BAB II
LANDASAN TEORI

2.1 Pengujian Otomatis

Proses pengujian merupakan tahapan untuk memastikan apakah sistem yang sedang dikembangkan memiliki kendala atau tidak. Maka dari itu proses pengujian haruslah dilalui agar suatu sistem dapat memenuhi standar dan minim akan *error* atau kesalahan sebelum sampai ke tangan pengguna. Pengujian sendiri terdiri dari dua jenis yaitu manual dan otomatis. Proses pengujian manual merupakan bagian dari pengujian sebuah perangkat lunak dimana pengujian secara manual membutuhkan bantuan input dari manusia, analisis dan juga evaluasi (Padmini, 2004). Pengujian manual sebenarnya memiliki manfaat yang relatif lebih murah dan efektif hanya saja rentan akan kesalahan oleh penguji sendiri karena kelelahan ataupun faktor lain.

Sedangkan pengujian secara otomatis juga bagian dari jenis pengujian perangkat lunak. Pengujian otomatis merupakan proses pengujian pada sebuah perangkat lunak dengan menggunakan sebuah alat bantu untuk melakukan proses pengujian (Padmini, 2004). Pengujian otomatis membutuhkan sebuah pengetahuan berupa skenario pengujian yang dijadikan bahan atau landasan untuk melakukan proses pengujian. Pengujian secara otomatis sendiri dapat dikatakan sebagai proses pengujian yang konsisten karena kita dapat melakukan proses pengujian yang sama di karenakan berdasarkan pada sebuah skenario pengujian sehingga apabila terjadi perubahan pada sistem langkah-langkah pengujian masih akan tetap sama. Pada proses pengujian otomatis hal-hal yang dibutuhkan yaitu berupa analisis sistem yang digunakan sebagai dasar untuk membuat skenario pengujian yang nantinya skenario pengujian tersebut digunakan pada alat bantu pengujian otomatis. Pengujian otomatis memiliki kelebihan dan kekurangan diantaranya:

a. Kelebihan
    1. Tingkat akurasi yang tinggi.
    2. Kelengkapan informasi selama pengujian.
    3. Sumber daya yang lebih sedikit.
    4. Durasi yang lebih singkat.
    5. Lingkupnya lebih dari pengujian secara manual.

b. Kekurangan
   1. Beberapa *tools* pengujian otomatis bersifat berbayar.
   2. Tidak mendapatkan *feedback* dari sisi *user experience*.

