

Studi Literatur: Pengujian Perangkat Lunak

Nama Andika

mhdandika03@polgan.ac.id

Politeknik Ganesha Medan

Abstrak

Membuat sebuah aplikasi/software yang berkualitas menurut saya tidaklah mudah, Semakin berkualitas suatu perangkat lunak atau aplikasi maka akan semakin mahal harga yang dijual. Perangkat lunak yang berkualitas tidak hanya sekedar mencari keuntungan belaka akan tetapi mutu dari kualitas dan kuantitasnya yang sudah menjadi jaminan dari aplikasi yang ditawarkan. Pembuatan suatu perangkat lunak yang berkualitas selalu memperhatikan kebutuhan akan penggunaannya, sehingga mampu memberikan solusi dari permasalahan yang dihadapi oleh para penggunaannya. Selain itu, suatu aplikasi yang berkualitas selalu melalui tahapan –tahapan pengujian yang tidak singkat sehingga benar –benar menjamin akan kualitas produknya. Nilai yang diperoleh dari pengujian untuk kelengkapan fitur diperoleh nilai standard error terkecil Pengujian perangkat lunak menurut saya sangat penting karena dijamin sekarang banyak yang tidak melakukan pengujian perangkat lunaknya sebelum di gunakan, karena pengujian perangkat lunak ini untuk melihat kualitas dari aplikasi/perangkat lunak yang telah dibuat berjalan dengan lancar atau tidak, jika terdapat masalah maka selanjutnya akan di proses untuk dibenahi sisi error atau ketidak teapatan fungsi dari software tersebut oleh pembuat aplikasi/software tersebut Dari masalah tersebut, maka dibutuhkan beberapa metode pengujian yang tepat untuk menguji aplikasi/software. Ada beberapa metode untuk melakukan pengujian diantaranya seperti metamorphic testing, random testing, adaptive random testing, state based testing, random sequence based testing.

Kata Kunci : *Metode Pengujian, Bahasa Pengujian, API*

PENDAHULUAN

Perkembangan teknologi pada saat ini telah memberikan dampak yang pesat pada kemajuan di berbagai aspek sosial. Tuntutan kebutuhan mengenai informasi, yang terkait dengan era globalisasi menjadi meningkat pesat dan bervariasi. Teknologi yang tadinya memanfaatkan suatu arsitektur yang besar sekarang dengan berkembang dan majunya teknologi, sistem tersebut bergerak ke dalam bentuk microservice. Microservice adalah suatu kumpulan proses independen yang saling berkomunikasi antara satu dengan yang lain bertujuan untuk membentuk aplikasi kompleks terhadap bahasa API apapun. API atau sering disebut dengan *Application Programming Interface* adalah suatu wadah yang menjembatani antara pengguna dengan sistem.

Beberapa metode pengujian yang sering digunakan adalah *metamorphic testing*, *random testing*, *adaptive random testing*. Metode *random testing* sering digunakan dalam industri, karena kesederhanaannya. Secara umum, pendekatan pengujian yang lain memerlukan teknik pengujian yang profesional karena lebih kompleks. Jika pada *metamorphic testing* user dapat memasukkan variable yang diinginkan, sehingga variable tersebut tidak random sedangkan jika pada *adaptive random testing* masukan variable random, tetapi keluaran yang sudah muncul tidak dimunculkan kembali, sehingga tidak ada data yang sama atau redundan. Pengujian *metamorphic* memberikan pandangan, jika kebenaran tidak ditentukan dengan memeriksa output dari suatu hasil tetapi dengan menerapkan transformasi input dan mengamati output yang dihasilkan sehingga valid atau tidak.

TINJAUAN PUSTAKA

Dalam melakukan menyusun studi literatur ini, penulis menggunakan suatu metode. Metode tersebut adalah dengan mengumpulkan beberapa paper dengan topik yang sejenis. Hal ini dilakukan,

*penulis korespondensi



agar pembaca dapat lebih mudah dan memahami maksud dari penulis. Kemudian dari paper yang sudah didapat disaring dengan cara *inhale* dan *exhale*. Paper yang membahas topik yang sama akan masuk ke kriteria *inhale*, sedangkan paper yang membahas topik berbeda akan masuk ke kriteria *exhale*.

Topik yang akan dibahas disini adalah mengenai software testing, lebih detailnya mengenai pentingnya melakukan software testing, metode software testing, bahasa software testing.

METODE PENELITIAN

Melakukan proses pengujian, merupakan hal yang penting dari suatu aplikasi atau produk sebelum muncul ke pasaran. Proses pengujian ini bertujuan untuk memastikan kualitas dan keandalan aplikasi tersebut

Aplikasi berbasis *object oriented* semakin berkembang dan banyak digunakan. Namun, pengujian pada *object oriented program* menghadapi tantangan untuk menerapkan pengujian tradisional ke *object oriented program*. Pada *object oriented program* terdapat beberapa karakteristik khusus dari segi bahasa seperti enkapsulasi, polimorfisme, dan pewarisan .

