

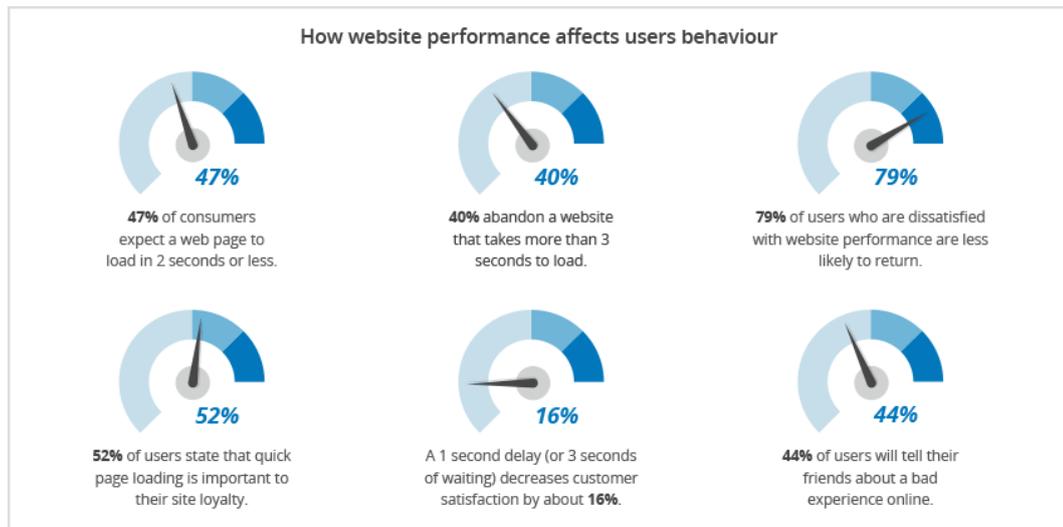
BAB I

PENDAHULUAN

1.1 Latar Belakang

Error ataupun *bug* pada sebuah *software* merupakan bagian yang tidak dapat dihindari dalam pengembangan sebuah *software*. *Error* atau *bug* ini sendiri tidak sepenuhnya dapat dihilangkan, akan tetapi dapat di meminimalisir kemungkinan terjadinya *error* hingga sekecil mungkin [1]. *Error* ataupun *bug* yang terjadi pada sebuah *software* dapat mengganggu produktivitas, bahkan pada skala bisnis ataupun perusahaan, *error* yang terjadi pada *software* dapat berdampak pada kualitas dari *software* tersebut. Pada beberapa kasus *error* pada *software* dapat mengakibatkan kerugian dari aspek finansial yang dapat mengancam keberlangsungan dari sebuah bisnis ataupun perusahaan yang menyediakan *software* tersebut [2]. Berdasarkan hasil analisis dari *The Consortium for Information and Software Quality*, kualitas *software* yang buruk merugikan perusahaan-perusahaan di Amerika Serikat hingga \$2,08 triliun per tahun. Kerugian tersebut termasuk biaya dari kegagalan operasional, proyek yang gagal, dan kesalahan *software* pada sistem lama. Sebanyak 47% proyek pengembangan *software* memiliki permasalahan pada timescales dan juga budgets, dari permasalahan tersebut akan menyebabkan hasil *software* yang telah dikembangkan memiliki kualitas yang buruk atau *low-quality product* [3].

UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 1. 1 Data Perilaku Pengguna Ketika Menemui Error

Sumber: [4]

Berdasarkan data pada gambar 1.1 yang diperoleh dari situs resmi Microsoft menunjukkan perilaku pengguna ketika menemui *software* ataupun *website* yang memiliki performa serta kualitas yang buruk. Dari data tersebut terlihat bahwa sekitar 79% pengguna akan meninggalkan *software* ataupun *website* yang memiliki performa serta kualitas yang buruk. Hasil yang ditunjukkan oleh data tersebut berhubungan dengan analisis ketika sebuah *software* memiliki kualitas yang buruk akan berdampak pada kerugian perusahaan pada aspek finansial dikarenakan tingkat penunjang yang rendah atau semakin menurun setiap waktunya.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A

What is the biggest barrier to finding and fixing bugs in your backlog faster?



Gambar 1. 2 Persentase Siklus Pengujian

Sumber: [5]

Kendala terbesar yang dihadapi saat proses produksi sebuah *software* berdasarkan data pada gambar 1.2 yang diperoleh dari Undo.io adalah sebagian besar waktu digunakan untuk menemukan *bug* pada *software*. Hal tersebut tentunya tidak efektif dalam perusahaan maka dari itu penggunaan *automation* menjadi salah satu jawaban untuk mengurangi waktu yang diperlukan dan juga sebagai baris pertahanan untuk menguji kualitas *software* yang dikembangkan untuk mengurangi kemungkinan terjadinya *bug* ataupun *error* ketika digunakan oleh pengguna [6].