2.2 Pengujian Regresi

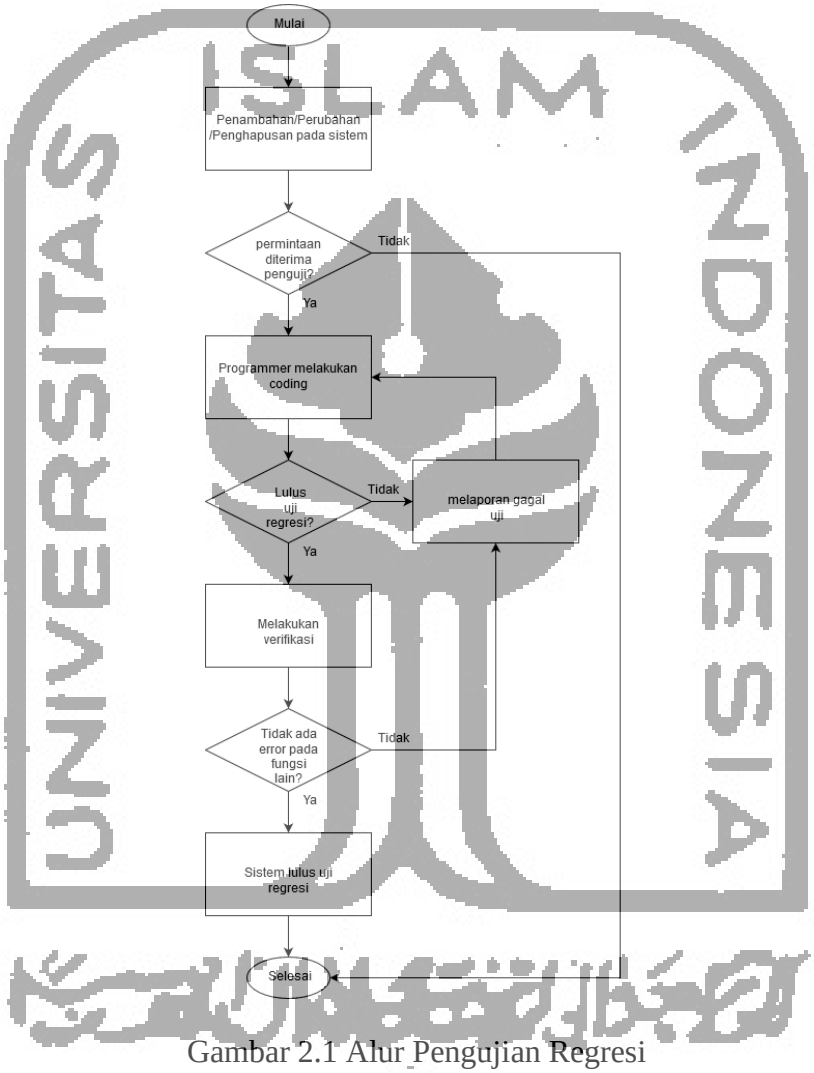
Pada saat proses pengembangan perangkat lunak, sistem pasti mengalami perubahan baik secara signifikan ataupun tidak yang dapat berpengaruh pada proses pengujian karena dapat menyebabkan masalah pada fungsi yang sebelumnya sudah bekerja dengan baik justru dapat berubah menjadi kendala. Pengujian dengan menerapkan metode regresi adalah salah satu solusi dari suatu kasus pengujian dalam pengembangan sebuah perangkat lunak. Pengujian perangkat lunak secara regresi adalah suatu proses pengujian dimana untuk membantu dalam memastikan perubahan pada sistem yang sedang dikembangkan tidak mengganggu fungsi yang lain (Pressman, 2010).

Proses pengujian sendiri dapat dilakukan secara manual ataupun otomatis, pengujian regresi secara manual dapat dilakukan dengan cara menjalankan sebagian pengujian dari semua skenario yang sudah dibuat. Sedangkan untuk pengujian regresi secara otomatis dapat dilakukan dengan menggunakan sebuah alat bantu yang memungkinkan pada saat sedang dilakukan proses pengujian dapat dilakukan proses perbandingan hasil yang baru diperoleh dengan hasil yang sebelumnya sudah diperoleh dari pengujian sistem tersebut. Pengujian regresi masuk kedalam lingkup pengujian otomatis, maka dari itu kelebihan dan kekurangan pada pengujian regresi kurang lebih sama dengan pengujian otomatis hanya saja pengujian regresi menjamin bahwa perubahan kode tidak mengganggu fungsi yang lain pada sistem. Pengujian regresi pada penelitian ini terletak pada saat melakukan proses pengujian berdasarkan alur bisnis dengan melakukan perhitungan waktu regresi. Ketika dilakukan proses pengujian, apabila dilakukan penambahkan, perubahan atau penghapusan fitur baru, maka pengujian selanjutnya dapat dilakukan dengan hanya memilih tombol *playback* saja tanpa harus mengulangi langkah-langkah yang ada pada skenario pengujian. Pada proses pengujian regresi yang di terapkan pada aplikasi berbasis web, ada beberapa tahapan menurut (Zarrad, 2015), diantaranya:

1. Pengembangan aplikasi web berupa analisis dan perancangan *test case*.
2. Melakukan pengujian *Unit Testing*.
3. Mengevaluasi hasil tes.

4. Melakukan modifikasi berupa perubahan, penambahan atau penghapusan fungsionalitas sistem.
5. Memastikan fungsionalitas baru yang ditambahkan tidak mengganggu fungsionalitas yang lain.

Setelah melakukan tahapan diatas maka pengujian dapat dikatakan sebagai pengujian regresi, implementasi proses pengujian regresi pada sistem ditunjukan pada Gambar 2.1.