Sedangkan karakteristik web API adalah terbuka, kolaboratif, dan dinamis meningkatkan ancaman baru pada kualitas sistem. Hal itu disebabkan API yang terbuka untuk akses terbuka oleh sejumlah besar pengguna di internet, kesalahan dalam API yang bersifat publik dapat tersebar luas. Hal ini juga menyebabkan kegagalan perangkat lunak dalam skala besar. Pengujian API dengan demikian menjadi perlu untuk memastikan kualitas dan kendalan API.

HASIL DAN PEMBAHASAN

Metode Pengujian

Metamorphic Testing

Metamorphic testing diusulkan untuk mengatasi masalah-masalah bug yang terjadi pada oracle , karena saat menguji program tanpa uji oracle adalah hal yang mustahil untuk mengetahui apakah test case menyebabkan kegagalan atau tidak. MT menggunakan *metamorphic relation* (MR) untuk menentukan kasus uji atau *test case*, yang nantinya akan menghasilkan test-test dari *case* sumber tes input itu sendiri . Kasus uji ini berguna untuk memudahkan pengujian dalam melakukan pengujian. Contohnya adalah pengujian pada masukan input di spotify apakah outputnya sesuai dengan hasil input atau tidak.

Random Testing

Random Testing adalah metode pengujian perangkat lunak yang banyak digunakan karena kemudahan dan kepraktisannya. Metode ini sangat sederhana dan mudah untuk di otomatissi .

Random Testing dilakukan dengan cara memasukkan variabel masukan (test case) dan akan di seleksi secara acak. Dari inputan tersebut, akan didapatkan hasil yang dimana akan ada proses pengacakan dulu. Metode ini merupakan metode paling mudah dan simple, untuk itu pengujian ini tidak cocok dengan aplikasi berskala besar.

Adaptive Random Testing

Adaptive Random Testing memiliki fokus yang mempengaruhi jarak pada kinerja pengujian acak dan efisiensi pendeteksian kesalahan . Cara pengujian adaptive random testing ini dengan memberikan input uji yang acak dan menyebar menggunakan pengukuran jarak antara input yang berurutan , sehingga hasil input yang salah dapat dikelompokkan. Kesalahan mendeteksi kemampuan adaptif pengujian acak yang terjadi dipengaruhi oleh kesalahan penghitungan jarak dan kesalahan pemodelan pola kegagalan blok. Metode adaptive random testing sudah mengalami perkembangan.

*penulis korespondensi



Mirror Adaptive Random Testing

Mirror Adaptive Random Testing merupakan pembaharuan metode dari *adaptive random testing*. Sekilas cara kerja metode ini sama, yaitu dengan menggunakan kasus uji (*test case*). Metode ini pun tetap menggunakan algoritma *adaptive random testing* sebagai acuannya. Perbedaannya adalah pada metode ini kasus uji dibagi menjadi beberapa subdomain, dan salah satu subdomain dijadikan subdomain utama dan dapat dikerjakan dengan menggunakan metode *adaptive random testing*. Jika hasil dari kasus uji subdomain utama tersebut tidak ditemukan eror, maka kasus uji diteruskan dengan cara mencerminkannya pada subdomain yang lain. *State Based Testing*

State-based testing adalah suatu metode baru untuk menguji program berorientasi objek. terbatas. Setiap kelas yang diuji diibaratkan sebagai pemetaan dari status awal ke status penyelesaian tergantung pada parameter yang diteruskan. Jadi metode ini melakukan pengujian dengan memvalidasi interaksi antara operasi dan keadaan suatu objek. Tetapi *state based testing* berkaitan dengan nilai aktual yang disimpan oleh atribut objek. Hal ini tidak secara langsung berkaitan dengan validasi kombinasi nilai parameter tertentu, meskipun nilai ini digunakan untuk memvalidasi transisi status objek,

Metode *state-based testing* ini memiliki beberapa kelebihan dibandingkan dengan pemodelan konseptual, khususnya kemudahan manipulasi nilai-nilai fisik dan kemandirian setiap operasi dari operasi lain yang disediakan oleh suatu objek.

Random Sequence-based Testing

Adaptive Random Testing (ART) merupakan suatu metode untuk pengujian dengan menyebarkan input tes secara merata ke seluruh domain input menggunakan metrik kesamaan atau tidak kesamaan. Hasil dari ART tidak hanya untuk menghasilkan urutan sendiri dari kasus uji, tetapi juga dapat memberikan rangkaian uji untuk meningkatkan peluangnya mendeteksi kegagalan sebelumnya, dengan urutan terurut yang disebut dengan *Adaptive Random Sequence* (ARS).

ARS bekerja dengan menyebarkan seluruh domain input dan lebih efektif untuk menemukan kesalahan atau kegagalan lebih cepat. Penerapan metode ARS pada pengujian regresi, dan dapat berupa urutan alternatif ke urutan acak sederhana, yang biasanya digunakan pada pengujian regresi. Untuk itu ARS sangat efektif untuk meningkatkan kinerja pengujian regresi untuk OOS. Metode ini dapat diuji berdasarkan :

1. Clustering

Clustering adalah proses untuk mengelompokkan data ke beberapa *cluster* (kelompok) sehingga data dalam satu *cluster* memiliki tingkat kemiripan yang banyak dan data antar *cluster* memiliki tingkat kemiripan yang sedikit.