PT. Fata Organa Solusi merupakan sebuah perusahaan startup asal Indonesia yang berfokus pada bidang teknologi khususnya pada pengembangan *software* untuk menunjang *artificial intelligence* dalam mengenali emosi khususnya emosi dari manusia. Perusahaan memiliki banyak produk yang dikembangkan secara bersamaan antara lain yaitu seperti *Beluga Box*, *Auto ML*, *Annotation Tool* dan lainnya. Pada perusahaan PT. Fata Organa Solusi ini sendiri juga mengalami permasalahan yang hampir serupa yaitu perusahaan memiliki banyak produk yang sedang dikembangkan dan setiap produknya memiliki fitur yang banyak pula [7] khususnya pada produk *Annotation Tool*. Untuk dapat memastikan produk yang dikembangkan tetap memiliki kualitas yang bagus demi mempertahankan

keberlangsungan perusahaan maka tentunya perusahaan melakukan pangujian fungsional.

Annotation Tool adalah sebuah *software* berbasis *website* yang dikembangkan oleh perusahaan untuk dapat menunjang tugas yang dikerjakan oleh tim *Annotator*. *Annotator* sendiri merupakan sebuah tim yang dibentuk oleh perusahaan untuk mengidentifikasi emosi dan juga memberikan label pada data dalam bentuk audio yang telah disediakan. Cara kerja dari *Annotation Tool* adalah dimana pengguna dapat mengunggah *file* dalam bentuk audio dan juga csv data. Setelahnya pengguna dapat membuat sebuah *project* dengan data yang telah di unggah tersebut. Setelah *project* terbentuk maka pengguna sudah dapat memberikan label pada data yang berada dalam *project* tersebut. Berikutnya pengguna dapat melakukan validasi pada data yang telah diberikan label. Terakhir pengguna dapat mengunggah data yang telah di validasi tersebut ke *Auto ML* untuk diproses lebih lanjut.

Pada kondisi tersebut permasalahan muncul dimana pengujian secara manual akan menjadi sulit dilakukan apabila sebuah *software* tersebut memiliki terlalu banyak fitur sehingga untuk dapat menjalankan semua proses pengujian memerlukan waktu yang cukup banyak dimana perusahaan memiliki waktu yang sangat terbatas untuk dapat melakukan semua pengujian tersebut, belum lagi ditambah dengan faktor pengembangan produk yang terus dilakukan secara berkala sehingga perlu dilakukan pengujian ulang untuk memastikan bahwa pembaharuan tersebut tidak mempengaruhi fitur ataupun fungsi *software* yang telah berjalan. Selain itu tingkat konsistensi yang sulit untuk di pertahankan karena dilakukan oleh manusia secara manual, faktor efisiensi dan juga sumber daya yang diperlukan juga menjadi kendala yang perlu di ditangani untuk dapat menjaga kualitas dari *software* yang dikembangkan [8].

Proses pengujian yang dikerjakan pada perusahaan PT. Fata Organa Solusi sudah dirasa terlalu banyak dan akan sulit untuk dikerjakan secara manual dalam waktu yang singkat maka solusi untuk dapat mengatasi beberapa tantangan yang

dihadapi pada perusahaan tersebut adalah dengan mengimplementasikan otomatisasi pada tahap pengujian atau yang disebut *automation testing*. *Automation testing* secara sederhana dapat diartikan sebagai sebuah rangkaian pengujian *software* yang dimana skenario pengujian akan dijalankan secara otomatis dengan bantuan *tools* ataupun *script* pemrograman pada produk ataupun *software* yang akan diuji untuk dapat mengidentifikasi kekurangan dan kesalahan yang mungkin terjadi pada *software* tersebut [9].

Pada tahap pengujian sebuah *software* tentunya terdapat metode yang digunakan, dalam penelitian ini metode pendekatan yang digunakan dalam tahapan pengujian adalah metode *black box testing*. Fokus utama dari pendekatan *black box testing* adalah pada fungsionalitas dan perilaku eksternal dari sebuah *software* tanpa memperhatikan struktur internalnya [10]. Pendekatan ini sangat sesuai dengan kebutuhan PT. Fata Organa Solusi untuk memastikan bahwa perangkat lunak yang dihasilkan memenuhi spesifikasi dan kebutuhan pengguna. Dengan *black box testing*, perusahaan dapat mengidentifikasi potensi kegagalan fungsional dan memastikan kualitas *software* sebelum dirilis ke pengguna akhir. Selain itu, *black box testing* juga membantu mengurangi bias yang mungkin muncul dari pengetahuan internal tentang kode, sehingga memberikan perspektif pengujian yang lebih objektif.