Gambar 2.1 Alur Pengujian Regresi

2.3 Sistem Pengelolaan Surat Yayasan Badan Waqaf UII

Sistem Pengelolaan Surat Yayasan Badan Waqaf Universitas Islam Indonesia adalah sebuah sistem yang sedang dikembangkan saat ini yang digunakan untuk melakukan proses pengelolaan surat. Sistem Pengelolaan Surat Yayasan Badan Waqaf Universitas Islam Indonesia ini berfokus pada fitur utamanya yaitu proses disposisi surat antara karyawan-karyawan yang terdapat di lingkungan Yayasan Badan Waqaf Universitas Islam Indonesia dan juga proses pencarian surat. Pada proses pengembangan sistem, proses pengujian yang

dilakukan atau diterapkan pada sistem hanya sebatas pada apakah ditemukannya kesalahan atau *error* pada sistem atau tidak, dan juga proses pengujian yang dilakukan dengan berdasarkan pada skenario pengujian yang sudah dibuat. Dengan diterapkannya proses pengujian secara otomatis dengan menggunakan metode regresi ini diharapkan dapat meminimalisir kesalahan atau *error* yang terdapat pada sistem yang sedang dikembangkan, dan juga diharapkan sistem dapat berjalan dengan baik dan sempurna tanpa adanya kesalahan atau *error* mulai dari proses awal surat masuk sampai dengan proses surat keluar (Pratama, 2019).

Suatu proses pengujian secara otomatis dengan menerapkan metode regresi merupakan salah satu solusi yang cocok untuk melakukan proses pengujian pada sistem yang sedang dikembangkan. Karena proses pengujian dengan metode regresi menerapkan pengujian secara terstruktur, yaitu selama proses pengujian memastikan fitur baru yang ditambahkan tidak mengganggu fungsi yang lain. Maka dari itu Sistem Pengelolaan Surat Yayasan Badan Waqaf Universitas Islam Indonesia dapat menerapkan proses pengujian secara otomatis dengan menerapkan metode regresi, agar selama proses pengembangan sistem dan pengujian sistem, dapat diminimalisir kesalahan atau *error*nya, sehingga sistem dapat berjalan dengan baik ketika sudah sampai di tangan pengguna.

2.4 *Selenium IDE*

Selenium merupakan sebuah alat bantu pengujian sebuah perangkat lunak yang bersifat terbuka untuk sistem *web* pada *browser* yang juga mendukung beberapa bahasa pemrograman (Niranjanamurthy, Kumar, Srinivas, & Manoj, 2014). *Selenium* terdiri dari beberapa jenis, seperti *Selenium IDE, Selenium RC, Selenium Web Driver,* dan Selenium Grid. Dari setiap jenis, masing-masing *Selenium* memiliki pendekatan tersendiri untuk melakukan proses pengujian secara otomatis. Beberapa jenis *Selenium* digunakan hanya untuk melakukan proses merekam kembali proses pengujian. Seperti *Selenium IDE* adalah sebuah *plugin* yang dapat digunakan pada *web browser Firefox* yang menyediakan sebuah fasilitas untuk merekam, mengedit, dan juga melakukan proses *debugging* selama proses pengujian sehingga penguji dapat berfokus pada proses pengembangan kasus uji. Sedangkan *Selenium RC* adalah sebuah alat pengujian otomatis yang bersifat dinamis di karenakan menggunakan bahasa pemrograman. Kemudian *Selenium Web Driver* yang lebih berfokus kepada proses implementasi pendekatan moderen dan stabil dalam mengotomasi tindakan pada *web browser*, dengan kata lain proses pengujian dilakukan dengan berkomunikasi langsung dengan *web browser*. Lalu *Selenium Grid* yang merupakan alat yang digunakan secara bersamaan dengan

menggunakan *Selenium RC* sebagai alat yang menjalankan tes secara paralel di berbagai mesin dan *browser* secara bersamaan.