Clustering dilakukan dengan proses partisi satu set objek data ke dalam himpunan bagian yang disebut *cluster*. Objek dalam *cluster* ini memiliki kemiripan karakteristik antara satu dengan lainnya dan berbeda dengan *cluster* lain. Partisi dilakukan menggunakan algoritma *clustering*. Sehingga *clustering* sangat berguna dan bisa menemukan kelompok yang tidak dikenal dalam data. *Clustering* banyak digunakan dalam berbagai aplikasi seperti pada aplikasi *business intelligence*, pengenalan pola citra, *web search*, bidang ilmu biologi, dan bidang keamanan (*security*). Ada beberapa metode algoritma pada *clustering*, misalnya adalah algoritma Kmeans dan K-medoids.

2. K-means

K-means adalah suatu algoritma bagian dari *clustering* dengan menggunakan metode analisis kelompok yang mengarah pada pemartisian N objek pengamatan ke dalam K kelompok, dimana setiap objek pengamatan dimiliki oleh sebuah kelompok dengan nilai *mean* (rata-rata) terdekat. Dimana keduanya akan menemukan pusat dari kelompok dalam data sebanyak iterasi perbaikan yang dilakukan.

*penulis korespondensi



3. *K-medoids*

K-medoids adalah suatu algoritma atau sering disebut dengan PAM (*Partitioning Around Medoids*) menggunakan metode partisi *clustering* untuk mengelompokkan sekumpulan n objek menjadi sejumlah k *cluster*. Objek yang menjadi wakil dari sebuah *cluster* disebut dengan *medoid*. *Cluster* dibangun dengan menghitung kedekatan yang dimiliki antara *medoid* dengan objek *non-medoid*. Metode ini dapat mengerjakan inputan berupa numerik .

BAHASA PENGUJIAN

Bahasa yang dimaksud adalah bahasa pemrograman untuk software testing itu sendiri. Ada beberapa jenis bahasa pemrograman yang biasa digunakan untuk membuat program software testing diantaranya adalah :

Bahasa C#

Bahasa C# adalah bahasa yang dikembangkan oleh Microsoft. Bahasa ini banyak digunakan, karena kemudahan untuk memahami dan prakteknya. Bahasa ini adalah bahasa pemrograman yang berfokus pada orientasi objek. Bahasa pemrograman ini membantu untuk proses pengujian yang membutuhkan objek, karena lebih memudahkan dalam prosesnya. Hasil yang didapatkan juga lebih mudah, karena memiliki fokus yang sama yaitu berorientasi objek

Javascript

Javascript menurunkan bahasa dengan format data JSON, keunggulan JSON memiliki banyak format yang dapat diakses dengan bahasa pemrograman Python, Ruby, PHP, dan Java. Selain itu, JSON memiliki library untuk membantu pemrograman khususnya untuk pengujian menggunakan metode yang memanfaatkan API suatu website dengan menyediakan fungsi yang lebih fleksibel dan umumnya memilih eksekusi interpretative untuk bahasa script yang dapat menyederhanakan pemrosesan respon . JAVA

JAVA adalah Bahasa pemrograman yang paling banyak digunakan. Dengan Bahasa pemrograman JAVA, dapat memberikan hasil berupa nilai-nilai yang dapat membedakan antara yang asli dan mutase . Untuk membantu memudahkan mendapatkan hasil pada proses pengujian dibutuhkan test case, yang kemudian akan di proses dengan Bahasa pemrograman ini. Sehingga hasil dari proses pengujian lebih akurat dan memberikan kemudahan selama proses pengujian berlangsung. Kebanyakan bahasa ini dipadukan dengan metode pengujian *metamorphic testing*.

KESIMPULAN

Pengujian perangkat lunak sangat penting untuk digunakan untuk melihat kualitas dari software yang telah dibuat apakah ada kendala atau tidak dalam pengoperasiannya Pengujian perangkat lunak sukses apabila telah mencapai :

1. Memenuhi kebutuhan (*requirement*) yang mendasari perancangan dan pengembangan perangkat lunak
2. . Berjalan sesuai dengan yang diharapkan.
3. Dapat diterapkan menggunakan karakteristik yang sama.
4. Memenuhi kebutuhan semua pihak yang berkepentingan.

DAFTAR PUSTAKA

<http://e-journal.unipma.ac.id/index.php/assets/article/view/1188>

https://repository.usd.ac.id/32377/2/055314010_Full.pdf

<https://unisbank.ac.id> > ojs > sintak > article > view

<https://journal.uny.ac.id>

*penulis korespondensi



<http://tip.ppj.unp.ac.id/index.php/tip/article/view/215>
<https://media.neliti.com/media/publications/146884-ID-pengujian-perangkat-lunak-dengan-menggun.pdf>

*penulis korespondensi



This is an Creative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.