Ketika sebuah perusahaan ingin mengimplementasi *automation testing* tentunya diperlukan sebuah analisis terlebih dahulu terkait *automation testing tool* yang sesuai untuk diterapkan pada sebuah Perusahaan. Berdasarkan penelitian dari Fatima Azzam beserta timnya menunjukkan beberapa kriteria yang perlu dipertimbangkan dalam mengadaptasi sebuah *automation testing tool* seperti tujuan pengujian, ketersediaan sumber daya, kompatibilitas, skalabilitas, kemudahan pengguna, kemampuan skripting dan juga dukungan komunitas [11]. Dari beberapa kriteria tersebut akan disesuaikan dengan kebutuhan perusahaan sebagai bahan evaluasi untuk penentuan *automation testing tool* dengan hasil kriteria yang dibutuhkan adalah *tools* yang dapat digunakan untuk pengujian fungsional, bersifat open source, dapat menguji *software* berbasis web, *tool* yang

dapat menangani pengujian yang kompleks, mudah digunakan, dapat mendukung pembuatan *script* yang fleksibel, dan juga dukungan dari komunitas yang luas dan terus berkembang. Berdasarkan semua kriteria tersebut, *tools* yang memenuhi untuk di implementasikan pada perusahaan PT. Fata Organa Solusi adalah Selenium dan juga *Cypress* [12]. Terdapat 2 *tools* yang dapat di implementasikan oleh perusahaan PT. Fata Organa Solusi akan tetapi perusahaan tentunya tidak dapat mengadopsi keduanya maka dari itu penelitian ini dilakukan sebagai langkah lanjutan untuk dapat menghasilkan perbandingan antara Selenium dan juga *Cypress*.

Terdapat beberapa parameter yang akan digunakan untuk mengevaluasi masing-masing *automation testing tool* yang akan di implementasikan pada PT. Fata Organa Solusi antara lain yaitu kemampuan pengujian paralel, durasi pengujian, dan banyaknya *test case* yang dapat di cakup. Beberapa parameter tersebut ditentukan berdasarkan kepentingan dari perusahaan sehingga parameter tersebutlah yang akan dijadikan tolak ukur dalam mengvaluasi *automation testing tool*. Pengujian otomatis akan dilakukan dengan menjalankan masing-masing *automation testing tool* untuk melakukan *test case* yang telah dirancang kemudian hasil yang diperoleh akan dianalisis dengan membandingkan keduanya [13].

Sebelum menjalankan pengujian dengan menggunakan *automation testing tool* pada sebuah *software*, perlu dilakukan penyusunan *test document* terlebih dahulu. *Test document* merupakan sebuah dokumen yang berisi informasi rinci tentang rencana, strategi, lingkup, dan hasil dari seluruh proses pengujian perangkat lunak [14]. Dokumen ini sering kali disusun sebelum memulai tahap pengujian dan terus diperbarui sepanjang siklus pengembangan perangkat lunak. Dalam *test document* tentunya terdapat *test scenario* dan juga *test case* yang merupakan bagian dari *test document*. *Test scenario* sendiri merupakan sebuah deskripsi umum tentang langkah-langkah pengujian, biasanya digunakan sebagai gambaran tentang apa yang akan diuji dan untuk mengidentifikasi area pengujian utama. *Test case* merupakan rincian spesifik dari langkah-langkah pengujian yang akan dilakukan, biasanya digunakan sebagai panduan dalam melakukan pengujian

sebuah *software*. Tujuan dari perancangan *test scenario* dan *test case* adalah untuk memastikan bahwa *software* berperilaku sesuai dengan yang diharapkan dan memenuhi persyaratan fungsional atau non-fungsional yang telah ditetapkan.

Dalam perancangan *test scenario* yang akan digunakan dalam penelitian ini juga tidak dapat dirancang dengan sembarangan, tentunya ada beberapa faktor yang perlu diperhatikan dalam melakukan perancangan *test scenario*. Berdasarkan penelitian dari Muhammad Sholeh terdapat beberapa teori yang akan digunakan dalam penelitian ini untuk melakukan perancangan *test scenario* sehingga menghasilkan *scenario* yang tepat dan juga efisien dalam proses pengujian. Teknik pengujian yang akan digunakan dalam penelitian adalah teknik *equivalence class partitioning* dan dikombinasikan dengan teknik *boundary value analysis* [15]. Penggunaan kedua teknik tersebut bertujuan untuk merancang *test scenario* dengan akurasi yang lebih tepat sesuai dengan apa yang akan dilakukan oleh pengguna.