Dari berbagai jenis *Selenium* yang ada, pada *Selenium IDE* memiliki beberapa kelebihan seperti yang ditemukan pada penelitian (Yuda, 2019) diantaranya:

a. *Selenium IDE* merupakan alat bantu pengujian otomatis *open source*.
b. *Selenium* mendukung bahasa *Java*, *.Net*, *Perl*, *PHP*, *Pyhton*, dan *Ruby*.
c. *Selenium IDE* dapat digunakan pada sistem operasi *Windows*, *MacOs*, dan *Unix*.
d. Untuk melakukan pengujian otomatis dengan *Selenium IDE* dapat dilakukan dengan menambahkan *plugin* pada aplikasi *web*.
e. Dalam pemakaian *Selenium IDE* membutuhkan sedikit keahlian.
f. Kemampuan memprogram juga sedikit dibutuhkan pada pengujian dengan *Selenium IDE*.

Selenium IDE juga memiliki kekurangan diantaranya:

a. *Selenium IDE* hanya mendukung aplikasi berbasis web.
b. *Selenium IDE* tidak dapat merekam aktifitas diluar web browser.

Pada penelitian ini berfokus pada penggunaan *Selenium IDE* sebagai alat bantu pengujian otomatis dengan menerapkan metode regresi. *Selenium IDE* sangat berguna dalam melakukan proses pengujian fungsional sistem dan yang berhubungan dengan *web browser*. Hasil yang diperoleh dari proses pengujian menggunakan *Selenium IDE* juga dapat dikatakan cukup kompleks di karenakan melibatkan proses integrasi dari banyak komponen lainnya. Proses pengujian secara otomatis dengan menerapkan metode regresi dan alat bantu *Selenium IDE* diperlulan untuk membuat pengembang perangkat lunak lebih teratur dalam melakukan proses pengujian.

2.5 Studi Literatur

Proses pengujian regresi secara otomatis dengan menggunakan *Selenium IDE* biasa digunakan untuk melakukan proses pengujian secara otomatis pada perangkat lunak yang sedang dikembangkan. Hal tersebut dikarenakan proses pengujian dengan menggunakan *Selenium IDE* lebih mudah dalam penggunaanya. Penguji dapat melakukan proses perekaman, mengedit dan juga melakukan *debugging* pada selama proses pengujian. Sehingga proses pengujian menjadi lebih teratur, terkontrol dan juga dimungkinkan dilakuan proses pengulangan untuk melakukan proses pengecekan kembali. Selama penulis melakukan proses

studi literatur, ditemukan beberapa contoh penggunakan *Selenium IDE* untuk proses pengujian secara otomatis sebuah perangkat lunak yang sedang dikembangkan.

Seperti pada Analisa Uji Coba Aplikasi *Web* Tugas Akhir (TADJ) *Selenium IDE* digunakan pada saat proses pengujian. Proses pengujiannya sendiri menerapkan 2 metode yaitu yaitu dengan menggunakan metode *black box* dan juga *Selenium IDE*. Metode *black box* digunakan untuk proses kategori pengujian validasi *login* mahasiswa pada *website*, pengujian *interface* pengguna, pengujian data, dan juga pengujian performansi. Sedangkan pengujian dengan menggunakan *Selenium IDE* untuk menghasilkan struktur alur tiap perintah yang digunakan dalam menjalankan fungsi-fungsi dari sistem yang sedang dikembangkan. Pengujian menggunakan *Selenium IDE* dilakukan dengan menggunakan *plugin* pada *Firefox* yang digunakan untuk mengembangkan kasus uji. Penelitian ini memiliki kelebihan dan kekurangan yaitu kelebihanya adalah proses pengujian tidak berfokus hanya satu metode, kekurangannya yaitu pada proses pengujian *Selenium IDE* hanya digunakan sebagian pembanding tidak digunakan lebih jauh (Lucyda, 2015).

