikasi
11. Pengujian instalasi

# Ilustrasi pengujian verifikasi




# Pendekatan pengujian validasi



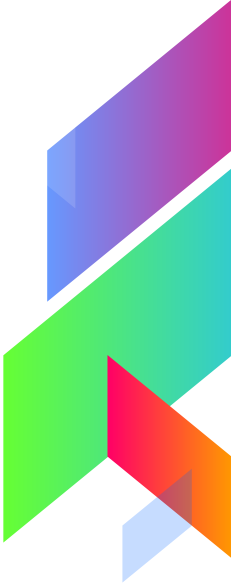
### 3.2 Pengorganisasian Pengujian Perangkat Lunak

Proses pengujian sebuah perangkat lunak sebaiknya melibatkan pihak ketiga yang memang secara khusus bertanggung jawab untuk melakukan proses pengujian secara independen. Untuk itulah diperlukan Independent Test Group (ITG).

Peran dari ITG adalah untuk menghilangkan "*conflict of interest*" yang terjadi ketika pengembang perangkat lunak berusaha untuk menguji produknya sendiri.



Walaupun seperti itu, sering terjadi beberapa kesalahan pemahaman berkaitan dengan peran ITG, antara lain:

1. Pengembang tidak boleh melakukan pengujian sama sekali.
  2. Perangkat lunak dilempar begitu saja untuk diuji secara sporadis.
  3. Penguji tidak terlibat pada proyek sampai tahap pengujian dimulai.
- 

# Referensi

S, Rosa A. & Shalahuddin, M. 2018. *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek Edisi Revisi*. Bandung : Informatika  
<https://medium.com/@makersinstitute/mengenal-lebih-jauh-tentang-software-quality-assurance-844361cd50db>  
[https://toghr.com/id\\_ID/pengertian-dari-software-quality-assurance/](https://toghr.com/id_ID/pengertian-dari-software-quality-assurance/)

Nugroho, Eddy Prasetyo dkk. 2009. *Rekayasa Perangkat Lunak*. Bandung : Politeknik Telkom  
<https://docplayer.info/42040924-Metrik-proses-dan-proyek-perangkat-lunak-karmilasari.html>



# Terima kasih!

Segala materi presentasi rencana pembelajaran semester mata kuliah Rekayasa Perangkat Lunak Teknik Informatika UPN “Veteran” Jawa Timur 2019 dapat dengan mudah ditemukan di :

**[github.com/rakhid16/rpl](https://github.com/rakhid16/rpl)**