1.2 Rumusan Masalah

Mendasari latar belakang yang telah dikemukakan pada bagian sebelumnya maka dari itu terdapat beberapa rumusan masalah yang dapat disimpulkan sebagai berikut:

1. Bagaimana penerapan teknik *equivalence class partitioning* dikombinasikan dengan *boundary value analysis* dalam merancang test case?
2. Bagaimana hasil perbandingan dari *Selenium* dan *Cypress* dalam menjalankan pengujian pada *software Annotation Tool*?

1.3 Batasan Masalah

Terdapat beberapa batasan masalah yang ditetapkan supaya penelitian yang dilakukan dapat lebih terfokus pada bagian permasalahan yang ingin diselesaikan, batasan masalah yang ditetapkan antara lain sebagai berikut:

1. Pengujian akan dilakukan pada *platform* komputer/laptop dengan spesifikasi yang ditentukan.
2. Pendekatan yang digunakan berupa metode *black box testing*.
3. Teori pengujian yang akan digunakan adalah *equivalence class partitioning* dan juga *boundary value analysis*.
4. Pengujian yang dilakukan berupa *functional testing* dengan menjalankan *test case* pada 2 fitur utama dari *Annotation Tool* pada setiap *framework*.
5. Pengujian dilakukan dengan 2 *automation framework* yaitu *Selenium* dan juga *Cypress*.
6. Pengujian akan dilakukan pada lingkup *annotator side* dari *Annotation Tool*.
7. Parameter perbandingan yang akan diukur yaitu kemampuan pengujian paralel, durasi pengujian, dan banyaknya *test case* yang dapat di cakup.
8. Produk yang akan diterapkan *automation testing* adalah *Annotation Tool* (*software* berbasis web).

1.4 Tujuan dan Manfaat Penelitian

1.4.1 Tujuan Penelitian

1. Merancang *test case* yang akurat berdasarkan teori *equivalence class partitioning* dan *boundary value analysis* untuk pengujian fungsional pada *software Annotation Tool* pada PT. Fata Organa Solusi.
2. Melakukan analisis perbandingan *automation testing tool Selenium* dan juga *Cypress* dengan pengujian fungsional pada *software Annotation Tool* pada PT. Fata Organa Solusi.
3. Mengembangkan program automation dengan *library Selenium* dan juga *Cypress* untuk melakukan pengujian pada *software Annotation Tool* dari perusahaan PT. Fata Organa Solusi.

1.4.2 Manfaat Penelitian

1. Meningkatkan kualitas produk *Annotation Tool* dengan melakukan *functional testing* berdasarkan *testcase* yang dirancang.

2. Mengetahui perbandingan antar *automation testing tool* untuk *functional testing* pada produk Annotation Tool.
3. Menghasilkan sebuah program automation yang dapat digunakan untuk melakukan testing pada *software* yang dikembangkan oleh perusahaan PT. Fata Organa Solusi.

1.5 Sistematika Penulisan

BAB 1 PENDAHULUAN

Dalam rangkaian BAB I akan membahas tentang beberapa hal seperti latar belakang permasalahan yang terjadi pada PT. Fata Organa Solusi, rumusan masalah yang disimpulkan berdasarkan permasalahan yang sedang dihadapi, batasan masalah yang akan dijadikan sebagai objek penelitian, tujuan serta manfaat dari penelitian maupun sistematika penulisan yang digunakan pada laporan skripsi.

BAB II LANDASAN TEORI

Dalam rangkaian BAB II akan membahas tentang penjelasan dari beberapa teori pendukung yang digunakan terkait dengan topik yang akan diangkat dalam penelitian pada studi kasus PT. Fata Organa Solusi seperti teori tentang functional testing, tentang automation framework, dan juga tentang metode SAW.

BAB III METODOLOGI PENELITIAN

Dalam rangkaian BAB III akan membahas tentang alur penelitian yang mencakup gambaran umum objek penelitian, metode penelitian yang digunakan, teknik pengumpulan data, serta kerangka berpikir yang diterapkan pada penelitian ini.

BAB IV ANALISIS DAN HASIL PENELITIAN

Dalam rangkaian BAB IV akan membahas tentang penjelasan hasil pengujian pada produk Annotation Tool dan juga perbandingan antara beberapa automation framework berdasarkan parameter yang diterapkan dengan metode SAW yang dilakukan berdasarkan permasalahan dan disesuaikan dengan kebutuhan penelitian pada studi kasus PT. Fata Organa Solusi.

BAB V SIMPULAN DAN SARAN

Dalam rangkaian BAB V akan membahas mengenai kesimpulan yang diperoleh dari hasil penelitian yang telah dilakukan beserta beberapa saran untuk pengembangan penelitian selanjutnya.