Kemudian ditemukan juga pada pengujian Tingkat Komponen dan Implementasi Pengujian Mutasi pada Aplikasi Berbasis *Web*. Proses pengujian secara otomatis dilakukan dengan menggunakan *Selenium IDE*, dimana penggunaan *Selenium IDE* dilakukan untuk proses pengujian fungsional pada tingkat komponen dari sistem yang sedang dikembangkan. Tahapan pengujian dimulai dengan menyiapkan skenario pengujian terlebih dahulu yang dimana skenario pengujian tersebut akan diterapkan pada *Selenium IDE*. Pengujian tingkat komponen dengan menggunakan *Selenium IDE* memastikan apakah komponen yang terdapat pada sistem yang sedang dikembangkan dan dilakukan proses pengujian sudah sesuai dengan spesifikasi yang sudah ditentukan saat dilakukan proses pengujian secara otomatis dengan menggunakan *Selenium IDE*. Pengujian menggunakan *Selenium IDE* ditentukan dengan dua hal, apabila proses eksekusi pengujian berhasil maka *Selenium IDE* akan menunjukan indikator berwarna hijau. Sedangkan apabila saat eksekusi pengujian tidak berhasil maka akan ditunjukan dengan indikator berwarna merah. Pada penelitian ini memiliki kelebihan dan juga kekurangan yaitu kelebihannya adalah proses pengujian yang dilakukan berdasarkan pada skenario pengujian dan juga hasil dari pengujian dengan *Selenium IDE*, sedangkan kekurangannya adalah penggunaan *Selenium IDE* hanya pada bagian pengujian tingkat komponen (Hawari et al., 2015).

Pengujian menggunakan *Selenium IDE* juga ditemukan pada Perancangan dan Implementasi Sistem Informasi Manajemen Material Berbasis *Web*. Pengujian dengan

menggunakan *Selenium IDE* digunakan untuk membandingkan hasil akhir dari pengujian yang dimana proses pengujian yang digunakan yaitu proses pengujian dengan menggunakan metode *black box* dengan *Selenium IDE* dan juga proses pengujian *white box*. Pada pengujian otomatis dengan menggunakan *Selenium IDE* digunakan untuk memastikan apakah fungsional pada sistem dapat berjalan dengan baik atau tidak. Pengujian dengan menggunakan *Selenium IDE* juga digunakan untuk memastikan apakah sistem dapat berjalan dengan semestinya dari proses awal sampai dengan proses akhir. Sehingga apabila ditemukan sebuah kesalahan atau *error* sebelum sampai selesai dapat di identifikasi terlebih dahulu kesalahan atapun *error* tersebut. Pada penelitian ini memiliki kelebihan dan juga kekurangan yaitu kelebihannya adalah pada pengujian menggunakan dua buah metode sedangkan kekurangannya adalah hasil pengujian dilakukan dengan menggunakan 2 buah alat bantu pengujian dimana apabila hasilnya dibandingkan bisa tidak seimbang dikarenakan setiap alat bantu pengujian memiliki kelebihan dan juga kekurangan (Patria, Sadwono, & Komarudin, 2013).

Pengujian menggunakan *Selenium IDE* juga ditemukan pada Pengujian Fungsionalitas *Website* Berbasis *UML Activity diagram* . Pengujian dengan menggunakan *Selenium IDE* digunakan untuk makukan proses pengujian dari kasus uji yang sudah dibuat oleh peneliti sebelumnya. Pada penelitian ini *Selenium IDE* menjadi penentu juga apakah apakah *test derivation* berhasil atau tidak. Pada penelitian ini memiliki kelebihan dan juga kekurangan, kelebihan dari penelitian ini adalah sebelum melakukan pengujian dengan menggunakan *Selenium IDE* peneliti terlebih dahulu melakukan pembuatan kasus uji, sehingga lebih mudah pada penggunaan *Selenium*nya. Sedangkan kekurangan penelitian ini adalah pembuatan *command* pada *Selenium IDE* dilakukan secara otomatis (Yuda, 2019).

Pada penelitian-penelitian yang sudah ada sebelumnya dapat diketahui bahwa *Selenium IDE* hanya digunakan sebagai alat bantu pengujian secara otomatis. Penelitian ini berfokus pada proses pegujian yang dilakukan dengan merancang skenario pengujian terlebih dahulu yang nantinya skenario pengujian tersebut dapat diterapkan pada alat bantu pengujian yaitu *Selenium IDE*. Skenario pengujian juga digunakan sebagai dasar pengetahuan selama proses pengujian yang nantinya akan diterapkan pada pengujian secara otomatis dengan menggunakan metode regresi dengan alat bantu *Selenium IDE* yang nantinya berguna untuk memastikan apakah sistem dapat berjalan dengan baik dan benar yang dimulai dari proses awal sampai dengan proses akhir sesuai dengan skenario yang sudah dirancang sebelumnya.